Empirical Software Engineering (2025) 30:176
https://doi.org/10.1007/s10664-025-10710-5

Exploring the black box: analysing explainable AI challenges and best practices through stack exchange discussions

Mohammad Mahdi Sayyadnejad[1] · **Ali Asgari**[2] · **Ashkan Sami**[3] · **Hooman Tahayori**[1]

Accepted: 28 July 2025

Abstract
Explainable Artificial Intelligence (XAI) is a crucial domain within research and industry, aiming to develop AI models that provide human-understandable explanations for their decisions. While the challenges in AI, deep learning, and big data have been extensively explored, the specific concerns of XAI developers have received limited attention. To address this gap, we analysed discussions on Stack Exchange websites to delve into these issues. Through a combination of automated and Manual analysis, we identified 6 overarching categories, 10 distinct topics, and 40 sub-topics commonly discussed by developers. Our examination revealed a steady rise in discussions on XAI since late 2015, initially focusing on conceptualisation and practical applications, with a notable surge in activity across all topic categories since 2019. Notably, Concepts and Applications, Tools Troubleshooting, and Neural Networks Interpretation emerged as the most popular topics. Troubleshooting challenges were commonly encountered with tools like SHAP, ELI5, and AIF360, while visualisation issues were prevalent with Yellowbrick and SHAP. Furthermore, our analysis suggests that addressing questions related to XAI poses greater difficulty compared to other machine-learning questions.

Keywords Artificial intelligence · Explainability · Interpretation · Transparency · Stack exchange · Human-Centered AI

Communicated by: Davide Falessi.

✉ Ashkan Sami
a.sami@napier.ac.uk

Mohammad Mahdi Sayyadnejad
mmehdisayyad@gmail.com

Ali Asgari
A.Asgari-2@tudelft.nl

Hooman Tahayori
tahayori@shirazu.ac.ir

[1] Department of Computer Science & Engineering and IT, Shiraz University, Shiraz, Iran

[2] Department of Software Technology, Delft University of Technology, Delft, The Netherlands

[3] Computer Science Subject Group, Edinburgh Napier University, Edinburgh, UK

Published online: 20 October 2025

1 Introduction

Artificial Intelligence (AI) algorithms are becoming increasingly embedded in diverse aspects of modern life, from industrial applications to healthcare and finance (Reddy et al. 2020; Lee et al. 2018). However, as these models—particularly complex deep learning models—advance in sophistication, they often become less interpretable and harder to understand (Guidotti et al. 2018). This lack of transparency poses challenges, especially in high-stakes domains where AI-driven decisions must be understandable to foster trust, accountability, and user confidence (Hagras 2018). Furthermore, regulatory frameworks like the General Data Protection Regulation (GDPR) mandate individuals' rights to meaningful explanations of automated decision-making processes, emphasising the need for transparent AI systems (Goodman and Flaxman 2017; Malgieri and Comandé 2017). Consequently, there is an urgent demand for AI systems that not only perform well but also offer transparency to users with diverse backgrounds and levels of expertise (Hagras 2018).

Explainable Artificial Intelligence (XAI) has emerged to address this need, focusing on making AI systems more transparent and accountable, which, in turn, increases user trust and enhances decision-making processes (Danilevsky et al. 2020). For effective XAI systems, interpretability, transparency, and accountability must be prioritised in both development and application (Treviso and Martins 2020).

Reflecting the growing importance of XAI, recent studies (Mersha et al. 2024; Schwalbe and Finzel 2024) show a marked increase in XAI-related publications, especially since 2018. This trend highlights XAI's role as a critical research area, with many researchers focusing on developing tools and methods that make complex, black-box AI models more interpretable. While tools such as SHAP and LIME provide substantial value, they often bring specific challenges and complications when applied in real-world scenarios, potentially impacting their usability and effectiveness.

As developers encounter and address these challenges in practical settings, they increasingly share their experiences, insights, and issues in open-source communities, facilitating collaborative learning and innovation (Son and Kim 2023). Stack Exchange, the world's largest developer community, serves as a key platform for XAI discussions. Beyond troubleshooting, these forums reveal broader trends and technological advancements, providing a unique window into real-world applications and the challenges faced by XAI practitioners.

Several studies have used Stack Exchange data to understand developers' challenges across various domains, including machine learning (Alshangiti et al. 2019), big data (Bagherzadeh and Khatchadourian 2019), deep learning (Han et al. 2020), and AutoML (Wang et al. 2023). However, most XAI studies to date have focused on surveys and model selection strategies, lacking a comprehensive analysis of real-world discussions and developer concerns in practical settings. Critical questions, such as the challenges users face in real-world XAI scenarios, the tools most frequently employed, the difficulties with each tool, and the evolving nature of these challenges, remain largely unexplored.

In this study, we leverage data from Stack Exchange to examine these questions and gain deeper insights into the real-world obstacles associated with implementing XAI systems. Our approach combines automated topic modelling through Latent Dirichlet Allocation (LDA) to identify general themes, followed by manual card sorting to categorise and refine subtopics. This hybrid methodology enables us to systematically categorise the topics

discussed by XAI practitioners and provides a structured framework for analysing XAI challenges. To address the existing gap, our research focuses on the following questions:

RQ1: What Topics are XAI Developers Asking About? As mentioned earlier, exploring the real-world queries and obstacles encountered by developers and users of XAI systems is crucial. This exploration is necessary to fill knowledge voids and enhance comprehension of XAI developers' needs. From the outcomes of both LDA and manual examination of topics, We identified a total of 10 topics and 40 subtopics, which were then categorised into 6 high-level categories: Troubleshooting (38.14%), Feature Interpretation (20.22%), Model Analysis (13.81%), Data Management (6.41%), Visualisation (14.31%), and Concepts and Applications (7.11%). Key topics include Tools Troubleshooting, Model Barriers, and Visualisation, with the largest subtopics being Plot Customisation and Styling, Tools Implementation and Runtime Errors, and Model Misconfiguration and Usage Errors.

RQ2: What Kind of Questions are XAI Developers Asking? After identifying the most common and significant XAI topics in RQ1, the subsequent step entails analysing the types of questions posed by XAI developers on technical Q&A platforms. Previous studies (Rosen and Shihab 2016; Abdellatif et al. 2020; Alamin et al. 2023) have highlighted that developers pose various types of questions, including "how", "why", "what", and "others". Analysing these questions can provide insights into the nature of challenges encountered during XAI development. Our analysis reveals that "how" questions dominate, particularly prominent in discussions related to Visualisation (67.5%) and Data Management (66.67%), suggesting a strong emphasis on practical implementation. "What" questions indicate a desire for comprehension, with higher proportions observed in XAI Concepts and Applications (45%) and Model Analysis (23.%). Troubleshooting (20.0%) encompasses a notable share of "why" questions, reflecting an interest in understanding underlying AI issues, and "Others" questions thrive in XAI Concepts and Applications (10%).

RQ3: What Topics Exhibit Higher Levels of Popularity and Difficulty? Upon identifying prevalent and critical topics in XAI and discerning the types of questions posed by XAI developers, the subsequent phase involves examining the difficulty and popularity levels of questions within each topic. Understanding the popularity and difficulty of different topics can aid in pinpointing areas necessitating further research and development.

We assess three metrics of popularity (score counts, view counts, and comment counts) and two metrics of difficulty (percentage of questions without accepted answers and average time for a question to receive an accepted answer). Notably, Concepts and Applications, Tools Troubleshooting, and Neural Networks Interpretation emerge as the most popular topics. Conversely, topics related to model analysis and troubleshooting present notable challenges, often lacking accepted answers. Sub-topics exploring the Importance and Implications of XAI and addressing Installation Issues exhibit considerable popularity. Conversely, challenges are evident in sub-topics such as Layer-wise Relevance Propagation (LRP) and Gradient-based Methods, with significant unanswered questions. Moreover, Library Compatibility and Version Issues pose notable challenges, with a substantial number of questions remaining unanswered.

RQ4: Which Technologies are XAI Developers Using? As XAI grows, developers must keep pace with the latest technologies and their challenges. Understanding the programming languages and tools in use, along with their associated hurdles, offers valuable insights for creating transparent AI systems. By recognising the technologies prevalent in Q&A platforms, developers can make informed decisions about tools and models for their projects. This ensures accuracy, efficiency, and transparency in their systems, fostering the development of improved tools. Staying updated with the latest XAI technologies is essential for success in this rapidly evolving field.

Each question within the Stack Exchange forums is associated with tags representing the high-level categories of the questions. Analysing these tags can offer context to specific topics (Alfayez et al. 2023). We first determine tag frequencies in the XAI dataset posts within each forum. We focus on the commonly used tags to assess software packages and programming languages. Additionally, we analyse popularity, difficulty, and category distribution similar to RQ3. Across forums, SHAP is the top tag among 482 unique ones. Key AI tools include Scikit-learn, TensorFlow, XGBoost, Keras, and Matplotlib. Python and R are the main XAI languages, with Python queries taking 5 hours longer on average than R for accepted solutions, and R having more unanswered questions. In Q&A forums, popular tools are SHAP, ELI5, Yellowbrick, DALEX, LIME, and AIF360. SHAP leads with 67.2% iterations, while DALEX, LIME, and AIF360 are the most challenging tools. Troubleshooting is common for SHAP, ELI5, and AIF360, while visualisation issues are frequent for Yellowbrick and SHAP. LIME and DALEX primarily face Model barrier issues.

RQ5: How do the XAI Topics Evolve? XAI encompasses a range of topics, each exhibiting distinct characteristics. Our endeavour involves monitoring the evolution of these categories to provide ongoing support to the expanding XAI community, pinpointing areas warranting further attention.

Drawing from prior research, we undertake a comprehensive analysis to quantify each category's absolute and relative impact. This entails assessing the monthly influx of new questions within each category and determining the proportion of new questions added to a category compared to others. Our findings indicate that discussions surrounding XAI commenced in late 2015, initially emphasising conceptualisation and practical applications. Since 2019, these discussions have garnered increased interest, witnessing an initial dominance of Troubleshooting and Feature Interpretation topics, followed by a recent surge in the popularity of Visualisation-related questions.

2 Background

XAI is gaining audience and traction, driven by a burgeoning community. Despite its broad applications, XAI system development poses challenges, sparking active discussions on platforms like StackExchange. To comprehend these discussions, topic modelling is essential. Numerous studies have explored XAI, revealing trends, challenges, and solutions. This approach helps structure and analyse conversations, offering insights for both XAI enthusiasts and software engineering researchers. It aids in identifying key areas of interest and

difficult aspects of developing an XAI system, knowledge gaps, and collaboration opportunities, propelling the advancement and adoption of XAI in diverse sectors.

2.1 Topic Analysis in Q&A Platforms

The main focus of various studies lies in utilising topic modelling and unstructured data from software repositories, which has been a proven and effective methodology for uncovering various software engineering challenges (Bridge 2011).

Barua et al. (2014) pioneered this approach and analysed what developers generally discuss. Rosen and Shihab (2016) used topic modelling on Stack Overflow data to categorise challenges in mobile development, and they categorised the questions of mobile developers into 6 categories. Similar techniques have since been applied to investigate challenges in chatbot development (Abdellatif et al. 2020), IoT (Uddin et al. 2021), modern release engineering (Openja et al. 2020), low code software development (Alamin et al. 2023), big data (Bagherzadeh and Khatchadourian 2019), blockchain (Wan et al. 2019) and Docker containerization (Haque et al. 2020). Li et al. (2021) provide a multi-platform perspective on quantum software engineering by combining Stack Exchange and GitHub Issues data and identified 9 topics through stack exchange data. Other studies have manually classified the topics addressed within question and answer platforms focusing on MATLAB (Naghashzadeh et al. 2021), Technical Debt (Alfayez et al. 2023), and discussions related to code smells and anti-patterns (Tahir et al. 2020).

Extensive research has been conducted in the domain of AI and machine learning to analyse discussions and categorise the challenges developers encounter. This process commenced with investigating machine learning questions on Stack Overflow (Alshangiti et al. 2019), revealing that the primary challenges are concentrated in the data preprocessing and model deployment stages. Han et al. (2020) delved into the specific context of deep learning frameworks, analysing GitHub and Stack Overflow data using LDA topic modelling and analysing their trends, popularity, and difficulty. Chen et al. (2020) also analysed challenges in deploying deep learning-based software and classified developers' discussions into 3 main categories: 25 topics and 72 detailed sub-topics. Similarly, Wang et al. (2023) investigated challenges in AutoML, employing manual analysis of Stack Overflow data.

2.2 Explainable AI

The rapidly growing body of research on XAI underscores the crucial need to understand complex AI models in this domain, as highlighted in a systematic review by Cao et al. (2024). Also, the significance of trust in human interactions with machine learning systems and the importance of providing explanations for individual predictions to foster that trust are emphasised (Ribeiro et al. 2016). Several surveys have explored different aspects of XAI, commonly classifying methods based on their operational scope local or global, procedural stage ante-hoc or post-hoc, and output format numerical, visual, textual, or hybrid (Angelov et al. 2021; Tjoa and Guan 2020; Vilone and Longo 2021; Minh et al. 2022; Speith 2022).

Previous literature (Yang et al. 2023) explores the applications of XAI in fields such as medicine and cybersecurity. They address model limitations, propose human-centred research directions, introduce a taxonomy for classifying XAI methods, and suggest prom-

ising avenues, including context-aware XAI, interactive explanations, as well as hybrid models. Watson (2022) highlights challenges in interpretable machine learning, noting that XAI-generated algorithms can be unfamiliar and resource-intensive and demand increased user input. Saeed and Omlin (2023) emphasise the importance of effectively communicating data quality in AI design for fairness, performance, and explainability while also discussing XAI's role in integrating human knowledge, addressing challenges and exploring applications in human-machine collaboration. Furthermore, Arrieta et al. (2020) emphasise audience-centric explainability in AI and introduce two taxonomies: one distinguishing post-hoc explainability from inherent transparency and another evaluating models' suitability for explaining deep learning algorithms. Another survey (Sahakyan et al. 2021) discusses the explainability of AI with tabular data, categorising the literature into various techniques such as feature importance, feature interaction, simplified models, and counterfactuals. Additionally, Das and Rad (2020) presents a comprehensive taxonomy that categorises explanations in deep learning based on their scope, methodology, and level of explanation. They further examine both the limitations and advancements in explanation visualisations.

Prior research (de Bruijn et al. 2022) classifies challenges in the XAI community, including the need for public expertise, contested explanations, variable explanations, diverse data sources, and algorithms. Ding et al. (2022) examine XAI's trust-building role in complex AI systems, offering a taxonomy for methods, differentiating local and global explanations, and advocating for evaluating explanations using both human-centred metrics such as user satisfaction and mental model alignment. Additionally, XAI techniques provide versatile methods for explaining complex models; Ribeiro et al. (2016) introduce LIME, an adaptable and expandable approach to explaining predictions in a comprehensible manner, while Lundberg and Lee (2017) introduce SHAP, a powerful framework for interpreting predictions through feature importance values. In the case of available tools in XAI, Previous studies (Dwivedi et al. 2023; Ding et al. 2022) have identified various XAI tools, Among these tools, SHAP, LIME, and ELI5 have been consistently highlighted as essential tools.

Lastly, Visual explanations, including techniques like Saliency maps (Simonyan et al. 2013) and Grad-CAM (Selvaraju et al. 2017), play a crucial role in XAI by understanding complex model decision-making, while recent advancements, such as Grad-CAM++ (Chattopadhay et al. 2018), address limitations in localisation and explanation. Alicioglu and Sun (2022) address challenges and future directions in visual analytics for interpreting neural networks.

Despite various studies that have identified challenges in XAI or explored its applications across different domains (Saeed and Omlin 2023; Ding et al. 2022; Hanif et al. 2021; Mersha et al. 2024), there remains a lack of focus on the practical challenges faced by developers during the implementation of XAI systems. Most prior research has approached XAI challenges from theoretical perspectives or through user-centric evaluations, often employing surveys or literature reviews. However, these methods may not fully capture the nuanced, real-world difficulties that practitioners encounter. To address this gap, our study uniquely analyses developers' discussions on Stack Exchange platforms to identify and specify the importance of practical challenges in XAI development. By gaining insights into the technical issues that developers face, we aim to contribute to the advancement of XAI by addressing practical concerns within the development community. Our research aims to foster a deeper understanding of XAI implementation challenges, thereby facilitat-

ing its responsible and effective application in real-world scenarios, ultimately enhancing acceptance, improving quality, and informing the development of more effective tools.

3 Methodology

The Methodology section is divided into two main parts. The first part describes the datasets, preprocessing algorithms, and analytical techniques used throughout the study. The second part outlines our research questions, explains the motivation behind them, and describes the approach we take to address them.

3.1 Data Collection and Analysis

Collecting XAI-related posts involves six steps, which are explained in detail in this section. The entire process is illustrated in Fig. 1.

Step 1: Collecting Q&A data. Our dataset was obtained from the Stack Exchange Data Explorer (Stack Exchange Data Explorer 2023), a platform offering comprehensive and current data extracted from Stack Exchange "data dumps". After reviewing multiple forums, we identified Stack Overflow (widely recognised as a key resource for researchers exploring various software engineering topics Naghashzadeh et al. 2021; Ahmed and Bagherzadeh 2018; Rosen and Shihab 2016; Michael Ayas et al. 2023) along with the Artificial Intelligence and Data Science forums, as esteemed hubs within the XAI community. These forums offer valuable insights into the challenges and solutions related to XAI development. Consequently, our research focused on three Stack Exchange forums: "Stack Overflow" (Stack Overflow 2023), "Artificial Intelligence" (AI Stack Exchange 2023), and "Data Science" (Data Science Stack Exchange 2023).
Step 2: Identify XAI-related tags. To facilitate topic-based classification and searching, Stack Exchange websites utilise a system of user-defined tags assigned to posts (Barua et al. 2014). For our research, we utilised these tags to identify relevant posts.

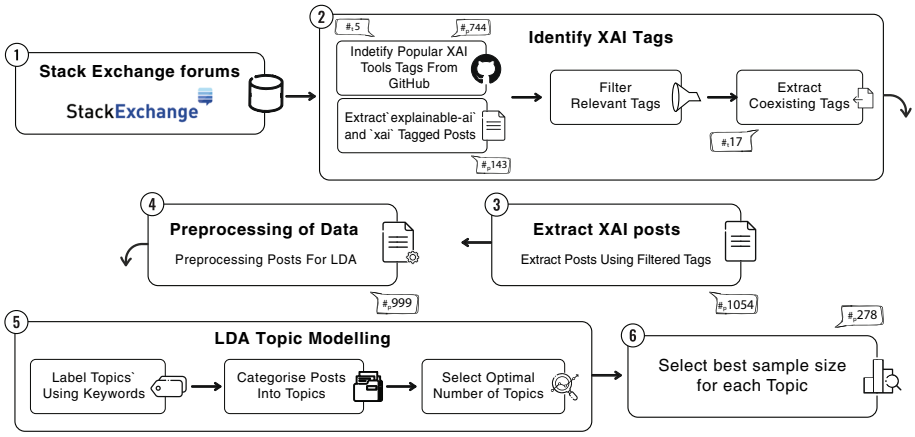

Fig. 1 Overview of study methodology

We employed a two-phase approach for the extraction of tags related to XAI. Initially, we focused on identifying the most widely used open-source packages and frameworks within the realm of XAI. Subsequently, we concentrated on identifying XAI-related tags, utilising specific thresholds of the tag significance threshold (TST) and the tag relevance threshold (TRT). This methodology ensured a comprehensive and focused extraction of pertinent tags in the field of XAI.

1. *Identify the most popular open-source XAI packages and frameworks Tags.* To achieve the most popular XAI tools, we first conducted a search using the GitHub Search API, employing key terms such as "explainable AI," "XAI," "explain model," and "interpretable machine learning." This initial phase yielded a comprehensive List of 7,840 GitHub repositories. Building on methodologies used in previous studies (Wang et al. 2023; Braiek et al. 2018), we then applied a filtering process to refine this list. Our primary criterion was the cumulative number of stars garnered by each repository, considering this as an indicator of a tool's popularity and acceptance in the community. We focused on repositories that collectively accounted for 80% of the total number of stars attributed to XAI projects on GitHub, which narrowed our List to 50 repositories with the highest star counts. A manual review process was undertaken to ensure our findings' relevance. Each of the 50 repositories was carefully examined to differentiate actual tools, such as packages or frameworks, from other types of content, such as educational Materials, books, or applications. This detailed scrutiny led us to identify 35 tools representing the most popular XAI open-source tools in the GitHub community, shown in Table 1. By searching for tags related to the 35 identified tools across three Stack Exchange forums, we identified five popular XAI tools: SHAP (Lundberg and Lee 2017), LIME (Ribeiro et al. 2016), ELI5 (TeamHG-Memex 2019), AIF360 (Bellamy et al. 2018), and DALEX (Biecek 2018), which had corresponding tags and were added to our list.
2. *Determine tags associated with XAI.* In the subsequent phase, we employed a widely recognised approach, previously used in numerous studies (Li et al. 2021; Alfayez et al. 2023; Abdellatif et al. 2020; Uddin et al. 2021), to expand our collection of tags. Initially, we utilised the tags acquired earlier, along with "explainable-ai" and "xai". Following this, we gathered all questions associated with these tags and any additional tags used in conjunction with them and found 409 unique tags. However, not all associated tags were related to XAI, like C# and Java. We followed the approach used in previous studies (Rosen and Shihab 2016; Bagherzadeh and Khatchadourian 2019; Abdellatif et al. 2020) to identify the most relevant XAI-related tags. We calculated two metrics for each tag: the tag significance threshold (TST) and the tag relevance threshold (TRT). The TRT calculates the ratio of XAI-related posts for a tag compared to the total number of posts having that tag. On the other hand, the TST measures how prominent a tag is in the XAI-tagged posts (Rosen and Shihab 2016; Abdellatif et al. 2020), which is calculated using the total number of XAI posts for that tag and the total number of XAI posts for the initial tag set. We calculate the TRT and TST values as follows,

Table 1 Most popular open-source XAI packages

Name	Stars
shap/shap	19819
marcotcr/lime	10776
jacobgil/pytorch-grad-cam	7823
interpretml/interpret	5619
DistrictDataLabs/yellowbrick	4116
pytorch/captum	4084
PAIR-code/lit	3169
TeamHG-Memex/eli5	2671
MAIF/shapash	2386
Trusted-AI/AIF360	2205
SeldonIO/alibi	2123
oegedijk/explainerdashboard	1911
jalammar/ecco	1762
Trusted-AI/AIX360	1371
ModelOriented/DALEX	1264
csinva/imodels	1152
cdpierse/transformers-interpret	1072
sicara/tf-explain	987
EthicalML/xai	925
RexYing/gnn-model-explainer	713
SelfExplainML/PiML-Toolbox	677
salesforce/OmniXAI	661
ModelOriented/DrWhy	643
hila-chefer/Transformer-MM-Explainability	598
tensorflow/tcav	578
tmadl/sklearn-expertsys	484
polyaxon/traceml	482
christophM/iml	480
ScalaConsultants/Aspect-Based-Sentiment-Analysis	480
thomasp85/lime	476
BCG-Gamma/facet	474
linkedin/FastTreeSHAP	456
understandable-machine-intelligence-lab/Quantus	394
interpretml/interpret-text	373
explainX/explainx	359

$$\mathrm{TRT}_{\mathrm{tag}} = \frac{\text{No. of XAI posts}}{\text{Total no. of posts}} \tag{1}$$

$$\mathrm{TST}_{\mathrm{tag}} = \frac{\text{No. of XAI posts}}{\text{No. of posts in the initial tag set}} \tag{2}$$

A tag is highly relevant to XAI if TRT and TST exceed a specific threshold value. We conducted independent checks using various TRT values, such as 0.05, 0.10, 0.15, 0.20, 0.25, 0.30, and 0.35, along with different TST values, including 0.001, 0.005, 0.010, 0.015, 0.020, 0.025, and 0.03. After collecting the results, the first two authors discussed their individual choices and worked towards a consensus. Ultimately, we found that TRT values higher than 0.1 and TST values higher than 0.001 yield an appropriate balance between including rel-

evant XAI posts and filtering unrelated posts. These values align with findings from previous work (Abdellatif et al. 2020; Li et al. 2021; Uddin et al. 2021; Alamin et al. 2023; Wang et al. 2023). We manually check the description of all 23 retrieved tags and find that 5 of them (i.e., "sensitivity", "vector-semantics", "cause-and-effect", "yellow", and "agglomerative") are not directly related to XAI so we remove them from the list, our final list of tags contains the following 17 tags: { shap, explainable-ai, lime, yellowbrick, xai, eli5, dalex, aif360, feature-importances, beeswarm, shapley, predictor-importance, saliency-map, grad-cam, partial-dependence-plot, boruta, feature-interaction}

Step 3: Extract XAI posts. To begin collecting data for our XAI dataset, we utilised the XAI-related tag set we obtained previously. This tag set served as a basis for identifying relevant posts on the Stack Exchange forums. We selected posts tagged with at least one of these tags to ensure they were related to XAI. To further refine our selection, following previous work (Tahir et al. 2018) we filtered the Posts table using the PostTypeId field, which allowed us to isolate only the questions asked by developers. Specifically, we selected only the posts with a PostTypeId of 1 to extract only questions from the Stack Exchange dataset. With these queries, we obtained 1054 posts that were mostly related to XAI. Recognising that each question may be linked to multiple tags within our defined tag set, it is possible for a single question to appear multiple times within the dataset. To ensure the integrity and quality of our dataset, we eliminated instances that contained identical text in the "Body" attribute, thereby maintaining the uniqueness of each question. After completing these steps, we were left with 999 unique XAI-related posts. This dataset was used in our subsequent analysis.

Step 4: Preprocessing Posts. To ensure the reliability and accuracy of our subsequent analyses on XAI posts (Ahmed and Bagherzadeh 2018; Barua et al. 2014), we undertook several steps to eliminate various forms of noise. This included the removal of code blocks, HTML tags (e.g., < p>, < /p>), URLs, and image tags from the text. Furthermore, we utilised Gensim (Řehůřek and Sojka 2010) to eliminate punctuation and tokenise the texts. Stopwords were then filtered out based on their presence in the NLTK (Loper and Bird 2002) stopword corpus to enhance the precision of the analysis. Additionally, bigrams were created using the Gensim Phrases model to refine the analysis further. Finally, the spaCy (Honnibal et al. 2020) library was employed to perform lemmatisation on the text, retaining only the noun, adjective, verb, and adverb parts of speech.

Step 5: Identify XAI topics. Given the broad categorisation of tags on Stack Exchange, we employ LDA-based topic models to find and capture the fine-grained topics among XAI questions. In our methodology, we employed the Mallet implementation of LDA (McCallum 2002), a method extensively applied in software engineering research due to its greater coherence score compared to the Gensim library (Abdellatif et al. 2020; Li et al. 2021; Alamin et al. 2023; Mahmood et al. 2023). The challenge of using LDA is identifying the optimal number of topics K that the LDA uses to group the posts. High K values lead to very specific topics. On the other hand, if the K value is small, the topics yielded may be too generic. We evaluated alternative K values ranging from 5 to 30 in increments of 1. We executed MALLET LDA on our dataset for a total of 1000 iterations, as outlined in previous studies (Bagherzadeh and Khatchadourian 2019), and computed the coherence metric value of the resultant topics. A strong correlation exists between the coherence metric, which assesses the understandability of the themes

produced by the LDA, and human comprehension. MALLET leverages a pair of crucial hyperparameters denoted as α (alpha) and β (beta). These parameters are pivotal in governing the distribution of words across topics and the allocation of posts to these topics. Drawing from established methodologies in the literature (Alamin et al. 2023; Al Alamin et al. 2021; Uddin et al. 2021; Bagherzadeh and Khatchadourian 2019; Ahmed and Bagherzadeh 2018; Rosen and Shihab 2016), we adhere to conventional values for these parameters. Specifically, α is set to $50/K$, where K represents the number of topics, and β is assigned a value of 0.01. This choice of standard parameters is underpinned by their widespread acceptance and proven effectiveness in prior research, which provides a solid benchmark for our experimental framework in topic modelling. we concluded that $K = 10$ yields the best possible collection of topics that balances the generalizability and specificity of the resultant XAI subjects. Using this method, we can find the most prevalent topics and patterns in the Stack Exchange posts on XAI.

Step 6: Identify the sampling size for each XAI topic. The original dataset had 999 entries in 10 topics. Using Cochran's formula (Woolson et al. 1986), the proper sample size was 278 with a Margin of error of 0.05, a confidence level of 0.95, and an estimated population percentage character of 0.5. Then, using Neyman's technique, samples were taken from each category so that the total sample size calculated using Cochran's formula was equal to the sum of the samples (Neyman 1938; Kubiak and Kawalec 2022). By employing this approach, the sample size for each category varied depending on the proportion of each group within the population. The details of both total and sample sizes for each identified topic are presented in Table 2.

3.2 Research Questions

In this section, we present our Research Questions. For each RQ, we describe the motivation behind it and the approach we took to answer it.

3.2.1 RQ1: What Topics are XAI Developers Asking About?

Motivation While the literature on XAI has primarily focused on surveys, comprehension, and methods for selecting interpretable models, it is essential to address the practical questions and concerns faced by developers and users of AI systems. By delving into the discussions, inquiries, and queries posed by XAI developers on Stack Exchange, we can uncover valuable insights into the topics and issues at the forefront of their minds during the

Table 2 The total and sample populations of each topic

Topic	Sample Size	Total Size
Topic #1	54	194
Topic #2	42	150
Topic #3	40	143
Topic #4	28	102
Topic #5	28	100
Topic #6	20	71
Topic #7	20	71
Topic #8	19	67
Topic #9	18	64
Topic #10	9	37

development and implementation of XAI systems. By understanding these concerns, we can identify common pain points, address gaps in knowledge, and cultivate a better understanding of the needs of XAI developers.

Approach To address the topics XAI developers are asking about, we adopted a mixed-method research strategy, employing both quantitative and qualitative analysis methods. Our first approach was to utilise LDA to identify predominant topics within the discourse. In line with prior research on labelling topics (Li et al. 2021; Ahmed and Bagherzadeh 2018; Bagherzadeh and Khatchadourian 2019; Alamin et al. 2023), we use a card sorting approach (Fincher and Tenenberg 2005). In this process, the first two authors—both holding master's degrees in software engineering and data science, with over four years of professional experience in software engineering, artificial intelligence, and XAI technologies—individually evaluated the 15-20 most common questions and the top 10 keywords within each identified topic. The third author, a full professor of software engineering with recent publications on fairness and transparency in AI systems, reviewed each stage of the work conducted by the first two authors, ensuring the accuracy and consistency of the evaluation process. Following this initial review, the authors engaged in a collaborative discussion. This led to an initial naming for each topic, with a high degree of consistency, evident from a Cohen's Kappa value (McHugh 2012) of 0.80. Ultimately, a collaborative discussion was undertaken by all authors to address any discrepancies and reach a consensus regarding the selection of definitive names for the 10 high-level topics. This led to an initial categorisation of high-level topics based on the overarching themes of the posts.

Yet, recognising the abstract nature of the topics and to delve deeper and detect more detailed topics, these high-level topics were kept separated and none of them were merged to avoid the inadvertent omission of any sub-topics.

In the succeeding step, as detailed in Section 3.1, we employed Nieman's algorithm as a sampling technique. This allowed us further to dissect each of the ten previously identified topics and pinpoint the sub-topics. The responsibility for analysing the sampled volume linked to each topic (Table 2) fell to the first two authors. They organised their proposals for each topic's sub-topics based on the names designated in the preceding step. They each submitted their proposals and entered into a dialogue until an understanding was reached. Following approximately ten rounds of virtual discussions, facilitated through Skype, the team finally agreed upon uncovering and naming 40 low-level topics (sub-topics).

Eventually, upon identifying and categorising topics and subtopics, it becomes apparent that certain topics addressing related aspects can be grouped into a more overarching category. This helps in a deeper understanding of the discussions and facilitates a succinct representation of the identified patterns and threads of conversation among the XAI developers.

3.2.2 RQ2: What Kinds of Questions are XAI Developers Asking About?

Motivation Once we have identified the most common and pressing XAI topics, the next step is to examine the types of questions XAI developers ask on technical Q&A websites. Previous studies (Rosen and Shihab 2016; Abdellatif et al. 2020; Mahmood et al. 2023; Alamin et al. 2023) have shown that developers ask different types of questions (e.g., how, why, what). Analysing the types of questions XAI developers ask can help us understand

the nature of the challenges encountered during XAI development. This can provide insights into the knowledge gaps and areas of difficulty faced by XAI developers and guide the development of more effective tools and resources for XAI development.

Approach To determine the types of questions being asked by developers on Stack Exchange, we adopt the approach used by previous works (Alamin et al. 2023; Abdellatif et al. 2020; Uddin et al. 2021) and use the labels "How," "Why," "What," and "Other". We use a statistically significant sample generated in Section 3.1 and used in the RQ1.

- *How.* is used for posts asking for methods or techniques to implement something. The questions primarily centre on the procedures necessary to address specific problems or accomplish particular undertakings e.g., How do I interpret GRAD-CAM's feature attribution to time series zero-padding in a CNN classifier? (Q121621).
- *Why.* is used for posts seeking reasons, causes, or purposes for unexpected behaviour. For example, in (Q54305070) the user asks why the LIME explainer shows prediction probabilities different from the classifier prediction.
- *What.* posts falling under this type are those where developers are looking for specific information to make informed decisions e.g., What exactly does gradient-based saliency map tell us? (Q32625).
- *Other.* is assigned to posts that do not fit the above categories e.g., Who is working on explaining the knowledge encoded into machine learning models? (Q16214).

Building on the definitions of Epistemic Curiosity (EC) (Lauriola et al. (2015); Litman (2019); Lievens et al. (2022)), we expanded our analysis to include its two primary dimensions: Interest-type (I-type) and Deprivation-type (D-type). These dimensions provide crucial insights into the cognitive motivations underlying the different types of questions developers pose, particularly in the context of XAI.

- *I-type EC.* reflects an individual's anticipation of discovering something intriguing, engaging, or aesthetically pleasing. Questions aligned with this dimension often exhibit open-ended curiosity, driven by a desire to explore concepts or broaden understanding without an immediate need for concrete answers (e.g., How do language models know what they don't know—and report it? (Q40874)).
- *D-type EC.* In contrast, arises from a sense of informational deprivation. This dimension is characterised by a need to address specific uncertainties or fill knowledge gaps. Questions in this category typically focus on finding practical solutions or obtaining precise explanations (e.g., How can SHAP be used with SpaCy models? (Q68545128)).

By applying labels to the samples we identified in Section 3.2.1, we can understand the kind of inquiries made by XAI developers. The initial pair of authors independently scrutinised the 276 sample posts, labelling each with one of these types. Following this, we measured their concordance levels using Cohen's Kappa and discovered a substantial level of concordance ($k = 0.80$) that aligned with previous research (Alamin et al. 2023; Uddin et al. 2021). Certain questions led to slight divergences among the authors, but these were eventually reconciled through thorough debate and multiple reviews until a mutual agreement was

established. We have made the comprehensive data from this procedure accessible online in the replication package we have published.

3.2.3 RQ3: What Topics Exhibit Higher Levels of Popularity and Difficulty?

Motivation Having identified the most common and pressing XAI topics and the types of questions XAI developers ask, the next step is to investigate the difficulty and the popularity level of questions in each topic. Understanding which topics are more popular and challenging to address can help identify areas for more research and development. Additionally, this analysis can help us pinpoint the topics that require more support and resources for XAI developers to overcome their challenges.

Approach We investigated the most popular XAI topics among developers using three different popularity metrics adopted in prior research (Yang et al. 2016). The first metric that we use is average view count; previous studies (Rosen and Shihab 2016; Uddin et al. 2021; Alamin et al. 2023; Abdellatif et al. 2020) have shown that the more frequently a post is visualised, the more relevant and significant it is to developers, and we are using the ViewCount column for calculating it. Another metric is the average score, which measures the overall quality of the post in terms of helpfulness, accuracy, and relevance to the community. A higher score indicates that the community considers the post valuable and can be accessible with the Score column in the dataset. The last metric we use for calculating popularity is the average comment count, which can reflect the level of engagement and interaction among developers regarding the discussed topics (Sengupta and Haythornthwaite 2020; Yang et al. 2016), and we calculate it with the CommentCount column in the dataset. Additionally, to accurately evaluate the difficulty of topics within XAI-related posts, we employ two metrics that have been previously utilised in research (Rosen and Shihab 2016; Barua et al. 2014; Abdellatif et al. 2020). The first metric is the percentage of posts within a topic that remain without accepted answers, represented as the "unaccepted answers percentage." Within a post, the author can mark an answer as accepted if it properly addresses and solves the original question. Therefore, topics with a high percentage of posts without accepted answers can be considered more challenging (Rosen and Shihab 2016; Bagherzadeh and Khatchadourian 2019). However, it is important to note that the quality of a question can also impact the ability of developers to provide satisfactory answers. The Stack Exchange community works together to edit questions to improve their quality and, in turn, the quality of the answers provided. Thus, the lack of accepted answers may also indicate that other developers are finding it difficult to suggest an answer due to the poor quality of the question. The second metric we utilise is the median time it takes for a post to receive an accepted answer, referred to as "Median Time to Answer (Hrs.)." This metric calculates the median number of hours it takes for a post to receive an accepted answer, considering the time the accepted answer was created rather than the time it was marked as accepted. The longer it takes for a post to receive an accepted answer, the more difficult the post is considered to be Rosen and Shihab (2016); Bagherzadeh and Khatchadourian (2019); Abdellatif et al. (2020).

Evaluating the popularity and difficulty in obtaining an accepted answer for various topics through multiple metrics can be complex. Therefore, we calculate two integrated metrics

based on previous research (Uddin et al. 2021; Alamin et al. 2023). The following outlines the details of the two combined metrics.

Fused Popularity Metrics First, we calculate popularity metrics separately for each of the 10 XAI topics and 40 XAI sub-topics. Following the approach outlined in prior research (Uddin et al. 2021; Alamin et al. 2023), we standardise these metrics by dividing their values by the average metric values across all groups—10 groups for topics ($K = 10$) and 40 groups for sub-topics ($K = 40$). This results in creating three novel normalised popularity metrics for each topic denoted as $ViewN_i$, $CommentCountN_i$, and $ScoreN_i$ (e.g., for XAI topics $K = 10$). In the final step, we ascertain the fused popularity, represented as $FusedP_i$ for a given group *i* by calculating the average of the three normalised metric values.

$$ViewN_i = \frac{View_i}{\frac{\Sigma_{j=1}^{K} View_j}{K}} \tag{3}$$

$$ScoreN_i = \frac{Score_i}{\frac{\Sigma_{j=1}^{K} Score_j}{K}} \tag{4}$$

$$CommentCountN_i = \frac{CommentCount_i}{\frac{\Sigma_{j=1}^{K} CommentCount_j}{K}} \tag{5}$$

$$FusedP_i = \frac{ViewN_i + ScoreN_i + CommentCountN_i}{3} \tag{6}$$

Fused Difficulty Metrics Like the approach used for popularity metrics, we compute difficulty metrics for each topic based on the percentage of without-accepted answer questions and the median time required to receive an accepted answer. Subsequently, we standardise these metric values by dividing them by the average metric value across all groups, with 10 groups for XAI topics and 40 for XAI sub-topics. Ultimately, the fused difficulty metric, represented as *FusedDi* for topic *i*, is determined by averaging the values of the $WoAccAnsPCTN_i$ and $MedHrsToGetAccAnsN_i$ metrics.

$$WoAccAnsPCTN_i = \frac{WoAccAnsPCT_i}{\frac{\Sigma_{j=1}^{K} WoAccAnsPCT_j}{K}} \tag{7}$$

$$MedHrsToGetAccAnsN_i = \frac{MedHrsToGetAccAns_i}{\frac{\Sigma_{j=1}^{K} MedHrsToGetAccAns_j}{K}} \tag{8}$$

$$FusedD_i = \frac{WoAccAnsPCTN_i + MedHrsToGetAccAnsN_i}{2} \tag{9}$$

Furthermore, we analyse the correlation between three metrics related to topic popularity and two metrics associated with difficulty. For this examination, we utilise Kendall's τ cor-

relation measure (Kendall 1938). Notably, Kendall's τ stands out from the Mann-Whitney correlation (Kruskal 1957) by being resistant to the impact of outliers in the dataset. It is worth noting that tracking the temporal evolution of popularity and difficulty metrics is not feasible since the Stack Overflow dataset lacks essential temporal information.

3.2.4 RQ4: Which Technologies Are XAI Developers Using?

Motivation As the field of XAI continues to grow, it is becoming increasingly important for developers to stay up to date on the latest technologies being used in this area and their issues (Treude et al. 2011). Understanding which programming languages and tools are utilised and their challenges can provide valuable insights into the best practices and approaches for developing transparent and interpretable AI systems. By knowing which technologies are being used in question-and-answer platforms for XAI, developers can make informed decisions about which tools and models to use in their projects. This can help them ensure their systems are accurate, efficient, transparent, and explainable to end users. It can also lead to the development of better and more efficient tools. With the latest XAI technologies, it is essential for any developer looking to succeed in this exciting and rapidly evolving field.

Approach As previously mentioned, each question within the Stack Exchange forums is associated with tags representing the high-level categories of the questions. Each question must be accompanied by at least one tag and can have a maximum of five tags. Previous research has demonstrated that analysing these tags can offer additional context to the field, shedding light on the specific topics and related categories discussed (Alfayez et al. 2023). In this section, our initial step involves determining the frequency of each tag's usage in the posts of the XAI dataset within each forum. By identifying tags used more than 10 times in the posts, we focus on assessing the dataset to understand the inclusion of software packages and programming languages participants utilise. Additionally, we analyse the popularity and difficulty of each, applying the same method as Section 3.2.3 while examining the distribution of categories. Furthermore, we assess the outcomes of RQ1 within each tool utilised by the participants, aiming to gain insights into the popularity, difficulty, and distribution of the issues discussed within them. This approach can provide an in-depth understanding of the challenges participants face when using these tools and lead to the enhancement of existing tools or the development of improved ones.

Finally, to provide a more detailed analysis of how these tools are used in different contexts or projects, we obtained a representative sample of questions related to each XAI package using the sampling method described in Step 6 of Section 3.1. The first two authors independently categorised these questions into six distinct usage contexts: Computer Vision, Data Preprocessing, General Machine Learning Tasks (e.g., classification and regression), Natural Language Processing (NLP), Time Series Analysis, and Unspecified. This categorisation was based on a thorough review of each question's content, ensuring accurate assignment to the most relevant usage category. To further ensure the reliability of the categorisation, the third author reviewed and double-checked the results, resolving any discrepancies through discussion with the first two authors.

3.2.5 RQ5: How do the XAI Topics Evolve?

Motivation XAI involves diverse topics, each having unique characteristics. The requirements of these categories change over time, as do the associated themes. We analyse these changes to document and support the expanding XAI community, identifying areas that still require attention.

Approach Drawing inspiration from previous studies (Uddin et al. 2021; Han et al. 2020; Alamin et al. 2023), we investigate the individual and comparative influences of each of the six identified XAI topic categories using a systematic approach.

Topic Absolute Impact For our collection of documents (c), we use LDA to generate K distinct topics, which we will refer to as $t_1, t_2, ..., t_K$. We then determine each theme's influence during a specific month (m) using an "absolute impact" measurement. This measure reflects the significance of a given topic within that particular month. $D(m)$ represents the total number of posts in the month m, and $\theta(p_i; t_K)$ signifies the probability of a post (p_i) belonging to a topic t_K.

$$Impact_{absolute}(t_K; m) = \sum_{p_i=1}^{D(m)} \theta(p_i; t_K), \quad 1 \leq i \leq c \tag{10}$$

After conducting our topic modelling, we recognised 10 topics organised into six primary topic categories. To enhance the equation for measuring the absolute impact of XAI topics, we calculate the absolute impact metrics for a specific category (C_j) within a designated month (m) using the following approach:

$$Impact_{absolute}(C_j; m) = \sum_{t_K}^{C_j} Impact_{absolute}(t_K; m), \quad 0 < j < C \tag{11}$$

Topic Relative Impact The impact metric associated with a specific XAI topic (t_K) indicates the proportion of posts related to that topic relative to all the posts within our corpus (c) for a particular month (m). In line with prior studies (Han et al. 2020; Croft et al. 2022; Uddin et al. 2021; Alamin et al. 2023), we calculate these associated impact metrics for XAI topics. The total number of posts assigned to a topic t_K in a given month m is denoted by $D(m)$. In this context, $\theta(p_i; t_K)$ signifies the probability of a particular post (p_i) in our corpus (c) being associated with a specific topic t_K. The computation of the metric for a topic t_K in a month m is carried out using the following equation:

$$Impact_{relative}(t_K; m) = \frac{1}{|D(m)|} \sum_{p_i=1}^{D(m)} \theta(p_i; t_K), \quad 1 \leq i \leq c \tag{12}$$

Ultimately, the equation for assessing the impact of XAI topic categories has been defined as follows (C_j represents one of our six categories, including the topics associated with each respective category):

$$Impact_{relative}(C_j; m) = \sum_{t_K}^{C_j} Impact_{relative}(t_K; m), \quad 0 < j < C \tag{13}$$

4 Results

This section presents the findings of our study, organised around the five Research Questions introduced in Section 3.2. For each RQ, we summarise the results of the empirical analyses described in Section 3.

4.1 What Topics are XAI Developers Asking About? (RQ1)

Through the LDA results, 10 topics were identified, and after labelling, they were grouped into 40 subtopics and 6 categories. High-level categories are Troubleshooting, Feature Interpretation, Model Analysis, Data Management, Visualisation, Concepts and Applications. Figure 2 presents these categories' distribution of questions, topics, and subtopics. Troubleshooting contains the highest percentage of questions (38.14%), the highest number of topics (3 topics), and the highest number of subtopics (11 subtopics), followed by Feature Interpretation (20.22% of questions, 2 topics, and 6 subtopics) and Visualisation (14.31% questions, 1 topic, and 6 subtopics).

Figure 3 displays a taxonomy comprising 10 topics, which have been organised into 6 distinct categories. You will also find related subtopics associated with each topic in the figure. We have included the percentage of posts for each category, topic, and subtopic, and these values have been arranged in descending order based on their respective percentages. In the following sections, we will provide in-depth discussions on each of the 6 categories, 10 topics, and 40 subtopics.

Troubleshooting This category emerges as a prominent theme, accounting for 3 out of 10 topics (38.14%). Within this category, users exchange ideas and solutions for various chal-

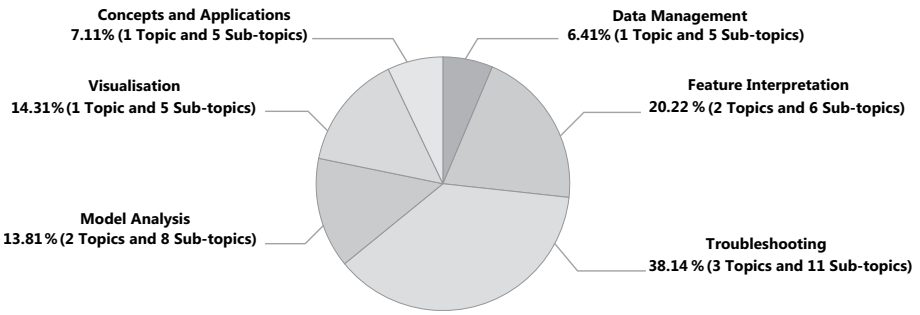

Fig. 2 Distribution of XAI questions, topics, and sub-topics per category

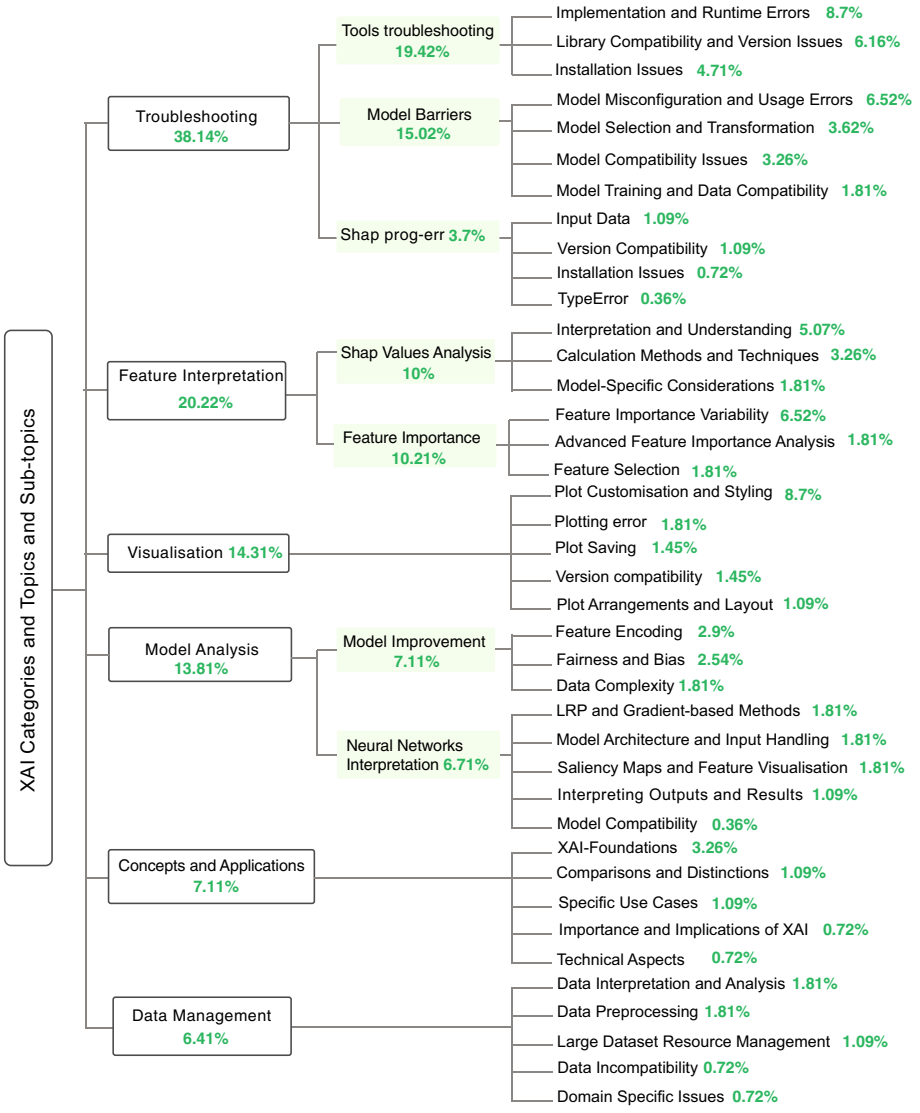

Fig. 3 Taxonomy of XAI discussions

lenges and obstacles while utilising various XAI tools and models. Three distinct subtopics fall under the umbrella of Troubleshooting: Tools Troubleshooting addresses issues related to popular XAI libraries and tools such as LIME, AIF360, and DALEX; Model Barriers sheds light on the complex hurdles and errors that arise when working with various machine learning models, and because SHAP (SHapley Additive exPlanations) library has such a large community, one of the topics in this category is dedicated to SHAP prog-err, which focuses on troubleshooting errors and complications that occur specifically when using the SHAP library.

Tools troubleshooting this topic is the largest topic in our dataset (19.42%) that comprises three sub-topics. The first sub-topic, Implementation and Runtime Errors, is one of the two subtopics with the highest number of questions within our dataset, accounting for 8.7%. It encompasses discussions focused on resolving challenges encountered while implementing and developing various XAI tools. These errors can arise for various reasons, such as Permission Hurdles (e.g., Q66807617), Memory allocation errors (e.g., Q71185663), and Type Errors (e.g., Q67197907). The second largest sub-topic, Library Compatibility and Version Issues, makes up 6.16% of our dataset and deals with problems arising from the incompatibility between two or more libraries, such as the compatibility issues between Yellowbrick and Sklearn (e.g., Q67428836). The third sub-topic, Installation Issues (4.71%), addresses common obstacles and problems encountered during the installation process, including environment issues (e.g., Q56942378) and missing build tools (e.g., Q61083193).

Model Barriers is another significant focal point within our dataset, constituting a substantial portion of the discussions at 15.02%. This topic encompasses multifaceted challenges, errors, and intricacies that users encounter while selecting and using different machine learning and XAI models. Within this topic, four sub-topics dive deeper into specific aspects of these challenges. Model misconfigurations and usage errors (6.52%) can lead to unexpected or inaccurate outcomes. This sub-topic addresses these issues, delving into discussions on how to configure models correctly, handle Usage Errors (e.g., Q75248343), and troubleshoot model misconfigurations (e.g., Q70443299). The Model Selection and Transformation errors (3.62%) sub-topic further explores the nuances and technical issues of effectively selecting and transforming models for improved performance and interpretability, such as using SHAP or LIME or any other model interpretability tools for TPOT classifiers (Q67394585) and converting the PyCaret model to LGBMClassifier for SHAP compatibility (Q76623054). Model Compatibility Issues (3.26%) revolve around the compatibility challenges that arise when integrating and deploying machine learning models. Users engage in discussions to troubleshoot issues stemming from model incompatibilities with existing software, infrastructure, or other models such as incompatibility between the R version of LIME and XGBoost models with count Poisson objective function (Q49280345). Model Training and Data Compatibility (1.81%) is another sub-topic that ensures machine learning models are trained effectively and are compatible with input data is pivotal for XAI. Discussions in this sub-topic revolve around challenges such as the Absence of a labelled dataset for LIME explainer training (Q92593).

SHAP prog-err We identified three subjects that accounted for 3.7% of the total inquiries in our dataset: 1. Input Data (1.09%), 2. Version Compatibility (1.09%), 3. Installation Challenges (0.72%), and 4. TypeError (0.36%).

Feature Interpretation In the realm of Feature Interpretation, our analysis reveals a diverse landscape of topics, collectively encompassing 20.22% of the total dataset. This category contains two topics: 1. Feature Importance (10.21%) and 2. SHAP Values Analysis (10%).

Feature Importance Feature Importance is a topic under the Feature Interpretation category. It can be divided into sub-topics related to feature selection, Feature Importance Variability, and Advanced Feature Importance Challenges. The sub-topic of Feature Selection, which accounts for 1.81% of the questions, focuses on selecting important and relevant features

in a dataset such as dividing features into groups of high correlation (Q64424669). The sub-topic of Feature Importance Variability, which covers 6.52% of the questions, revolves around the Interpreting and Measuring Feature Importance in Various Models and Conditions like Interpretation of complex linear regression effects graph with variable details (Q110683) and Discrepancy between XGBoost feature importance and SHAP output for a specific variable (Q72626523). Finally, the Advanced Feature Importance Analysis sub-topic (1.81%) delves into the more intricate aspects of evaluating and comprehending the contributions of individual features in predictive models. This includes tasks like calculating feature contributions in anomaly detection (Q113982) and handling missing values (Q52382857).

SHAP Values Analysis SHAP Value Analysis is another area in the domain of Feature Interpretation. It encompasses subheadings including Calculation Methods and Techniques, Interpretation and Understanding, and Model-Specific Considerations. The Calculation Methods and Techniques sub-topic (3.26%) provides an understanding of the methodologies and procedures to compute SHAP values for thorough feature evaluation in datasets; for example, How is the "base value" of SHAP values calculated? (Q73553) or Clarification sought regarding Shapley value definition in linear models (Q118086). The Interpretation and Understanding sub-topic, having 5.07% of relative queries, turns around the comprehension and analysis of SHAP values across Various Models and Conditions such as Interpreting Shapley values and their implications in XGBoost (Q116227) and Understanding SHAP values for binary classification (Q75090510). The Model-Specific Considerations subcategory (1.81%) explores the nuances and unique challenges when applying SHAP values to different machine learning models, providing solutions related to feature interpretation, such as obtaining SHAP values in native units after log-transformed regression (Q74027051).

Model Analysis The Model Analysis category has a multifaceted set of topics, amounting to 13.81% of the entire dataset. Two topics within this domain are Model Improvement (7.11%) and Neural Networks Interpretation (6.71%).

Model Improvement It is a cardinal aspect of this category. It resides in three sub-topics, one of which is Fairness and Bias (2.54%), This subject looks into the strategies to ensure equitable model treatment across diverse data categories, such as Understanding the necessity of specifying privileged/unprivileged attribute values in fairness metrics (Q65327235) or Understanding faithfulness score's role in model interpretability (Q111076). The Feature Encoding (2.9%) sub-topic concerns methods for encoding features in different models like choosing between dummy variables and one-hot encoding for categorical variables in models, considering explainability (Q26747). Lastly, Data Complexity (1.81%) deals with complex data structures and challenges related to data understanding and customisation that can improve the model behaviour, like Clustering mixed features with explanatory insights for data structure understanding. (Q71923250).

Neural Networks Interpretation Neural Networks Interpretation discusses the workings of neural networks, and this topic includes five sub-topics. Interpreting Outputs and Results (1.09%), which addresses the interpretation of neural network outputs, for example, which

addresses the interpretation of neural network outputs, for example, interpreting the trained CNN model output with SHAP (Q117064). LRP and Gradient-based Methods (1.81%) focus on issues with interpretability techniques based on Layer Relevance Propagation or gradient information, such as GRAD-CAM gives unwanted attribution to zero-padding in MTS classification (Q121621). The Model Architecture and Input Handling (1.81%) scrutinises the intricacies and issues of explainability in neural architectures and how inputs are catered to, such as explainability methods for autoencoders applied to dissimilar image pairs (Q34844). Model Compatibility (0.36%) reviews the issues related to compatibility between various model types, for example, version incompatibility between SHAP and Tensorflow (Q61864910). Lastly, Saliency Maps and Feature Visualisation (1.81%) discuss visual interpretation methods for neural networks, such as interpreting gradients in saliency maps and their significance (Q32625).

Data Management Data Management (6.41%) addresses data handling and analysis, with five sub-topics. Data Incompatibility (0.72%) deals with managing inconsistent data formats, including issues like SHAP value calculation errors in sparse binary data (Q72721427). Data Interpretation and Analysis (1.81%) represents the methods and strategies for effectively interpreting data for analytics. Data Preprocessing (1.81%) details methodologies for preparing data to improve model interpretability, emphasising topics like Verifying if input standardisation is necessary for the SHAP background dataset (Q72434347) or Utilising SHAP values as thresholds to remove outliers from the dataset (Q69225478). Domain Specific Issues (0.72%) explore the challenges unique to specific data domains, such as how language models know what they don't know (Q40874). Large Dataset Resource Management (1.09%) provides insights into efficiently handling and managing extensive data collections. It sheds light on specific topics equivalent to slow LIME interpretation due to large dataset resource constraints (Q72001683).

Visualisation The visualisation (14.31%) encompasses a critical area within our dataset, consisting of five sub-topics. These sub-topics delve into aspects related to visualising data and models. First, Plot Arrangements and Layout (1.09%) specifically covers the organisation and configuration of plots on a visual canvas to provide a coherent visualisation, addressing potential concerns like Modify code to display two plots side by side in one row (Q63605976). Plot Customisation and Styling (8.7%) is another sub-topic with the highest number of questions within our dataset that elucidates techniques for individualising and enhancing the aesthetic of plots. It encompasses subjects like Increasing Yellowbrick ROCAUC plot font size for better visibility (Q68020143) or Changing SHAP plot colours in dark theme for better visibility (Q60238873). Plot Saving (1.45%) delves into the issues in the process of storing generated plots such as the trouble of saving the SHAP summary plot as an image (Q57976902). Plotting Error (1.81%) encapsulates challenges faced during the creation of plots, such as the SHAP multioutput decision plot does not plot predicted values correctly (Q75945841). Lastly, Version Compatibility (1.45%) encapsulates issues associated with the alignment of different versions of visualisation libraries. This branch touches on questions like the Matplotlib font issue after installing and importing the Yellowbrick module (Q75034227).

Concepts and Applications Concepts and Applications (7.11%) encompasses five sub-topics. These sub-topics delve into various dimensions of XAI. Comparisons and Distinctions (1.09%) provide comprehensive insights into various XAI definitions' differences and similarities. It covers concerns such as Distinguishing between explainable and interpretable ML definitions and concepts (Q99827). Importance and Implications of XAI (0.72%) discusses the role and significance of XAI in modern applications. It addresses questions like "Why do we need explainable AI? (Q14224)". Specific Use Cases (1.09%) target the application of explainable AI in various particular contexts such as Seeking foundational and recent research on Explainability in LLMs (Q40519) or Human-centric reasons for the prevalent use of tree-based models in medical diagnosis (Q3777). Technical Aspects (0.72%) investigates the technicalities and practicalities of implementing XAI methodologies. This encompasses elements such as "Why do people favour neural networks over decision trees in visual question answering? (Q6176)". The largest segment, XAI-Foundations (3.26%), discusses XAI's foundational principles, concepts, and theories. The emphasis is given to basic questions like, Is explainable AI more feasible through symbolic AI or soft computing? (Q12949). As stated in Section 2.2, many previous studies have tried to identify the challenges faced by developers and researchers in the field of XAI through surveys (Saeed and Omlin 2023; Yang et al. 2023; Ding et al. 2022; Das and Rad 2020). For instance, previous research in data management has highlighted the long time needed to train ML models, especially Deep Learning models. This extended time frame usually ranges from hours to days due to the large datasets used for model training (Choo and Liu 2018). This prolonged process presents a major challenge, made worse by additional problems faced by users dealing with large datasets, such as Data Incompatibility and Data Preprocessing. Furthermore, many studies on visualisation have stressed the crucial role of visualisation in XAI (Choo and Liu 2018; Samek et al. 2017; Chatzimparmpas et al. 2020; Saeed and Omlin 2023; Zhang and Zhu 2018), alongside the development of diverse visualisation techniques (Jiang et al. 2019; Choo and Liu 2018). Our study confirms the importance of visualisation but also shows that real-world users face difficulties in different aspects of using visualisation tools, like customising plots and Version Compatibility issues. Furthermore, prior research on model analysis (Zhang and Zhu 2018; Hall et al. 2019; Chatzimparmpas et al. 2020) has identified several challenges, including improving model debugging techniques, model innovation, and the utilisation of interpretation and explanation capabilities for models and architecture comparison.

Our findings support these observations and reveal additional issues encountered by users. These include the need to increase fairness, dealing with complex datasets when creating models, and addressing compatibility problems among different models. Lastly, while certain prior studies (Yang et al. 2023; Choo and Liu 2018) have outlined challenges within XAI tools, none have systematically categorised the troubleshooting challenges inherent in XAI.

Finding 1: XAI discussions within Q&A forums encompass a diverse spectrum of 10 primary topics, divided into 40 sub-topics and six categories: Troubleshooting, Feature Interpretation, Visualisation, Model Analysis, XAI Concepts and Applications, and Data Management. Troubleshooting stands out as the most discussed category. Tools Troubleshooting, Model Barriers, and Visualisation are the key topics respectively, and the largest sub-topics are Plot Customisation and Styling, Tools Implementation and Runtime Errors, and Model Miscon guration and Usage Errors, respectively.

4.2 What Kinds Of Questions are XAI Developers Asking About? (RQ2)

Table 3 displays the proportions of various question types within the six overarching XAI categories.

4.2.1 Question Types Distribution

How (60.1%) This question type takes centre stage, representing the predominant theme in XAI discussions. It reflects a keen interest in the practical aspects of implementing and applying XAI techniques. Like this question that has been asked philosophically, how do large language models know what they don't know? (Q40874). Another question related to using a specific model in the LIME explainability package (Q59995744).

Why (21.4%) These questions form a substantial portion of the discourse about AI behaviours' underlying motivations and rationale in the XAI context. Technical discussions are also raised with this type of question, such as the question asked about SHAP values (Q75638742). In some instances, simpler and fundamental questions arise. For instance, a common inquiry might involve the necessity of XAI, asking why it is needed in the first place (Q66524).

What (16.3%) These questions, while less prevalent, demonstrate an ongoing quest to comprehend XAI's fundamental principles and real-world applications, such as the difference between interpretation and explanation in machine learning (Q70164). In another instance, the question concerns discerning the disparities between Interpretable and Transparent Machine Learning (ML) algorithms (Q99827).

Table 3 Topics question types

Topic	how	what	why	others
Data Management	66.7	22.2	5.6	5.6
Feature Interpretation	57.1	12.5	26.8	3.6
Model Analysis	61.5	23.1	12.8	2.6
Troubleshooting	64.8	15.2	20.0	0.0
Visualisation	67.5	0.0	32.5	0.0
XAI Concepts and Applications	25.0	45.0	20.0	10.0
Overall	60.1	16.3	21.4	2.2

Others (2.2%) The Others category, though relatively minor in volume, serves as a space for diverse, niche, and unconventional inquiries, showcasing the broad spectrum of XAI interests. Some questions of this type explore scenarios to understand if certain conditions exist. For example, one inquires if predictive features with zero SHAP values are used in models (Q109007). Another question is assessing the feasibility of methods, such as using feature importance from classification for clustering (Q90714). Additionally, academic questions seek recommendations for relevant research papers and even look for researchers in this field (Q16214).

4.2.2 Question Types in XAI Categories

Data Management In this category, "How" questions dominate (66.67%), highlighting a strong focus on the practical implementation of XAI in data-related tasks.(Q69258055) Additionally, "What" questions (22.22%) emphasise the importance of understanding XAI's role in data management. (Q76852, Q73279201)

Feature Interpretation Here, "How" (57.14%) and "Why" (26.79%) questions coexist prominently. This suggests a dual emphasis on practical implementation and understanding the rationale behind feature interpretation in XAI. (Q115149, Q108913)

Model Analysis In the realm of Model Analysis, "How" questions (61.54%) prevail, showcasing an eagerness to delve into the inner workings of AI models. (Q121621) "What" questions (23.08%) also maintain significance. (Q32625)

Troubleshooting This category exhibits a balanced distribution of "How" (63.73%) and "Why" (20.59%) questions. Troubleshooting in XAI encompasses practicality and understanding. (Q61401627, Q121755)

Visualisation With a strong emphasis on "How" questions (67.5%) and "Why" questions (32.5%), Visualisation appears to revolve around the practical implementation of visual aspects in XAI. (Q74731157, Q69986745)

XAI Concepts and Applications "What" questions (45%) take the lead here, underlining the importance of understanding XAI concepts and their real-world applications. (Q70164) "Why" questions (20%) are also significant, reflecting an interest in the rationale behind XAI in diverse contexts (Q3777).

Table 4 presents the distribution of Epistemic Curiosity (EC) types, including Deprivation-type (D-type) and Interest-type (I-type), across the question categories: "How," "What," "Why," and "Others." The results indicate that D-type EC, which focuses on resolving specific uncertainties and filling knowledge gaps, dominates the dataset, accounting for 95.32% of all questions, while I-type EC represents only 4.68%. Among the question types, 96.43% of "How" questions Align with D-type EC, reflecting a strong emphasis

Table 4 EC type of questions

EC type	how	others	what	why	Overall
D-type	96.43%	66.67%	88.89%	100.00%	95.32%
I-type	3.57%	33.33%	11.11%	0.00%	4.68%

on practical problem-solving and the implementation of methods or techniques. Similarly, 88.89% of "What" questions and 100% of "Why" questions are categorised as D-type EC, highlighting the developers' goal-oriented approach to seeking concrete information and understanding underlying mechanisms. Overall, the analysis underscores that XAI developers predominantly ask questions driven by D-type EC, focusing on addressing practical challenges and filling knowledge gaps.

Finding 2: "How" questions take the lead, particularly prevalent in Visualisation (67.5%) and Data Management (66.67%), indicating a strong emphasis on practical implementation. "What" questions signify a quest for understanding, with higher percentages observed in XAI Concepts and Applications (45%) and Model Analysis (23.08%). Troubleshooting (20.0%) sees a signi cant share of "Why" questions, re ecting a focus on unravelling AI issues, and "Others" questions thrive in XAI Concepts and Applications (10%). The results also show that 95.32% of XAI-related questions align with D-type EC, emphasising practical problem-solving, while only 4.68% re ect I-type EC.

4.3 What Topics Exhibit Higher Levels of Popularity and Difficulty? (RQ3)

Initially, we address the popularity of topics, followed by an examination of their level of challenge. Lastly, we delve into the connection between the popularity and difficulty of topics.

Popularity Table 5 presents data on the popularity of 10 topics, including average statistics for the number of views, scores, and comment counts. Additionally, it includes a composite popularity metric known as FusedP, which is derived from the previously mentioned three metrics.

The topic Concepts and Applications stands out with the top FusedP score at 1.22, the highest average rating of 3.07, and the second-highest comment count at 1.2. This topic typically covers various principles and practices within the realm of Explainable Artificial Intelligence (XAI). Within this topic, the subtopic Importance and Implications of XAI is the most popular, earning the highest score of 37. Following closely is the topic Trouble-

Table 5 Topics popularity

Topic	Category	FusedP	#View	#Score	#CommentCount
Concepts and Applications	Concepts and Applications	1.22	468.96	3.07	1.2
Tools troubleshooting	Troubleshooting	1.18	1543.81	1.18	1.12
Neural Networks Interpretation	Model Analysis	1.16	765.68	2.45	1.11
SHAP prog_err	Troubleshooting	1.11	1183.05	1.54	1.08
Visualisation	Visualisation	1.03	1254.52	1.25	0.94
SHAP Values Analysis	Feature Interpretation	1.02	1127.3	1.47	0.92
Data Management	Data Management	0.89	633.06	0.85	1.47
Model Barriers	Troubleshooting	0.82	860.4	1.15	0.81
Model Improvement	Model Analysis	0.82	850.14	1.2	0.79
Feature Importance	Feature Interpretation	0.75	514.29	1.2	0.96

shooting Tools from the Troubleshooting category, maintaining the second-highest FusedP value at 1.18. It also boasts the highest number of views at 1543.81 and the third-highest number of comments at 1.12. Discussions under this topic revolve around addressing challenges encountered while using different tools and technologies. The subtopic Installation Issues is the most popular, with the highest average views at 3697.85 and the highest average comments at 2.69. This is followed by the subtopic on Implementation and Runtime Errors with an average score of 1.5 and an average comment count of 2.

In third place, with a FusedP score of 1.16, is the subject of interpreting neural networks within the model analysis category. This topic boasts the second-highest average score, averaging at 2.45 per post. Conversations in this domain typically revolve around understanding and interpreting the results and behaviours of neural networks. The subtopic LRP and Gradient-based Methods stand out as the most popular within this subject, with an average comment count of 2.6 and a score of 1. A user's question on implementing Grad-CAM for multi-channel 1D CNN activation maps, as seen in question Q72462089, exemplifies the engagement in this subtopic. Additionally, the subtopic Model Compatibility takes the lead in views, averaging 608. Conversely, the topic "Feature Importance" from the Feature Interpretation category holds the least popularity, scoring the lowest FusedP at 0.75. It also records the lowest average views, with only 514.29 views. Discussions in this area predominantly focus on understanding and assessing the importance of different features in machine learning models. However, its limited appeal can be attributed to the intricate factors influencing feature importance in various model contexts.

Difficulty Table 6 showcases two difficulty metrics for each topic: 1. Percentages of questions without an accepted answer, 2. Median hours taken to obtain an accepted answer, it also has FuseD value per topic based on the previous two metric values. These metrics gauge the challenge level of obtaining a correct answer to a question. The topics are ranked by FuseD value in descending order.

In the category of model analysis, improving models stands out as the most challenging topic, as indicated by the FuseD value of 1.51. A significant 64.79% of questions related to this topic lack an accepted answer, experiencing the lengthiest waiting time of 40.51 hours. Table 3 illustrates that this topic ranks second least popular, encompassing queries about enhancing and adjusting various aspects of predictive models and machine learning algorithms.

Table 6 Topics difficulty

Topic	Category	FusedD	Hrs To Acc.	W/o Acc. Ans
Model Improvement	Model Analysis	1.51	40.51	64.79%
Model Barriers	Troubleshooting	1.26	27.2	79.87%
SHAP Values Analysis	Feature Interpretation	1.22	27.87	71.00%
Visualisation	Visualisation	1.17	28.73	60.84%
XAI Concepts and Applications	XAI Concepts and Applications	1.03	19.99	73.91%
Neural Networks Interpretation	Model Analysis	0.86	10.27	86.36%
Tools troubleshooting	Troubleshooting	0.81	11.49	74.16%
SHAP prog_err	Troubleshooting	0.8	9.66	81.08%
Data Management	Data Management	0.68	8	69.35%
Feature Importance	Feature Interpretation	0.66	4.97	77.55%

Topics within this category explore ways to boost model performance, optimise predictions, and tackle challenges related to data complexity, fairness and bias, as well as methods for refining feature encoding and addressing categorical data challenges. Notably, the Data Complexity subtopic is the most challenging subtopic within this topic, with an average time of 175 hours to receive an accepted answer and 60% of questions remaining without accepted answers. The second challenging topic, Model Barriers, falls under the troubleshooting category. Here, a substantial 79.87% of questions lack accepted answers, and the average response time for accepted answers is 27.2 hours. Questions in this domain primarily revolve around overcoming obstacles and challenges tied to model development and deployment. The subtopic of Model Misconfiguration and Usage Errors takes the lead as the most difficult, with 83.33% of unanswered questions and an average waiting time of 60.3 hours to receive an accepted answer in this topic.

The topic of interpretation of neural networks has the highest number of questions without accepted answers, with a value of 86.36, and Table 5 shows that the popularity of this topic is also significant, as shown by the FusedP value of 1.16. This topic contains questions about deep learning and neural network interpretation methods and tools. It contains posts like Q34526 about designing interpretable neural networks for object recognition in images, with nine comments and two answers without any accepted answer. The subtopic of LRP and Gradient-based Methods, being the most favoured, also proves to be the most challenging within this topic.

The Feature Importance topic from the Feature Interpretation category is the least challenging, as per the FuseD value of 0.66. It is also the least popular topic. This topic experiences a relatively low median response time of 4.97 hours for an accepted answer. Nevertheless, about 77.55% of questions still lack accepted answers, indicating that although responses are quicker, they might not always be satisfactory.

The visualisation topic within our categories stands out in terms of its response statistics. Notably, it has the lowest percentage of unaccepted answers (60.84%), indicating that users frequently find satisfactory and comprehensive solutions with responses under this topic. However, it conversely records the second-largest median response time (28.73). This suggests that while satisfactory answers are often provided, they are not typically immediate and require some time for formulation. Essentially, questions regarding visualisation tend to be answered successfully, but may necessitate a more extended period for expert review or detailed response generation. It underscores discourse's complex yet reliable nature within the Visualisation domain.

Additionally, among all subtopics, the subtopic addressing Library Compatibility and Version Issues has the highest percentage of questions without an accepted answer, at 94.12%.

Generally, topics related to model analysis and troubleshooting seem to be quite challenging within this context. These subjects often lack accepted answers, demonstrating that more precise and tailored support is needed. Even though Feature Importance demonstrates a swift median response time, a high percentage of questions within this topic remain unanswered. This discrepancy might suggest that while quick responses are offered, they might not fully address the posed questions, with none marked as accepted. Therefore, topics that provide swift responses do not necessarily provide satisfactory solutions, indicating a need for more detailed and focused attention when addressing these questions.

Correlation Between Topic Difficulty and Popularity The purpose here is to investigate the presence of any positive or negative association between the percentage of questions without accepted answers, view count, comment count, and scores. Also, the median hours to get accepted answers is compared with these factors. This is done to understand if the popularity and difficulty of topics have a direct or indirect relationship.

Table 7 displays six correlation measures between topic popularity and difficulty. From the table, we can observe varying degrees of positive and negative correlations between the variables considered. However, none of these correlations are statistically significant at the 95% confidence level, indicated by the p-values greater than 0.05. For instance, the most direct measure of difficulty, % w/o acc. answer, shows a slightly negative correlation with View and a slightly positive one with Comment-Count and the Score. This means popular topics (reflected by views and scores) do not necessarily have a lower difficulty level (lower percentage of unanswered questions), and vice versa. Likewise, Median Hrs to acc. answer, another metric for difficulty, also does not reveal any significant correlation with the popularity reflected by views, comment counts, or scores.

Therefore, based on these findings, one cannot assert that the most popular topics are the least difficult to get an accepted answer and vice versa. However, these insights may prove quite valuable for the platform to devise their strategies further. It would be beneficial to ensure that popular topics have easily accessible answers to improve user satisfaction and engagement.

Finding 3: Concepts and Applications is the most popular topic, boasting the highest count of FusedP and Score. Following closely behind are Tools Troubleshooting and Neural Networks Interpretation. In contrast, Feature Importance emerged as the less popular and challenging topic despite having a substantial proportion of unanswered questions. The Visualisation topic has the lowest percentage of unaccepted answers, however, it recorded the second-largest median response time, the time needed for a satisfactory answer, and it is very popular due to having the second-largest average number of views. Topics associated with model analysis and troubleshooting have proven to be challenging, with many of them lacking accepted answers. The subtopics exploring the Importance and Implications of XAI and addressing Installation Issues are quite popular. On the other hand, LRP and Gradient-based Methods present challenges, and Library Compatibility and Version Issues have a signi cant number of unanswered questions.

4.4 Which Technologies Are XAI Developers Using? (RQ4)

In total, 482 distinct tags were utilised in all three forums, with 54% being used only once and 10% being repeated more than 10 times. Figure 4 displays the count of tags that have been used more than 10 times in each forum, arranged in descending order. It is evident from

Table 7 Correlation between the topic popularity and difficulty

Coefficient/p-value	View	Comment-Count	Score
% w/o acc. answer	–0.07/0.86	0.07/0.86	0.20/0.48
Med. Hrs to acc. answer	0.29/0.29	–0.38/0.16	0.02/1.00

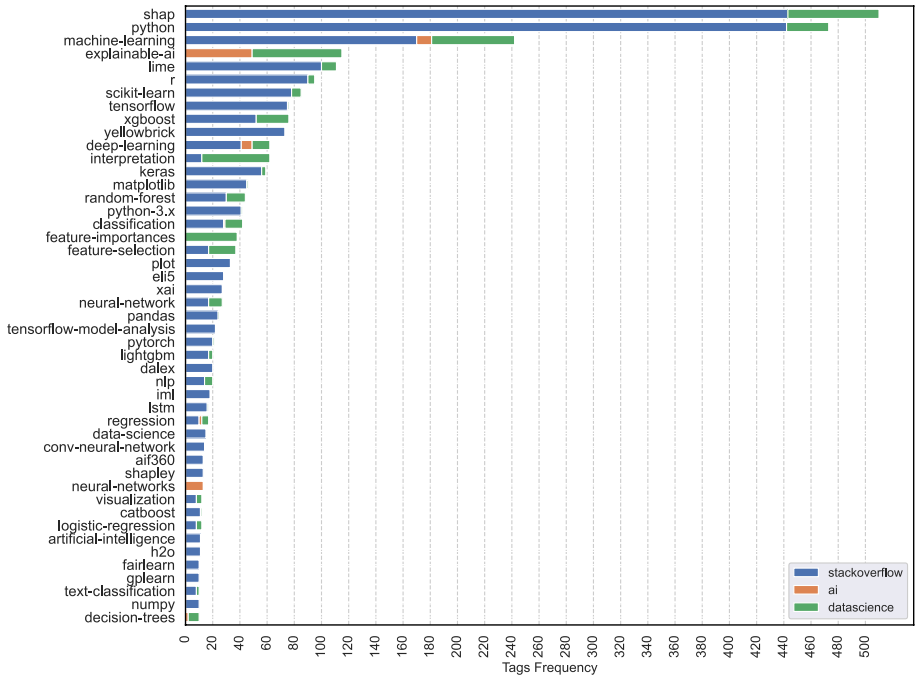

Fig. 4 Distribution of most used tags

this graph that five packages - SHAP, LIME, Yellowbrick, ELI5, AIF360 - and two programming languages, namely Python and R, were the most sought-after XAI tools. Additionally, the artificial intelligence tools Scikit-learn, Tensorflow, Xgboost, and Keras were the most commonly used in XAI questions. Moreover, Matplotlib, a widely popular tool in the field of visualisation, was among the 15 popular tags, indicating substantial usage and interest within the XAI community.

Further analysis shows that 56% of the questions contain tags related to programming languages. Figure 5 demonstrates the distribution of programming languages among questions containing programming languages. Python dominates the XAI developer community with a share of 83.2%, while R is the second and least preferred choice with 16.8%. Table 8 delineates the level of difficulty of these two programming languages. Answering questions related to Python is more challenging based on FusedD. In contrast, R has more questions without accepted answers, and questions related to Python require a longer time to receive an accepted answer (12.7 hours).

74.5% of the inquiries include XAI software package tags. The distribution of these tools is depicted in Fig. 6. SHAP stands out as the most frequently mentioned software package in this context, accounting for 67.2% of the questions, positioning it at the forefront of the software packages regarding question volume. Following SHAP, LiME is associated with 14.8% of the questions, Yellowbrick with 9.8%, ELI5 with 3.8%, DALEX with 2.7%, and AIF360 with 1.7%.

Figure 7 presents the distribution of usage contexts for each XAI package across various machine learning domains. The majority of questions for most packages fall under the

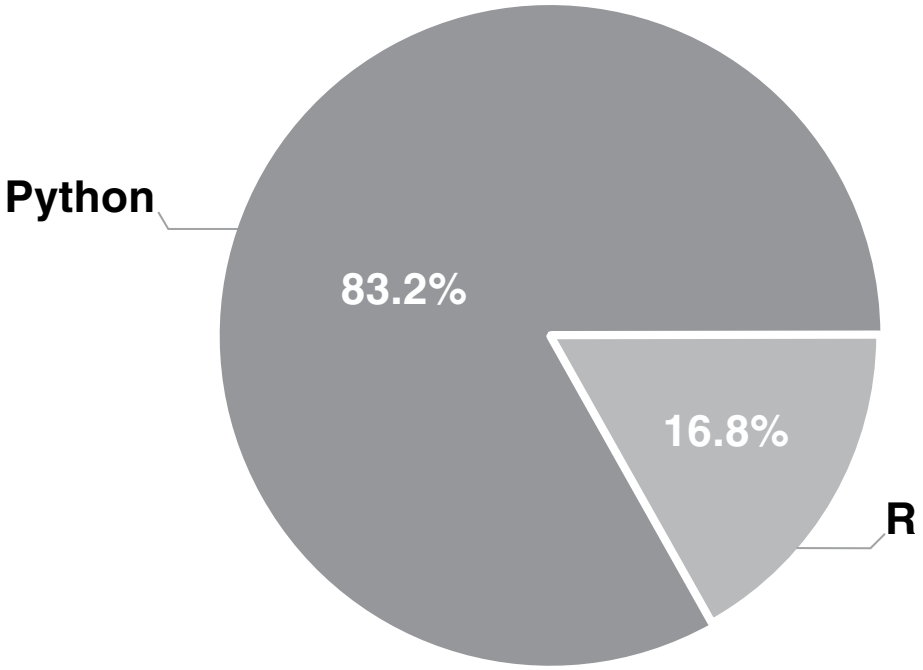

Fig. 5 Distribution of programming languages

Table 8 Difficulty of programming languages

Language	FusedD	Hrs To Acc.	W/o Acc. Ans
Python	1.1	12.7	66.9
R	0.9	7.7	70.5

General Machine Learning Tasks category, which includes standard machine learning procedures such as regression and classification. This indicates that developers frequently use XAI tools to explain and interpret models in common machine-learning scenarios.

For example, ELI5 has 70% of its usage in general machine learning tasks, highlighting its strong application in explaining and interpreting traditional models. Similarly, Yellowbrick shows a significant usage of 64% in this category, emphasising its role in visualising and diagnosing general machine learning models. SHAP and LIME also demonstrate substantial usage in general tasks, with 49.71% and 44.74% respectively, indicating their versatility in providing interpretability across a range of models.

In addition to general tasks, some packages are notably used in specialised domains. SHAP and LIME have appreciable usage in Computer Vision (11.11% and 10.53%, respectively) and Natural Language Processing (NLP) (9.94% and 21.05%, respectively), suggesting that these tools are effective in explaining models dealing with image and text data. ELI5 also shows usage in NLP (10%) and Time Series Analysis (10.00%), reflecting its applicability in diverse contexts.

AIF360 is primarily utilised for Data Preprocessing (66.67%), which aligns with its focus on fairness and bias mitigation during the data preparation phase. It also has usage in

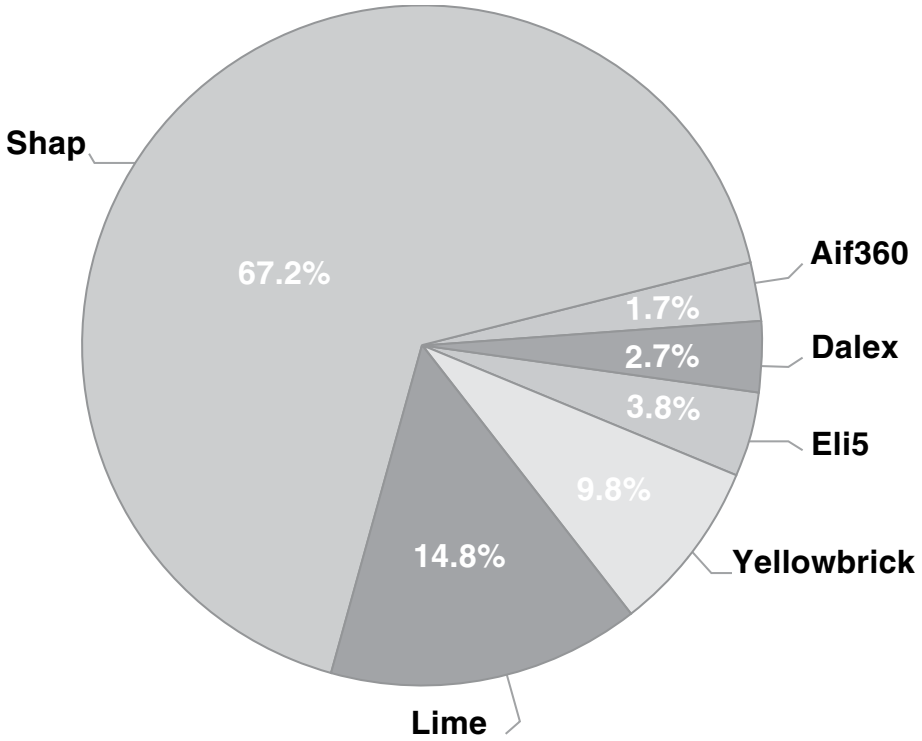

Fig. 6 Distribution of tools

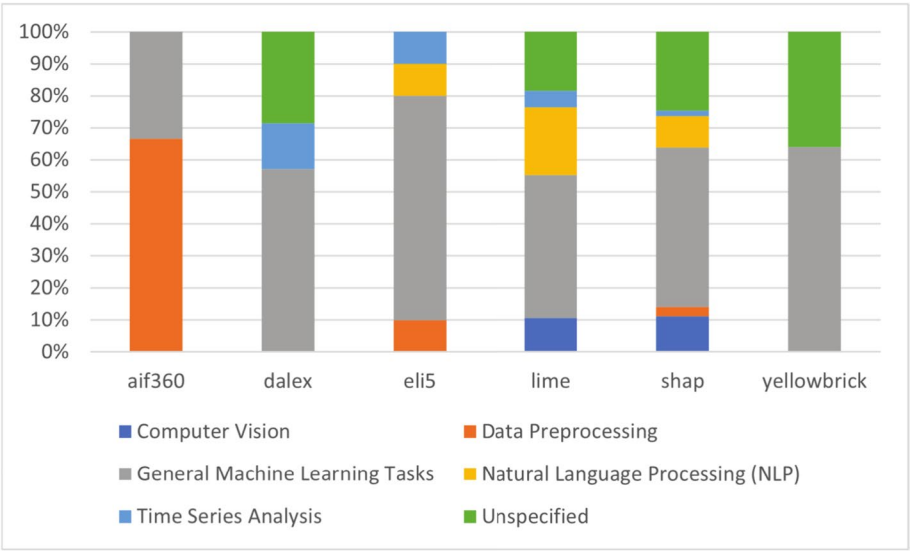

Fig. 7 Distribution of usage contexts of XAI packages

general machine learning tasks (33.33%), indicating its role in ensuring fairness throughout the modelling process. DALEX shows usage in Time Series Analysis (14.29%).

Table 9 outlines the popularity of the software packages based on the criteria mentioned in Section 3.2.3. SHAP garners the highest number of views at 1277.3, establishing itself as the most popular software package. ELI5 follows, securing the second position in popularity despite ranking fourth in the number of questions. According to the popularity criteria, ELI5 emerges as the second most sought-after XAI software package, boasting the highest average score of 1.36 among XAI software packages. Subsequently, Yellowbrick and DALEX take the third and fourth spots, respectively, with Yellowbrick claiming the second position regarding views with an average of 1147.7. DALEX, in turn, holds the highest average number of comments. Conversely, LIME, while ranking second in question volume, is not as popular, positioning it as the second least sought-after package, preceding AIF360 in terms of popularity.

Table 10 outlines the difficulty assessments for each package based on the criteria established in Section 3.2.3. DALEX emerges as the most challenging package by a significant margin, boasting the lengthiest median time required to get an accepted answer (221.51 hours) and a notably high percentage of questions without accepted answers (80%). Following closely, LIME occupies the second position with the highest rate of questions without accepted answers (84.55%). AIF360, securing the third spot, demonstrates the second-highest median time for receiving accepted answers (69.99 hours) and the lowest percentage of questions without accepted answers (46.15%). Notably, the initial three popular packages—SHAP, ELI5, and Yellowbrick—find themselves at the lower end of the difficulty spectrum, underscoring their significance as essential tools in this domain. Yellowbrick, distinguished by the shortest median time for obtaining an accepted answer (14.69 hours), occupies the final position in this ranking.

The questions of each package revolve around topics from Section 4.1. For example, Fig. 8 displays a query from the SHAP package, specifically about calculating SHAP Values. To gain a deeper understanding of the challenges faced by users of various XAI development

Table 9 Popularity of packages

Package	FusedP	#View	#Score	#CommentCount
SHAP	1.31	1277.3	1.35	1
ELI5	1.23	988.4	1.36	1.07
Yellowbrick	1.04	1147.7	1.07	0.64
DALEX	0.94	332.9	1.1	1.2
LIME	0.92	624	0.95	0.99
AIF360	0.56	802.1	0.23	0.46

Table 10 Difficulty of packages

Package	FusedD	Hrs To Acc.	W/o Acc. Ans
DALEX	2.29	221.51	80
LIME	0.97	43.92	84.55
AIF360	0.88	69.99	46.15
SHAP	0.64	15.15	70
ELI5	0.64	27.33	57.14
Yellowbrick	0.58	14.69	63.01

Fig. 8 An example of a question regarding a software package

tools, we constructed Table 11 to illustrate the proportion of each challenge encountered by users utilising XAI development tools, based on the issues identified in Section 4.1. The SHAP prog_err topic within the Tools troubleshooting topic was combined for more detailed investigations.

Users' most frequently asked questions regarding the most popular tool (SHAP) relate to troubleshooting and rectifying programming errors, accounting for 25.8% of the queries. This is followed by inquiries about fixing visualisation errors (19.2%), and further questions about the method of calculation and problems related to SHAP values (16.8%), indicating the necessity for additional clarification in this area. Regarding the second most popular programming tool, ELI5, inquiries predominantly pertain to troubleshooting and rectifying errors (60.71%), followed by a substantial number of questions related to feature importance and its calculation methods (14.29%).

There is a substantial number of inquiries related to Yellowbrick regarding visualisation issues (49.32%), underscoring the significant challenges users face when employing this tool in visualisation. Users of DALEX and LIME tools express significant concerns about model barriers and problems, with a notable volume of questions directed at the Model Barriers topic (55% for DALEX and 47.27% for LIME). Finally, 69.23% of questions from AIF360 users pertain to troubleshooting and fixing errors, while 30.77% focus on utilising this package to enhance the model.

Table 11 Package comparison concerning the topics

Topic	SHAP	ELI5	Yellowbrick	DALEX	LIME	AIF360
Concepts and Applications	1.8	7.14	0	0	3.64	0
Data Management	5.2	7.14	2.74	5	13.64	0
Feature Importance	7.4	14.29	4.11	5	3.64	0
Model Barriers	12.2	3.57	10.96	55	47.27	0
Model Improvement	6.6	3.57	4.11	5	4.55	30.77
Neural Networks Interpretation	5	3.57	0	0	2.73	0
SHAP Values Analysis	16.8	0	0	0	0.91	0
Tools troubleshooting	25.8	60.71	28.77	25	19.09	69.23
Visualisation	19.2	0	49.32	5	4.55	0

Previous studies (Dwivedi et al. 2023; Ding et al. 2022) have identified various XAI tools, and we have also listed 35 tools in Table 1. Among these tools, SHAP, LIME, and ELI5 have been consistently highlighted as important tools. Our study aligns with these findings, demonstrating that developers in the real world encounter challenges with tools such as DALEX, AIF360, LIME, SHAP, ELI5, and Yellowbrick, as specified in this section. We analysed package-specific challenges, revealing distinct trends (Table 11). Tables 9 and 10 also highlight popularity and difficulty metrics. This granular insight, often absent in traditional surveys, underscores the value of our real-world developer analysis.

Additionally, Dwivedi et al. (2023) discusses frameworks such as TensorFlow, Keras, PyTorch, and PyGam for model explainability. In this section, we adopt a reliable approach by focusing on the most widely used tools in the XAI community. Our research corroborates the prevalence of Python and R as the dominant programming languages in XAI, consistent with the conclusions drawn by previous work (Ding et al. 2022). Finally, our examination of real-world data reinforces the prevalence of Python, as demonstrated in Table 8 and Fig. 5. Additionally, we note that in the case of more Python usage, users generally face more perceived challenges than those using R.

Finding 4: Across three forums, 482 unique tags exist, with SHAP topping the list. Key AI tools include Scikit-learn, TensorFlow, XGBoost, Keras, and Matplotlib is a leading visualisation tool. Python and R are the prevalent XAI languages, with Python queries taking, on average, 5 hours longer than R to obtain accepted solutions. R also has a higher rate of unanswered questions. In Q&A forums, SHAP, ELI5, Yellowbrick, DALEX, LIME, and AIF360 rank as popular tools, respectively. SHAP leads with 67.2% iterations, and DALEX, LIME, and AIF360 are the most dif cult tools. Conversely, SHAP, ELI5, and Yellowbrick appear the least dif cult. Troubleshooting dominates queries for SHAP, ELI5, and AIF360, whereas Yellowbrick and SHAP often face visualisation issues. Additionally, LIME and DALEX have major model barrier issues.

4.5 How do the XAI Topics Evolve? (RQ5)

Figure 9 delineates the comprehensive and tangible influence of discussions surrounding XAI in our dataset. These discussions, originating in 2015 and evolving to 2023, reached their zenith in the latter part of 2022 and began their ascendency rapidly in January 2019. This trajectory demonstrates an increasing trend where the post frequencies heighten at an accelerated pace.

Subsequently, Fig. 10 enables us to trace the trend of the Categories from October 2015. The first query, pinpointed in the category of Data Management, explores how to comprehend the outcomes of KMeans clusters, which have been crafted with a matrix consisting of 81432 rows and 127 columns (Q33105634). A pattern commences in July 2016, with the discourse encompassing concepts and applications. As the first XAI tools emerged on the scene (LIME in August 2016), there was a significant focus on queries related to Troubleshooting in October 2017. Thus, a noticeable rising trend in this area became apparent. The Feature Interpretation category began its growth trajectory in April 2019, experiencing a more pronounced upward curve troubleshooting when compared to other categories. The

Fig. 9 Overall absolute impact

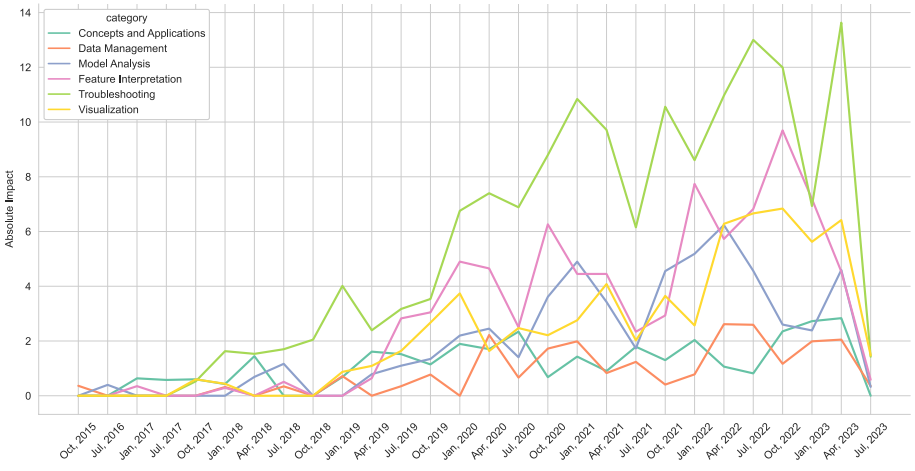

Fig. 10 The absolute impact of XAI categories

Visualisation category was initiated in October 2017 and the Model Analysis category followed suit in April 2018.

The introduction of the General Data Protection Regulation (GDPR) (Right to Explanation (Goodman and Flaxman 2017)) in May 2018 and the initial launch of the SHAP package in January 2019 marked the beginning of a robust upward trend in early 2019. Troubleshooting emerged as a significantly growing category at that time. Predominantly, the topics of Tools troubleshooting and Model barriers commanded the discourse at this point, and the sub-topic Model Training and Data Compatibility Challenges emerged, such as LIME explainer illustrating prediction probabilities different to the classifier prediction - sentiment analysis (Q54305070). Concurrently, queries related to fundamental concepts such as 'How can we create eXplainable Artificial Intelligence?' (Q10189) were still prevalent.

Between January 2020 and April 2020, there was an increase in the number of inquiries, followed by a slight decrease in July 2020, and then a peak in January 2021. These fluctuations coincided with the COVID-19 pandemic and the shift to online work, leading to various effects (Alamin et al. 2023). In January 2020, there was significant growth in the topics of Visualisation, Feature Interpretation, and Model Barriers, with subtopics including Feature Importance Variability, Model Misconfiguration and Usage Errors (e.g., How to use a 1D-CNN model in LIME? (Q59995744)). Moving on to the period between October 2020 and January 2021, where there was another increase, Tools troubleshooting remained the dominant topic, with additional growth in SHAP Values Analysis in October 2020 and Model barriers and Model Improvement in January 2021. Subtopics such as SHAP Values Calculation Methods, Shapley values without intercept (Q85022)), and Fairness and Bias, including calculating group fairness metrics with AIF360 (Q64506977)), were prevalent during this time.

In June 2022, SHAP released version 0.41.0, introducing various changes through contributions from different contributors. As illustrated in Fig. 9, we observe a peak in the number of questions from July 2022 to October 2022. During this period, the number of questions related to Visualisation and Feature Importance reached its zenith in October 2022. Subtopics within these categories, such as Plot Customisation and Styling (e.g., "How to edit the size of the axes text size on SHAP dependence plot in Python" - Q74121149) and Feature Selection (e.g., "How to select the best 30 features from 500 features for a sales prediction model where feature importance can change over time?" - Q115149), become noticeable.

The latest uptick is observable in April 2023, when the number of questions in the Tools Troubleshooting topic reaches its peak. Following this, Visualisation and Model Barriers become dominant topics, accompanied by subtopics such as Model Compatibility Issues (e.g., "Incompatibility between PySpark models and the SHAP package due to attribute mismatch and lack of direct support for PySpark models in SHAP" - Q76038173), Plot Customisation and Styling (e.g., "Change colour bounds for interaction variable in SHAP dependence_plot" - Q76013809), and Implementation and Runtime Errors.

In Fig. 11, we present additional insights into the progression of XAI topic classifications. Specifically, around mid-2017, it became evident that Concepts and applications emerged as the predominant subject category. According to the absolute impact metric, all six categories exhibit a uniform growth pattern. Conversely, the relative impact metric indicates a nearly identical evolution for the Troubleshooting and Feature Interpretation groups. Notably, in late 2018, there was a noticeable surge in discussions related to Visualisation, surpassing the Model Analysis category by January 2022, and eventually outpacing discussions on Feature Interpretation. This detailed analysis of the evolutionary trends is crucial as it highlights that, despite Feature Interpretation topics being the second-largest category, the discourse around Visualisation is rapidly advancing and requires heightened attention from the XAI community.

Finding 5: Discussions about XAI began in late 2015 and mostly centred on de ning concepts and applications. Since 2019, the discussions have gained more attention. Troubleshooting and Feature Interpretation were the main topics for a while, but recently, Visualisation inquiries have become more popular.

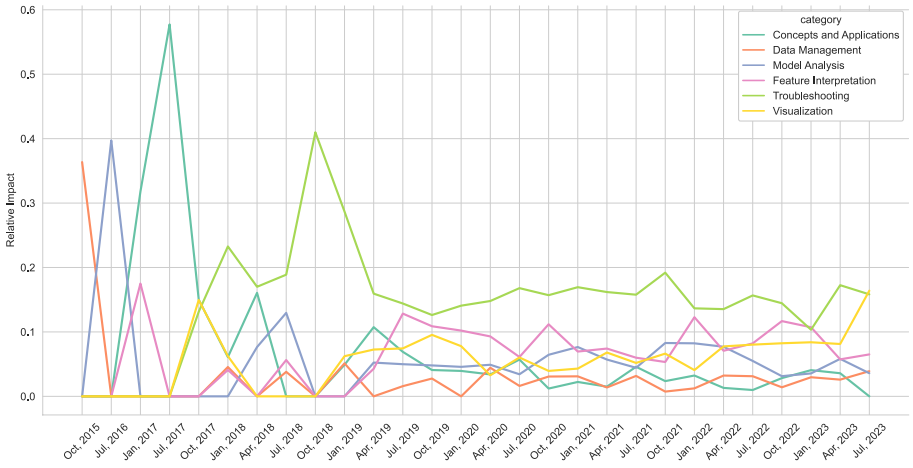

Fig. 11 The relative impact of XAI categories

5 Discussion

5.1 Topic Analysis

In this study, as explained in Section 3.2.1, we employed a mixed-method approach. Initially, we utilised LDA topic modelling with the Dirichlet distribution to explore discussions among practitioners regarding XAI. We followed established academic practices, as detailed in Section 3.1, for configuring parameters and hyperparameters. In line with the guidance in Section 3.2.1, we manually annotated topics to ensure optimal outcomes, avoiding suboptimal results (De Lucia et al. 2014). To enhance the granularity of LDA findings, we took inspiration from previous studies (Wang et al. 2023; Alfayez et al. 2023; Naghashzadeh et al. 2021), where discussions were manually categorised. Employing a reliable method and adhering to standards established in prior research, we systematically categorised the topics and classified each of them into different subtopics within this study.

For evaluating the LDA results, we assessed the coherence score for various values of K, in line with similar studies (Alamin et al. 2023; Uddin et al. 2021; Abdellatif et al. 2020; Han et al. 2020). Specifically, we used the C_V coherence score, which measures the degree of semantic similarity between high-scoring words in each topic. Given the probabilistic nature of LDA (Agrawal et al. 2018), multiple runs on the same XAI dataset can yield different results. To address this variability, we ran our LDA model three times for each value of K, ranging from 5 to 30, and computed the average coherence scores.

Figure 12 displays the coherence scores across the three runs, with the bold line representing the mean coherence and the shaded area indicating the variance. As can be seen in the figure, the average coherence increased up to $K = 8$ and gradually decreased afterward. The range of K values from 8 to 10 is particularly significant, as it shows the highest average coherence across multiple runs and also has the highest coherence in individual runs.

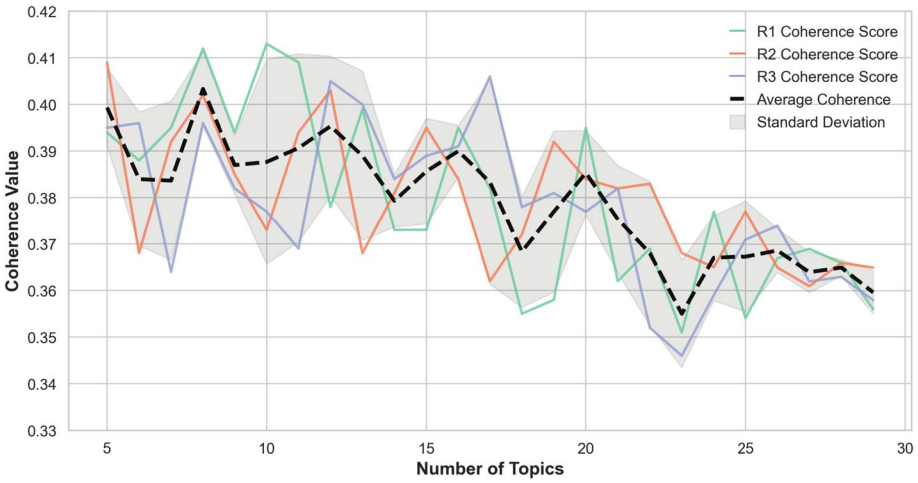

Fig. 12 The diverse coherence scores for different values of K in various runs

However, the model with $K = 10$ in round one had the highest coherence in all three runs. According to previous studies (Abdellatif et al. 2020) that have examined models with close coherence scores, we considered models in both scenarios: the one with the highest coherence and the one with the highest mean coherence with the lowest standard deviation.

It is important to note that coherence is only one of the criteria for evaluating the quality of topics in a model. Since we adopted a hybrid approach, the diversity of the topics generated in this study was also crucial. After examining the topics generated by the two models ($K = 8$ and $K = 10$), authors agreed that the model with $K = 10$ not only had the highest coherence score among all the models but also produced topics with higher diversity and interpretability (as shown in Fig. 13). Therefore, we selected the model with $K = 10$ as the final model.

Additionally, we used the PyLDAvis tool to visualise the LDA results for $K = 10$, which revealed well-distinguished and interpretable topics, as depicted in Fig. 13. This further validated our choice of $K = 10$ for the final model.

5.2 XAI in Comparison with other ML Areas

With the increasing demand for transparency and accountability in ML models, the concept of XAI has gained significant attention in recent years. As a result, there has been a considerable increase in the number of XAI-related questions posted on Stack Exchange, as shown in Section 4.5. However, whether XAI questions are inherently more challenging than other ML questions on the platform remains unclear. To investigate this, we aim to compare the difficulty of XAI questions with that of other ML questions. By analysing various difficulty metrics that we used in Section 4.3 (the percentage of unaccepted answers and the median time to get an accepted answer), we can gain insights into the perceived difficulty of XAI questions relative to other ML questions on Stack Exchange forums.

As indicated in Section 2, previous studies have explored Stack Exchange posts across various domains, such as big data (Bagherzadeh and Khatchadourian 2019), chat-bots

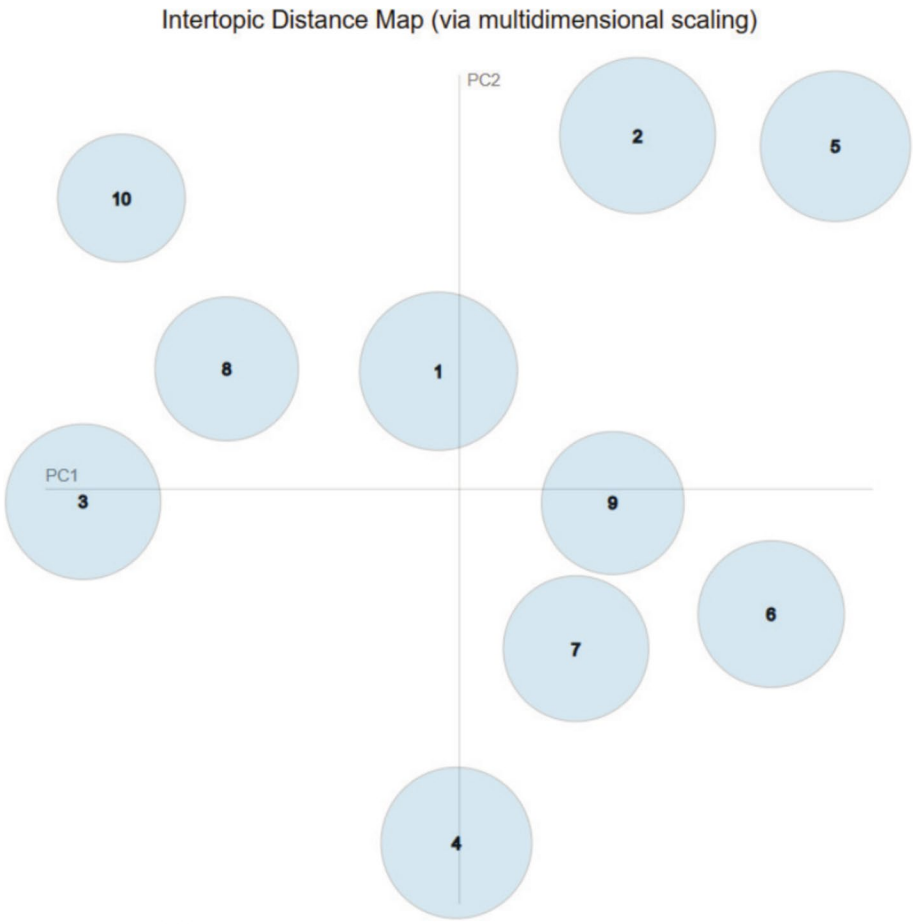

Fig. 13 Topic distinctiveness within the $K = 10$ model

(Abdellatif et al. 2020), machine learning (Alshangiti et al. 2019), and deep learning frameworks (Han et al. 2020). In this study, we specifically focus on papers within the machine learning domain, examining two metrics: the percentage of questions without an accepted answer and the median hours to accept an answer. Among the relevant literature, we identified papers in the machine learning domain that reported these metrics, they work on machine learning (Alshangiti et al. 2019), autoML (Wang et al. 2023), and deep learning-based software (Chen et al. 2020).

Table 12 compares XAI and other machine learning topics, assessing difficulty measures based on the mentioned articles. XAI posts are notably more challenging, likely due to their emergent and intricate nature. In contrast to ML (61%) and Deep Learning (DL) software deployment (70.7%) queries, XAI posts have a higher percentage of questions (72.7%)

Table 12 Evaluating difficulty metrics in different machine learning studies

Difficulty Metric	XAI	AutoML	DL Software Deployment	ML
Hrs To Acc.	20.19	6.89	6.75	3.83
W/o Acc. Ans	72.7%	73%	70.7%	61%

without an accepted question. AutoML questions, with a 73% rate, show only a slight 0.3% difference from XAI (72.7%) in questions without an accepted answer, highlighting the difficulty in obtaining accepted answers in this domain. The table also reveals significant disparities in the time taken to receive an accepted response for XAI posts compared to other machine learning topics. While ML posts took an average of 3.83 hours, DL software deployment questions took 6.75 hours, and AutoML questions took 6.89 hours, XAI posts required an average of 20.19 hours for an accepted answer. This underscores the evolving nature of XAI and emphasises the need for further development and community building in this field to match the maturity of other machine learning domains. Despite these challenges, as indicated in RQ3, XAI posts still attract considerable views and comments, showcasing an involved and dedicated community. In conclusion, the results underscore that XAI is in its early stages and demands more attention and growth to reach the level of maturity observed in other machine learning domains.

Based on the analysis conducted in Section 4.5, we observed the progression of inquiries and responses within Stack Exchange forums. To delve deeper into the tools reviewed in Section 4.4 and gain a more comprehensive understanding of how developers utilise these tools in real-world scenarios, we employed a method utilised in previous studies (Wang et al. 2023; Tan et al. 2022). We compiled downstream repositories of selected XAI packages using version V of World of Code (WoC) (Ma et al. 2019). WoC is an infrastructure designed to store vast amounts of open-source (version control system) data and extract Git objects from open-source repositories on major code hosting platforms. We utilised WoC to extract the downstream repositories of each XAI package, which we identified through the import directives and removed the fork repositories to avoid duplication. In total, we amassed 17,932 downstream repositories, and each package's overall downstream repositories trend is depicted in Fig. 14.

5.3 Evolution of XAI Packages Usage Overtime

Moreover, we scrutinised the progression of questions associated with each of the packages over time, showcased in Fig. 15. Figure 14 illustrates that, on average, XAI packages

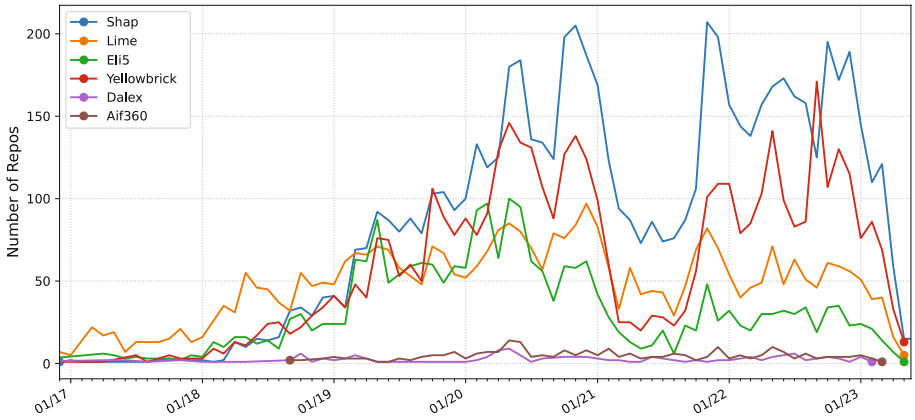

Fig. 14 The number of new downstream repositories for each package

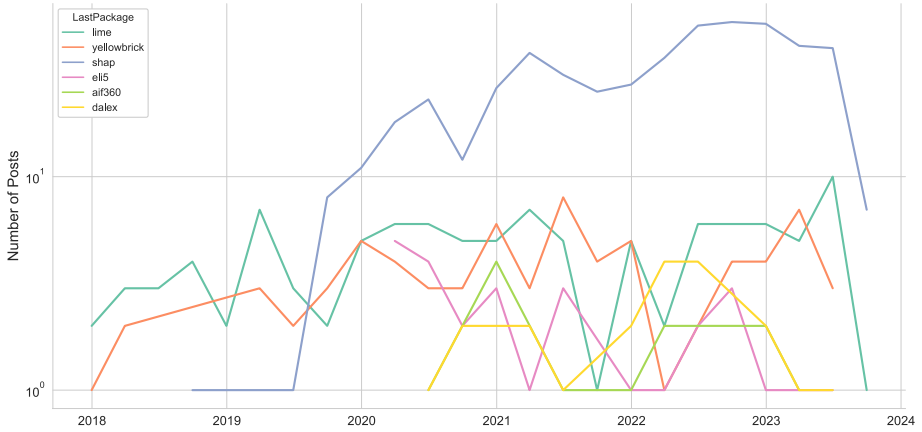

Fig. 15 The number of questions for each package

acquire their first downstream repository approximately 6 months after the release of their initial version. A noticeable upward trend commences within a year, leading to increased utilisation of these tools. Figure 14 also indicates an escalating trend in the number of downstream repositories from January 2018, signifying a rapid expansion in the usage of XAI packages. This is evidenced by the peak usage of SHAP tools in November 2021, Yellowbrick in October 2022, LIME in December 2021, and ELI5 in May 2020. A comparison of Figs. 12 and 13 demonstrates that as the utilisation of these packages surged in mid-2019, the volume of related questions also increased concurrently.

The increasing trend of downstream repositories reveals that the LIME package was dominant until the beginning of 2019. However, in May 2019, the rising adoption of SHAP, Yellowbrick, and ELI5 resulted in the displacement of LIME, making SHAP the primary tool, except in September 2022 when Yellowbrick took the lead. The results obtained in this section confirm the popularity of SHAP, Yellowbrick, and ELI5 packages.

5.4 Packages Specification-driven Analysis

In Section 4.4, we investigated the commonly used tools in StackExchange forums. We found that the growing interest in XAI has spurred the creation and acceptance of various tools to enhance the interpretability and transparency of machine learning models. Notable options gaining popularity in the XAI include SHAP, ELI5, Yellowbrick, DALEX, LIME, and AIF360. This section delves into the unique characteristics of these tools, providing insights into varying levels of user engagement. Table 13 shows the specifications of each XAI package.

SHAP stands out as the most popular choice, drawing on its robust foundation in game theory to provide both global and local explanations. Its flexibility shines through as it caters to both model-agnostic and model-specific scenarios, making it a preferred option for practitioners seeking comprehensive interpretability across various model types. However, it is essential to note that while agnostic to package dependencies, SHAP may present computational complexities, especially for larger datasets.

Table 13 Packages specifications

Package	Explanation Level	Implementation Level	Model Dependency	Package Dependency	Explanation Type
SHAP	Both	Post-hoc	Agnostic and specific	Agnostic	Game-theory-based
ELI5	Both	Post-hoc	Agnostic	Agnostic	Importance-Based
Yellowbrick	Both	Post-hoc	Agnostic	Specific (Scikit-learn)	Visualisation-Driven
DALEX	Both	Post-hoc	Agnostic	Agnostic	Importance and Dependence
LIME	Local	Post-hoc	Agnostic	Agnostic	Local-Surrogate
AIF360	Global	Post-hoc	Agnostic	Agnostic	Bias and Fairness Metrics

On a simpler note, ELI5 is another popular tool due to its straightforwardness and emphasis on importance-based explanations, applicable at both global and local levels. Its model-agnostic nature aligns with a broad user base, offering seamless integration with other Python tools without specific dependencies. This simplicity in interpretability makes ELI5 accessible to users who may not require the depth provided by game theory-based explanations.

Moving on, Yellowbrick sets itself apart by prioritising visualisation-driven interpretability, supporting both global and local explanations. However, its close tie to Scikit-learn might limit its adoption outside this ecosystem. For users within it, Yellowbrick offers an intuitive and visually appealing approach to understanding model behaviour.

DALEX, similar to ELI5, provides importance and dependence-based explanations while maintaining model-agnostic flexibility. Striking a balance between detailed importance outputs and nuanced dependence plots, DALEX caters to various interpretability needs without burdening users with extensive package dependencies.

LIME takes a specialised approach, specifically delivering local explanations for individual predictions. This targeted method appeals to users interested in detailed analyses of specific instances but may fall short for those requiring a broader, more global interpretation of model behaviour.

Lastly, AIF360 is a specialised toolkit concentrating on bias detection and mitigation, offering global explanations using bias and fairness metrics. Tailored for applications and research areas where fairness is a critical concern, AIF360 holds importance. AIF360's focus on bias and fairness made it popular compared to other tools.

The package preferences observed can generally be attributed to a trade-off between explanatory depth and ease of use, the specificity of application, dependability on existing machine learning tools, and the nature of explanation (be it global, local, or both). Understandably, practitioners may prioritise tools that align closely with their workflow preferences and the specific demands of their interpretability tasks. The future may see a shift in these preferences as new challenges in interpretability arise, and as these packages evolve to meet those challenges. In general, the analysis indicates that users seek a comprehensive and versatile tool that is user-friendly and provides clear explanations. The desire is for a tool that supports various functionalities, is easy to navigate, and yields comprehensible explanations.

6 Implications

6.1 XAI Developers

In the field of XAI, there is a growing recognition of the need for tools that effectively overcome model barriers and simplify data interpretation. As noted by prior research (Saeed and Omlin 2023), there's a pressing need for practical and comprehensive guidance, notably in the form of step-by-step instructions, as indicated by the prevalence of D-type EC and 'How' questions, underscoring a demand for XAI tools that integrate theoretical concepts with real-world applications (de Bruijn et al. 2022).

Figure 16 indicates that compatibility issues (involving models, versions, and libraries), model misconfigurations, and usage errors are intriguing and challenging, highlighting a need for novel solutions. Automated tools, clearer workflows, and educational resources could significantly aid the XAI community in addressing these complexities. Therefore, as stated in previous studies (Ali et al. 2023) developers should focus on creating user-friendly tools with clear, comprehensive documentation and automated features, especially in Python and R languages, to enhance ease of use and engage actively in open-source communities and projects. Despite certain challenges, the popularity of tools like SHAP, ELI5, and Yellowbrick reflects the necessity to balance usability and depth of explanations.

Analysing Table 11, we have observed that different XAI tools face varying challenges in different areas. For instance, tools like SHAP and Yellowbrick encounter difficulties in visualisation, ELI5 in feature importance, DALEX and LIME in model barriers, and AIF360 in model improvement. As XAI technology continues to develop, developers must prioritise making these tools more accessible while retaining their comprehensive details.

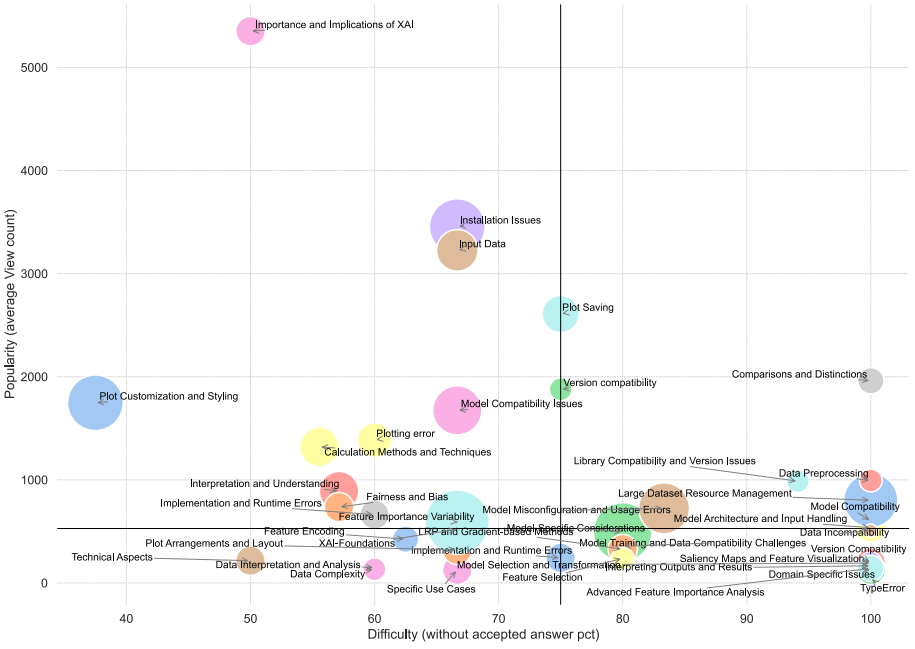

Fig. 16 The Popularity vs Difficulty bubble plot

Lastly, questions highlight a distinct requirement for tools to assess XAI tools outcomes and explanations (e.g., evaluation of machine learning explainers (Q55984) and Evaluation of SHAP outputs (Q72197794)). This necessitates developers' emphasis on formulating robust evaluation frameworks for XAI tools. Key to this process is the creation of metrics and methodologies for gauging the effectiveness and accuracy of XAI model explanations, ensuring they are technically proficient but also meaningful and trustworthy from the user's perspective. Implementing such evaluative mechanisms is pivotal in augmenting XAI's credibility and practical utility across diverse real-world applications.

6.2 XAI Educators

In Section 4.5, the discourse within the XAI community has evolved from theoretical to practical narratives. However, this shift should not undermine the significance of foundational debates, as evidence suggests ongoing educational deficiencies. For illustration, Fig. 17 was crafted from post tags and their frequency, depicting the trade-off between model accuracy and interpretability. The bubble sizes in the figure represent the recurrence of each tag in the dataset, This figure clearly shows that, although the most demand is for black box models with high accuracy and low interpretation, there is still the use of complex XAI models in models with high intrinsic interpretability. For instance, Shapley values for Logistic Regression (Q60434320). This trend highlights educational shortfalls in the field.

While research suggests that explanations are anticipated to play a crucial role within a broader optimisation framework to attain objectives such as enhancing a model's performance or simplifying its structure (Samek and Müller 2019), there are recommendations

Fig. 17 Relationship between model accuracy and interpretability and usage of them in Q&A platforms

supporting the use of XAI models to increase the clarity of AI systems, the community must approach this with caution and deliberation. Employing complex XAI models when simpler alternatives would suffice may lead to unnecessary development and resource distribution complications. Moreover, as highlighted by Springer and Whittaker (2019) an abundance of explanations can result in information overload, consequently impeding the ability of individuals to develop a practical and comprehensive understanding of the system's workings.

Moreover, prior research (Alshangiti et al. 2019) indicates that visualisation traditionally received low attention in broader AI studies. However, our findings in Sections 4.1 and 4.3 highlight its significant importance in XAI. Instructors should initially focus on diverse aspects, including creating various plots, customisation, and integrating popular tools like Matplotlib.

The prevalence of practical 'how' queries in data visualisation and management reflects a demand for actionable knowledge and skills in these areas. Conversely, theoretical 'what' questions dominate model analysis discussions, indicating a need for a foundational understanding. 'Why' questions, often associated with troubleshooting, indicate a desire for deeper insights into AI challenges. Educators must craft curricula that balance hands-on skills with a robust theoretical base, equipping learners to implement, comprehend, and critically engage with AI technologies. Such an educational framework is imperative to address the diverse learning requirements of the community effectively.

6.3 XAI Researchers

In Fig. 16, we observe significant interest and challenges related to compatibility issues in user XAI discussions. These issues revolve around visualisation tools, models, and software versions. This situation suggests that targeted research needs to develop integrated and standardised tools. Researchers should prioritise creating frameworks that ensure compatibility across various platforms and facilitate the seamless integration of XAI into diverse environments. They should also understand the specific needs and contexts in which these tools will be employed. The same figure also highlights that XAI frameworks reflect a broader need to simplify and harmonise the utilisation of intricate interpretation tools like SHAP values and neural network methods, including LRP and gradient-based methods. Practical challenges in this domain are common but require common solutions. Future studies should aim to demystify XAI by providing clear, concise, and context-specific explanations of different XAI methodologies to address the prevalent confusion surrounding XAI definitions and their practical implications. Researchers could develop comprehensive guides or frameworks that help users easily navigate the complex landscape of XAI, enhancing their understanding and ability to employ these tools effectively.

Figure 16 also reveals that despite the efforts of several studies (Nagahisarchoghaei et al. 2023; Das and Rad 2020; Angelov et al. 2021) to clarify various definitions of XAI for users, questions about comparing and distinguishing these definitions persist, even though XAI is popular. This presents an opportunity for research to articulate these definitions more practically. Prior research (Nagahisarchoghaei et al. 2023) has identified the applications of XAI technologies. The primary application of these technologies is to enhance AI system transparency for more practical reliability (Hanif et al. 2023). However, as discussed in Section 6.2, developers predominantly employ XAI for tasks related to the development process, such as debugging and model selection, rather than prioritising end-user transpar-

ency. This discrepancy highlights a gap between XAI's potential and its primary application. Researchers should strive to bridge this gap by developing inherently more user-centric tools, focusing on end-user interpretability and trust, and conducting empirical studies to understand the barriers and facilitators to use XAI to enhance end-user transparency.

This analysis reveals a potential mismatch between XAI surveys and developer practice. Surveys (Dwivedi et al. 2023; Ding et al. 2022) often overlook packages like Yellowbrick, despite their popularity among developers, and may highlight tools with limited real-world adoption. To bridge this gap, researchers should focus on tools that directly address practical challenges faced by the XAI community. This gap offers a research opportunity that addresses the "what" questions and reshapes XAI tools to serve end-users better. Researchers could also explore innovative ways to incorporate user feedback into the design and development of XAI systems. Finally, researchers are encouraged to foster collaborations across disciplines to leverage diverse expertise and perspectives, ensuring that the developed solutions are robust, versatile, and truly reflective of the multifaceted nature of XAI. By undertaking these suggestions, researchers can significantly contribute to advancing XAI as a theoretically rich and practically impactful field.

7 Best Practices

Building upon the CDAC life cycle (De Silva and Alahakoon 2022), which provides a comprehensive framework for the AI development process, and the insights outlined in Saeed and Omlin (2023), we present best practices for AI developers, organised into three key stages: design, development, and deployment.

7.1 Design Phase

Prior research (Ghorbani et al. 2019; Wang et al. 2019; Adebayo et al. 2018) has underscored the criticality of ensuring that the outcomes and explanations of XAI algorithms are intricately tied to the input data, with particular attention to fairness and potential biases. The discrepancies in data collection methods and inherent issues within learning algorithms can significantly skew the results (Das and Rad 2020).

The importance of disclaimers and data clarification for end users by developers and data handlers has been emphasised in the existing literature (Reiter 2019; Ahmad et al. 2019). This emphasis is reflected in our findings. In RQ1 (Section 4.1), we observed that while topics related to Fairness and Bias exist within the XAI community, they are not among the most frequently discussed issues. Specifically, the sub-topic Fairness and Bias constitutes only a small fraction of the discussions, indicating that developers may not focus sufficiently on these critical aspects during the design phase. For instance, questions like "How does one use the Fairlearn metrics to decide whether a feature is biased or not?" (Q63291746) demonstrate a growing interest in understanding data-centric biases during the design phase.

However, questions of this nature are relatively infrequent compared to questions focusing on detecting biases in learning models after development (e.g., "Fairness metrics for multi-class classification" (Q66574745)). This trend suggests a pivotal need within the XAI community to focus more on data quality and fairness during the design phase rather than solely on post hoc bias detection.

Our results in RQ3 (Section 4.3) further underscore this need, revealing that topics related to Data Management are among the less popular and less challenging topics, indicating that developers might be underestimating the importance of data quality issues. Additionally, the Discussion section highlights that despite the critical role of data fairness, there is a gap between the theoretical emphasis on this topic and its practical application by developers.

> **Best Practice 1:** Emphasise data quality and fairness from the start. Use tools like AIF360 to detect biases during data collection. Transparently report methodologies and address any biases to build trust in the model's explanations.

7.2 Development Phase

Drawing from previous studies (Felzmann et al. 2020), integrating transparency into AI system design and development is crucial. This principle is central to our approach in the AI development phase, starting with detailed data preprocessing and preparation, a critical and challenging step, particularly with large data volumes, as shown in Fig 16. Introducing XAI techniques early on, such as using SHAP values to filter outliers (69225478), offers significant advantages but also introduces complexities, especially in data management. Therefore, AI developers should seamlessly incorporate XAI methodologies from the beginning. This integration goes beyond technical aspects to include explainability, thus enhancing model transparency and meeting user needs.

As delineated in Section 4.4, the packages SHAP, LIME, ELI5, DALEX, and AIF360 are frequently discussed and utilised, each possessing distinct advantages and constraints. The selection of an optimal tool is paramount and should be tailored to varying requirements.

Our analysis for RQ4 reveals that tools such as SHAP and ELI5 are both popular and relatively easy to use, whereas tools like DALEX present greater challenges (Tables 9 and 10). Furthermore, the analysis based on package specifications highlights the distinct features of these tools, underscoring the importance of choosing the right tool according to the specific needs of the application. Figure 7 provides an additional perspective by categorising these tools based on their usage.

SHAP is known for its accurate, consistent explanations but is computationally intensive. LIME's model-agnostic nature offers wide applicability, yet its local approximations might lead to inconsistencies (e.g., the LIME package is not able to get predictions for CaretStack (55001428)). ELI5 is user-friendly and supports various frameworks but may oversimplify complex interpretations. DALEX provides a versatile and unified interface but might lack specific details compared to specialised tools. Lastly, AIF360 focuses on understanding and mitigating bias to enhance fairness but may not cover all aspects of model interpretation.

For simple AI applications, tools like ELI5 or LIME are recommended due to their ease of use and quick insights. In contrast, complex systems might benefit from a combination of SHAP for its comprehensive global perspective and DALEX for its unified explanation approach. Employing AIF360 is also crucial to ensure fairness and transparency. The choice of tools should balance the application's specific needs, model complexity, and computational resources with each package's positive and negative aspects. One of the most challenging parts is that the model should be equipped with robust evaluation criteria that assess its performance and the quality of the explanations, guaranteeing their comprehen-

sibility and utility to the intended audience. Questions like the evaluation of SHAP outputs (72197794) underscore the significance of this matter, and this question has yet to be answered.

> **Best Practice 2:** Integrate XAI techniques early in the development process to enhance transparency and address data management challenges. Select appropriate XAI tools based on application needs—use ELI5 or LIME for simple tasks, SHAP for complex systems, DALEX for uni ed explanations, and AIF360 for fairness. Develop robust evaluation criteria to assess both performance and explanation quality, ensuring explanations are understandable and useful to the intended audience.

7.3 Deployment Phase

Previous studies (Bhatt et al. 2020) indicate that the deployment phase of AI systems needs a shift in focus toward end-user interpretability. Our data confirm that current deployments are predominantly geared toward machine learning engineers for debugging purposes rather than for end users who these models directly impact. This is reflected in the limited number of questions concerning end-user explanations. For example, in (Q103545), the user is considering creating a system equipped with XAI to express explanations for the better choice of teachers by students, highlighting a rare focus on end-user needs.

According to the review of tags in RQ4 (Section 4.4), the three AutoML tools AutoML H2O (LeDell and Poirier 2020), Google Vertex AI (Google Cloud 2022), and Databricks (Etaati and Etaati 2019) have a significant number of questions related to creating explanations for their models. This indicates a focus on enhancing model performance and transparency, but primarily from a developer's perspective.

However, for a truly transparent system, developers must aim to increase trust and facilitate better decision-making for end users. This requires integrating popular XAI tools that offer clear explanations and reshaping the deployment approach to prioritise user-centric explanations. By doing so, the deployment phase can evolve from improving model accuracy and efficiency to enhancing the interpretability and accessibility of AI systems for all users, ultimately fostering a more trustworthy and inclusive AI environment.

> **Best Practice 3:** Shift deployment focus to user-centric explanations to enhance trust. Utilise AutoML tools like AutoML H2O, Google Vertex AI, and Databricks for accessible explanations. Improve interpretability and accessibility for end users.

8 Threats to Validity

Internal Validity To uphold the internal validity of our research, we employed LDA topic modelling to organise forum posts categorically, then engaged in manual analysis to identify sub-topics, ensuring a deeper and more precise understanding. We are cognizant of the potential threats to internal validity, particularly concerning experimenter bias. Multiple layers of review were instituted to address these concerns. The initial labelling of topics was performed collaboratively,

through a concerted effort involving four authors, to diminish any individual biases that might skew the findings. To further refine this process, the first author conducted a comprehensive review of the resultant labels, subsequently refining them through deliberations with the second author to attain consensus. This multistage approach not only enhances the robustness of our findings but also ensures that the thematic labels assigned are representative and inclusive of divergent perspectives. We employed a dual approach involving both LDA methods and manual topic selection to mitigate potential internal threats. While manual subtopic selection may introduce human biases, utilising the LDA algorithm for topic generation can also result in biases stemming from the manual assignment of topic names. Therefore, integrating both methods in this study validates the accuracy of topics generated by LDA. We posit that the synergistic application of manual methods and topic modelling algorithms enhances the likelihood of producing more robust and accurate outcomes.

External Validity The study's external validity focuses on extending results beyond the study's context to a broader domain (Malhotra 2016). Our analysis involves three Stack Exchange forums, but XAI developers may use other forums that are not covered. We use specific Stack Exchange tags to locate posts, but we are potentially missing some XAI-related tags. To address this, we follow previous research (Openja et al. 2020; Uddin et al. 2019; Li et al. 2021) and iteratively identify relevant tags, resulting in a dataset of around 1000 posts consistent with prior works (Wang et al. 2023; Alfayez et al. 2023). While Stack Exchange data offers generalizability due to its widespread use, we recognise the study can be enhanced by incorporating discussions from other forums and conducting surveys and interviews with XAI developers.

Construct Validity One potential threat revolves around selecting the topic modelling technique, where we determined $K = 10$ as the optimal number of topics for our dataset. This choice significantly impacts LDA's output, and we explored various K values based on prior research (Abdellatif et al. 2020; Alamin et al. 2023; Li et al. 2021). To enhance labelling accuracy, authors meticulously reviewed keywords and samples from LDA-generated topics, engaging in group discussions for labelling assignments, Additionally, the sub-topic selection process contributed to more effective labelling of topics. Another potential concern revolves around the metrics used to assess popularity and difficulty, posing a potential threat to construct validity. Despite their application in similar studies and recognition as reliable measures (Yang et al. 2016; Bagherzadeh and Khatchadourian 2019; Alamin et al. 2023; Uddin et al. 2021), these metrics require careful consideration.

In presenting our findings, the study utilised standard statistical measures like percentages and medians. The Cohen's Kappa coefficient evaluated inter-rater agreement, while the Kendall correlation test examined the relationship between difficulty and popularity metrics. These statistical measures and tests adhere to recommended practices in empirical standards for software engineering research (Ralph et al. 2020), ensuring alignment with the intended measurements and reinforcing the study's methodological rigour.

9 Conclusion

Explainable Artificial Intelligence (XAI) is increasingly recognised as essential for ensuring transparency and understanding in AI-driven systems, as evidenced by discussions in Q&A forums. Our thorough investigation examines the discourse surrounding XAI within these platforms. We analysed a diverse dataset covering a wide range of topics and categorised them into 40 sub-topics, 10 topics, and 6 main categories: Troubleshooting, Feature Interpretation, Visualisation, Model Analysis, XAI Concepts and Applications, and Data Management. Troubleshooting emerges as the most significant category, with notable topics such as Tools Troubleshooting, Model Barriers, and Visualisation taking precedence. Plot Customisation and Styling, Tools Implementation and Runtime Errors, and Model Misconfiguration and Usage Errors emerged as the most prominent sub-topics. Queries mainly take a "How" approach, with Visualisation (68.29%) and Data Management (66.67%) having the highest prevalence, indicating a strong emphasis on practical application. "What" and "Why" inquiries are also common, particularly within XAI Concepts and Applications (45%) and Model Analysis (23.08%), showing a desire for a deeper understanding of AI methodologies. XAI Concepts and Applications are the most popular topics, attracting the highest engagement scores. However, Feature Importance faces challenges in gaining attention despite Many unanswered queries. Visualisation has a low rate of unaccepted answers but a considerable response time, reflecting its complexity. Our investigation identified 482 unique tags across the forums, with SHAP being the most prevalent. Key AI tools such as Scikit-learn, TensorFlow, XGBoost, Keras, and Matplotlib, along with Python and R, are central to XAI discussions. Python-related issues tend to persist longer before resolution than R. Troubleshooting remains predominant in discussions surrounding SHAP, while Visualisation issues frequently arise alongside Yellowbrick. XAI discussions started in late 2015, with a notable surge in interest in 2019, focusing on Troubleshooting and Feature Interpretation. The insights from our analysis shed light on critical areas of engagement for practitioners, researchers, and developers, guiding stakeholders in prioritising efforts to address the complexities of XAI and improve its accessibility within the field. In the future, we plan to delve deeper into individual topics, such as conducting surveys of XAI developers to gain deeper insights into the observed topics and sub-topics.

Author Contributions The first three authors contributed to the conception and design of the study. Mohammad Mahdi Sayyadnejad was responsible for material preparation, data collection, and initial analysis. Mohammad Mahdi Sayyadnejad and Ali Asgari wrote the first draft of the manuscript. Revisions were assigned by the third author, carried out by the first author, and double-checked by the second author. All authors reviewed earlier versions of the manuscript, contributed to its revision, and approved the final version.

Data Availability Statement In our commitment to transparency and reproducibility, we have made all the data used in this study publicly available through a Zenodo repository (Sayyadnejad 2024). The repository includes Stack Exchange posts and tags, Manual Analysis Documentation, the data of downstream repositories, and the LDA model used in our analysis.

Declarations

Funding Not applicable.

Ethical Approval Not applicable.

Informed Consent Not applicable.

Conflicts of Interest The authors declare that they have no conflict of interest.

Clinical Trial Number Not applicable.

References

Abdellatif A, Costa D, Badran K, Abdalkareem R, Shihab E (2020) Challenges in chatbot development: a study of stack overflow posts. In: Proceedings of the 17th international conference on mining software repositories, pp 174–185

Adebayo J, Gilmer J, Muelly M, Goodfellow I, Hardt M, Kim B (2018) Sanity checks for saliency maps. Adv Neural Inf Process Syst 31

Agrawal A, Fu W, Menzies T (2018) What is wrong with topic modeling? and how to fix it using search-based software engineering. Inf Softw Technol 98:74–88

Ahmad MA, Eckert C, Teredesai A (2019) The challenge of imputation in explainable artificial intelligence models. arXiv:1907.12669

Ahmed S, Bagherzadeh M (2018) What do concurrency developers ask about? a large-scale study using stack overflow. In: Proceedings of the 12th ACM/IEEE international symposium on empirical software engineering and measurement, pp 1–10

AI Stack Exchange (2023) Artificial intelligence forum. Available: https://ai.stackexchange.com, online; Accessed 1 June 2023

Al Alamin MA, Malakar S, Uddin G, Afroz S, Haider TB, Iqbal A (2021) An empirical study of developer discussions on low-code software development challenges. In: 2021 IEEE/ACM 18th International Conference on Mining Software Repositories (MSR). IEEE, pp 46–57

Alamin MAA, Uddin G, Malakar S, Afroz S, Haider T, Iqbal A (2023) Developer discussion topics on the adoption and barriers of low code software development platforms. Empir Softw Eng 28(1):4

Alfayez R, Ding Y, Winn R, Alfayez G, Harman C, Boehm B (2023) What is asked about technical debt (td) on stack exchange question-and-answer (q &a) websites? an observational study. Empir Softw Eng 28(2):35

Ali S, Abuhmed T, El-Sappagh S, Muhammad K, Alonso-Moral JM, Confalonieri R, Guidotti R, Del Ser J, Díaz-Rodríguez N, Herrera F (2023) Explainable artificial intelligence (xai): What we know and what is left to attain trustworthy artificial intelligence. Inf Fusion 99:101805

Alicioglu G, Sun B (2022) A survey of visual analytics for explainable artificial intelligence methods. Comput Graph 102:502–520

Alshangiti M, Sapkota H, Murukannaiah PK, Liu X, Yu Q (2019) Why is developing machine learning applications challenging? a study on stack overflow posts. In: 2019 acm/ieee international symposium on empirical software engineering and measurement (esem). IEEE, pp 1–11

Angelov PP, Soares EA, Jiang R, Arnold NI, Atkinson PM (2021) Explainable artificial intelligence: an analytical review. Wiley Interdiscip Rev Data Min Knowl Disc 11(5):e1424

Arrieta AB, Díaz-Rodríguez N, Del Ser J, Bennetot A, Tabik S, Barbado A, García S, Gil-López S, Molina D, Benjamins R et al (2020) Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. Inf Fusion 58:82–115

Bagherzadeh M, Khatchadourian R (2019) Going big: a large-scale study on what big data developers ask. In: Proceedings of the 2019 27th ACM joint meeting on european software engineering conference and symposium on the foundations of software engineering, pp 432–442

Barua A, Thomas SW, Hassan AE (2014) What are developers talking about? an analysis of topics and trends in stack overflow. Empir Softw Eng 19:619–654

Bellamy RKE, Dey K, Hind M, Hoffman SC, Houde S, Kannan K, Lohia P, Martino J, Mehta S, Mojsilovic A, Nagar S, Ramamurthy KN, Richards J, Saha D, Sattigeri P, Singh M, Varshney KR, Zhang Y (2018) AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. https://arxiv.org/abs/1810.01943

Bhatt U, Xiang A, Sharma S, Weller A, Taly A, Jia Y, Ghosh J, Puri R, Moura JM, Eckersley P (2020) Explainable machine learning in deployment. In: Proceedings of the 2020 conference on fairness, accountability, and transparency, pp 648–657

Biecek P (2018) Dalex: Explainers for complex predictive models in r. J Mach Learn Res 19(84):1–5. http://jmlr.org/papers/v19/18-416.html

Braiek HB, Khomh F, Adams B (2018) The open-closed principle of modern machine learning frameworks. In: Proceedings of the 15th international conference on mining software repositories, pp 353–363

Bridge C (2011) Unstructured data and the 80 percent rule. http://www.clarabridge.com/default.aspx

Cao S, Sun X, Widyasari R, Lo D, Wu X, Bo L, Zhang J, Li B, Liu W, Wu D et al (2024) A systematic literature review on explainability for machine/deep learning-based software engineering research. arXiv:2401.14617

Chattopadhay A, Sarkar A, Howlader P, Balasubramanian VN (2018) Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In: 2018 IEEE Winter conference on applications of computer vision (WACV), IEEE

Chatzimparmpas A, Martins RM, Jusufi I, Kerren A (2020) A survey of surveys on the use of visualization for interpreting machine learning models. Inf Vis 19(3):207–233

Chen Z, Cao Y, Liu Y, Wang H, Xie T, Liu X (2020) A comprehensive study on challenges in deploying deep learning based software. In: Proceedings of the 28th ACM joint meeting on european software engineering conference and symposium on the foundations of software engineering, pp 750–762

Choo J, Liu S (2018) Visual analytics for explainable deep learning. IEEE Comput Graphics Appl 38(4):84–92

Croft R, Xie Y, Zahedi M, Babar MA, Treude C (2022) An empirical study of developers' discussions about security challenges of different programming languages. Empir Softw Eng 27:1–52

Danilevsky M, Qian K, Aharonov R, Katsis Y, Kawas B, Sen P (2020) A survey of the state of explainable ai for natural language processing. arXiv:2010.00711

Das A, Rad P (2020) Opportunities and challenges in explainable artificial intelligence (xai): A survey. arXiv:2006.11371

Data Science Stack Exchange (2023) Data science forum. Available: https://datascience.stackexchange.com, online; accessed 1-June-2023

de Bruijn H, Warnier M, Janssen M (2022) The perils and pitfalls of explainable ai: strategies for explaining algorithmic decision-making. Gov Inf Q 39(2):101666

De Lucia A, Di Penta M, Oliveto R, Panichella A, Panichella S (2014) Labeling source code with information retrieval methods: an empirical study. Empir Softw Eng 19:1383–1420

De Silva D, Alahakoon D (2022) An artificial intelligence life cycle: From conception to production. Patterns 3(6)

Ding W, Abdel-Basset M, Hawash H, Ali AM (2022) Explainability of artificial intelligence methods, applications and challenges: a comprehensive survey. Inf Sci 615:238–292

Dwivedi R, Dave D, Naik H, Singhal S, Omer R, Patel P, Qian B, Wen Z, Shah T, Morgan G et al (2023) Explainable ai (xai): core ideas, techniques, and solutions. ACM Comput Surv 55(9):1–33

Etaati L, Etaati L (2019) Azure databricks. Selecting the right architecture and tools for your project. Machine learning with microsoft technologies, pp 159–171

Felzmann H, Fosch-Villaronga E, Lutz C, Tamò-Larrieux A (2020) Towards transparency by design for artificial intelligence. Sci Eng Ethics 26(6):3333–3361

Fincher S, Tenenberg J (2005) Making sense of card sorting data. Expert Syst 22(3):89–93

Ghorbani A, Abid A, Zou J (2019) Interpretation of neural networks is fragile. In: Proceedings of the AAAI conference on artificial intelligence vol 33, pp 3681–3688

Goodman B, Flaxman S (2017) European union regulations on algorithmic decision-making and a right to explanation. AI Mag 38(3):50–57

Google Cloud (2022) Google cloud vertex ai: a unified platform for mlops. https://cloud.google.com/vertex-ai, [Online]. Available

Guidotti R, Monreale A, Ruggieri S, Turini F, Giannotti F, Pedreschi D (2018) A survey of methods for explaining black box models. ACM Comput Surv 51(5):1–42

Hagras H (2018) Toward human-understandable, explainable ai. Computer 51(9):28–36

Hall P, Gill N, Schmidt N (2019) Proposed guidelines for the responsible use of explainable machine learning. arXiv:1906.03533

Han J, Shihab E, Wan Z, Deng S, Xia X (2020) What do programmers discuss about deep learning frameworks. Empir Softw Eng 25:2694–2747

Hanif A, Beheshti A, Benatallah B, Zhang X, Habiba, Foo E, Shabani N, Shahabikargar M (2023) A comprehensive survey of explainable artificial intelligence (xai) methods: exploring transparency and interpretability. In: International conference on web information systems engineering. Springer, pp 915–925
Hanif A, Zhang X, Wood S (2021) A survey on explainable artificial intelligence techniques and challenges. In: 2021 IEEE 25th international enterprise distributed object computing workshop (EDOCW). IEEE, pp 81–89
Haque MU, Iwaya LH, Babar MA (2020) Challenges in docker development: a large-scale study using stack overflow. In: Proceedings of the 14th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), pp 1–11
Honnibal M, Montani I, Van Landeghem S, Boyd A (2020) spaCy: Industrial-strength natural language processing in python. https://doi.org/10.5281/zenodo.1212303
Jiang H, Yang K, Gao M, Zhang D, Ma H, Qian W (2019) An interpretable ensemble deep learning model for diabetic retinopathy disease classification. In: 2019 41st annual international conference of the IEEE engineering in medicine and biology society (EMBC). IEEE, pp 2045–2048
Kendall MG (1938) A new measure of rank correlation. Biometrika 30(1/2):81–93
Kruskal WH (1957) Historical notes on the wilcoxon unpaired two-sample test. J Am Stat Assoc 52(279):356–360
Kubiak AP, Kawalec P (2022) Prior information in frequentist research designs: the case of neyman's sampling theory. J Gen Philos Sci 53(4):381–402
Lauriola M, Litman JA, Mussel P, De Santis R, Crowson HM, Hoffman RR (2015) Epistemic curiosity and self-regulation. Personality Individ Differ 83:202–207
LeDell E, Poirier S (2020) H2o automl: scalable automatic machine learning. In: Proceedings of the AutoML Workshop at ICML, ICML, vol 2020
Lee J, Davari H, Singh J, Pandhare V (2018) Industrial artificial intelligence for industry 4.0-based manufacturing systems. Manuf Lett 18:20–23
Lievens F, Harrison SH, Mussel P, Litman JA (2022) Killing the cat? a review of curiosity at work. Acad Manag Ann 16(1):179–216
Li H, Khomh F, Openja M et al (2021) Understanding quantum software engineering challenges an empirical study on stack exchange forums and github issues. In: 2021 IEEE International Conference on Software Maintenance and Evolution (ICSME). IEEE, pp 343–354
Litman J (2019) Curiosity: Nature, dimensionality, and determinants. In: Renninger KA, Hidi SE (eds) The cambridge handbook of motivation and learning. Cambridge University Press, pp 418–442. https://doi.org/10.1017/9781316823279.019
Loper E, Bird S (2002) Nltk: The natural language toolkit. arXiv preprint cs/0205028
Lundberg SM, Lee SI (2017) A unified approach to interpreting model predictions. Adv Neural Inf Process Syst 30
Ma Y, Bogart C, Amreen S, Zaretzki R, Mockus A (2019) World of code: an infrastructure for mining the universe of open source vcs data. In: 2019 IEEE/ACM 16th International Conference on Mining Software Repositories (MSR). IEEE, pp 143–154
Mahmood K, Rasool G, Sabir F, Athar A (2023) An empirical study of web services topics in web developer discussions on stack overflow. IEEE Access 11:9627–9655
Malgieri G, Comandé G (2017) Why a right to legibility of automated decision-making exists in the general data protection regulation. Int Data Priv Law 7(4):243–265
Malhotra R (2016) Empirical research in software engineering: concepts, analysis, and applications. CRC Press
McCallum AK (2002) Mallet: A machine learning for languagetoolkit. http://mallet.cs.umass.edu
McHugh ML (2012) Interrater reliability: the kappa statistic. Biochem Med 22(3):276–282
Mersha M, Lam K, Wood J, AlShami A, Kalita J (2024) Explainable artificial intelligence: a survey of needs, techniques, applications, and future direction. Neurocomputing 128111
Michael Ayas H, Leitner P, Hebig R (2023) An empirical study of the systemic and technical migration towards microservices. Empir Softw Eng 28(4):85
Minh D, Wang HX, Li YF, Nguyen TN (2022) Explainable artificial intelligence: a comprehensive review. Artif Intell Rev 1–66
Nagahisarchoghaei M, Nur N, Cummins L, Nur N, Karimi MM, Nandanwar S, Bhattacharyya S, Rahimi S (2023) An empirical survey on explainable ai technologies: recent trends, use-cases, and categories from technical and application perspectives. Electronics 12(5):1092
Naghashzadeh M, Haghshenas A, Sami A, Lo D (2021) How do users answer matlab questions on q &a sites? a case study on stack overflow and mathworks. 2021 IEEE International Conference on Software Analysis. Evolution and Reengineering (SANER), IEEE, pp 526–530
Neyman J (1938) Contribution to the theory of sampling human populations. J Am Stat Assoc 33(201):101–116

Openja M, Adams B, Khomh F (2020) Analysis of modern release engineering topics:–a large-scale study using stackoverflow–. In: 2020 IEEE international conference on software maintenance and evolution (ICSME). IEEE, pp 104–114
Ralph P, Ali Nb, Baltes S, Bianculli D, Diaz J, Dittrich Y, Ernst N, Felderer M, Feldt R, Filieri A et al (2020) Empirical standards for software engineering research. arXiv:2010.03525
Reddy S, Allan S, Coghlan S, Cooper P (2020) A governance model for the application of ai in health care. J Am Med Inform Assoc 27(3):491–497
Řehůřek R, Sojka P (2010) Software Framework for Topic Modelling with Large Corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, ELRA, Valletta. Malta, pp 45–50, http://is.muni.cz/publication/884893/en
Reiter E (2019) Natural language generation challenges for explainable ai. arXiv:1911.08794
Ribeiro MT, Singh S, Guestrin C (2016) Why should i trust you? explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp 1135–1144
Rosen C, Shihab E (2016) What are mobile developers asking about? a large scale study using stack overflow. Empir Softw Eng 21:1192–1223
Saeed W, Omlin C (2023) Explainable ai (xai): a systematic meta-survey of current challenges and future opportunities. Knowl-Based Syst 263:110273
Sahakyan M, Aung Z, Rahwan T (2021) Explainable artificial intelligence for tabular data: a survey. IEEE Access 9:135392–135422
Samek W, Müller KR (2019) Towards explainable artificial intelligence. Explainable AI: interpreting, explaining and visualizing deep learning pp 5–22
Samek W, Wiegand T, Müller KR (2017) Explainable artificial intelligence: understanding, visualizing and interpreting deep learning models. arXiv:1708.08296
Sayyadnejad MM (2024) Exploring the black box: analyzing explainable AI challenges and best practices through stack exchange discussions. https://doi.org/10.5281/zenodo.11002960
Schwalbe G, Finzel B (2024) A comprehensive taxonomy for explainable artificial intelligence: a systematic survey of surveys on methods and concepts. Data Min Knowl Disc 38(5):3043–3101
Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2017) Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision, pp 618–626
Sengupta S, Haythornthwaite C (2020) Learning with comments: an analysis of comments and community on stack overflow. Proceedings of the Annual Hawaii International Conference on System Sciences https://doi.org/10.24251/hicss.2020.354
Simonyan K, Vedaldi A, Zisserman A (2013) Deep inside convolutional networks: visualising image classification models and saliency maps. arXiv:1312.6034
Son J, Kim B (2023) Trend analysis of large language models through a developer community: A focus on stack overflow. Information 14(11):602
Speith T (2022) A review of taxonomies of explainable artificial intelligence (xai) methods. In: Proceedings of the 2022 ACM conference on fairness, accountability, and transparency, pp 2239–2250
Springer A, Whittaker S (2019) Progressive disclosure: empirically motivated approaches to designing effective transparency. In: Proceedings of the 24th international conference on intelligent user interfaces, pp 107–120
Stack Exchange Data Explorer (2023) Stack exchange data explorer. Available: http://data.stackexchange.com, online; Accessed 1-June-2023
Stack Overflow (2023) Stack overflow forum. Available: https://stackoverflow.com, online; Accessed 1-June-2023
Tahir A, Dietrich J, Counsell S, Licorish S, Yamashita A (2020) A large scale study on how developers discuss code smells and anti-pattern in stack exchange sites. Inf Softw Technol 125:106333
Tahir A, Yamashita A, Licorish S, Dietrich J, Counsell S (2018) Can you tell me if it smells? a study on how developers discuss code smells and anti-patterns in stack overflow. In: Proceedings of the 22nd International conference on evaluation and assessment in software engineering 2018, pp 68–78
Tan X, Gao K, Zhou M, Zhang L (2022) An exploratory study of deep learning supply chain. In: Proceedings of the 44th international conference on software engineering, pp 86–98
TeamHG-Memex (2019) Eli5: Explain like i'm 5 [documentation]. https://eli5.readthedocs.io/, [Online;]
Tjoa E, Guan C (2020) A survey on explainable artificial intelligence (xai): toward medical xai. IEEE Trans Neural Netw Learn Syst 32(11):4793–4813
Treude C, Barzilay O, Storey MA (2011) How do programmers ask and answer questions on the web?(nier track). In: Proceedings of the 33rd international conference on software engineering, pp 804–807
Treviso MV, Martins AF (2020) The explanation game: Towards prediction explainability through sparse communication. arXiv:2004.13876

Uddin G, Baysal O, Guerrouj L, Khomh F (2019) Understanding how and why developers seek and analyze api-related opinions. IEEE Trans Software Eng 47(4):694–735
Uddin G, Sabir F, Guéhéneuc YG, Alam O, Khomh F (2021) An empirical study of iot topics in iot developer discussions on stack overflow. Empir Softw Eng 26:1–45
Vilone G, Longo L (2021) Classification of explainable artificial intelligence methods through their output formats. Mach Learn Knowl Extr 3(3):615–661
Wan Z, Xia X, Hassan AE (2019) What do programmers discuss about blockchain? a case study on the use of balanced lda and the reference architecture of a domain to capture online discussions about blockchain platforms across stack exchange communities. IEEE Trans Software Eng 47(7):1331–1349
Wang C, Chen Z, Zhou M (2023) Automl from software engineering perspective: landscapes and challenges. In: Proceedings of the 20th International Conference on Mining Software Repositories, MSR
Wang S, Zhou T, Bilmes J (2019) Bias also matters: Bias attribution for deep neural network explanation. In: International conference on machine learning. PMLR, pp 6659–6667
Watson DS (2022) Conceptual challenges for interpretable machine learning. Synthese 200(2):65
Woolson RF, Bean JA, Rojas PB (1986) Sample size for case-control studies using cochran's statistic. Biometrics 927–932
Yang XL, Lo D, Xia X, Wan ZY, Sun JL (2016) What security questions do developers ask? a large-scale study of stack overflow posts. J Comput Sci Technol 31:910–924
Yang W, Wei Y, Wei H, Chen Y, Huang G, Li X, Li R, Yao N, Wang X, Gu X et al (2023) Survey on explainable ai: From approaches, limitations and applications aspects. Human-Centric Intelligent Systems 3(3):161–188
Zhang QS, Zhu SC (2018) Visual interpretability for deep learning: a survey. Front Inf Technol Electron Eng 19(1):27–39

Mohammad Mahdi Sayyadnejad is a Researcher in the Department of Computer Science and Engineering and IT at Shiraz University (Shiraz, Iran) and an IT Expert at Kharazmi University (Tehran, Iran). He holds a Master's degree in Software Engineering from Shiraz University and a Bachelor's degree in Computer Engineering from Islamic Azad University. His research interests include software engineering, empirical studies, mining software repositories, explainable artificial intelligence, and AI ethics.

Ali Asgari is a PhD researcher in the Software Engineering Research Group at Delft University of Technology and a member of the AI4SE Research Group, a collaboration between TU Delft and JetBrains dedicated to advancing software engineering through artificial intelligence. He holds a Master's degree in Data Science from the University of Naples Federico II, Italy, and is an alumnus of the Apple Developer Academy, a joint initiative of Apple and the University of Naples. His research focuses on the intersection of software engineering and AI, with an emphasis on empirical studies and experimental approaches.

Ashkan Sami is a Professor of Computer Science (Software Engineering) at the Computer Science subject group at Edinburgh Napier University. His current research focuses on quality aspects of AI, especially GenAI systems, and the use of foundational models to tackle societal challenges across a broad spectrum of disciplines, including Software Engineering, medicine, and engineering. Throughout his academic journey, Ashkan has engaged in a wide range of interdisciplinary and transdisciplinary research, striving to contribute meaningfully to these areas. His efforts have led to impactful solutions for societal challenges and publications in highly respected journals and conferences. He has received several national awards for industrial and applied research, acknowledging his contributions to the field. Ashkan is the founder and co-lead of the System and Software Engineering research theme at the Scottish Informatics and Computer Science Alliance (SICSA), a collaboration of 14 Scottish universities and three Scottish Innovation Centres since 2024. From 2015 to 2020, he had his own startup company. His work has gained media attention, including features on the BBC, The Register, and Stack Overflow team blogs, as well as international news outlets in China, Japan, Germany, and Spain. Ashkan earned his Ph.D. from Tohoku University in Japan in 2006, which led to a significant research grant from MEXT, where he became an assistant professor at the university. His educational background also includes an M.Sc. in AI and Robotics from Shiraz University (1996) and a B.Sc. from Virginia Tech (1991).

Hooman Tahayori received his Ph.D. degree in Informatics from Universita Degli Studi di Milano, Italy in 2009. He completed his Honors master degree in Robotics and Artificial Intelligence and his Bachelor's degree with Honors in Computer Science-Hardware design respectively in 2000 and 1997 at Shiraz University. Currently he is an associate professor at the Department of Computer Science & Engineering and Information Technology, Shiraz University, Shiraz, Iran. His main areas of research interests are fuzzy sets of higher order, perceptual computing, granular computing and their applications in various disciplines. He has published over 70 technical papers in the areas of type-2 fuzzy sets, shadowed sets, granular and perceptual computing.