Titre: Title:	Towards a Reference Architecture of AI-Based Job Interview Systems
Auteur: Author:	Maryam Abedi
Date:	2022
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Abedi, M. (2022). Towards a Reference Architecture of AI-Based Job Interview Systems [Master's thesis, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/10215/

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Towards a Reference Architecture of AI-based Job Interview Systems

MARYAM ABEDI
Département de génie informatique et génie logiciel

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
Génie informatique

Janvier 2022

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Ce mémoire intitulé :

Towards a Reference Architecture of AI-based Job Interview Systems

présenté par **Maryam ABEDI**
en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
a été dûment accepté par le jury d'examen constitué de :

Marios-Eleftherios FOKAEFS, président
Jinghui CHENG, membre et directeur de recherche
Bram ADAMS, membre et codirecteur de recherche
Foutse KHOMH, membre

DEDICATION

To my beloved ones;
To my mom and dad with a special feeling of gratitude,
To my husband, for his enduring love,
To my little son, with a love to the moon and back,
To my best friend, Elahé, for her endless support and encouragement.

ACKNOWLEDGEMENTS

I would like to specially thank my supervisors, Prof. Bram Adams and Prof.Jinghui Cheng, for providing consistent support, guidance and feedback throughout this thesis project.

Further, I would like to express my sincere gratitude to Jan Rockemann and Amanda Arciero from Airudi Inc. for funding this research study as well as Mohammad Mehdi Morovati and Dr.Mohammed Yassin for all the considerate guidance and help.

RÉSUMÉ

Les entretiens d'embauche sont une activité majeure dans le processus de recrutement, mais ils impliquent un effort manuel considérable et, en tant que tels, sont une tâche coûteuse et chronophage. Afin de réduire ce coût, les organisations envisagent de plus en plus l'utilisation de l'IA pour automatiser leur processus de recrutement, ce qui promet des entretiens d'embauche plus efficaces et moins biaisés, tout en réduisant les coûts du processus d'embauche. Alors que la recherche et les produits existants sont apparus en se concentrant sur l'automatisation des étapes individuelles du processus d'entretien d'embauche, l'architecture globale d'un système logiciel pour l'automatisation des entretiens d'embauche semble encore décourageante et les compromis en termes d'exigences fonctionnelles ne sont pas clairs.

À cette fin, soutenus par une revue systématique de la littérature (SLR), nous proposons et documentons la structure statique et dynamique d'une architecture de référence complète (RA) pour les systèmes d'automatisation des entretiens d'embauche basés sur l'IA du point de vue fonctionnel. Nous avons identifié quatre composants de haut niveau qui doivent être traités dans un système d'entretien d'embauche basé sur l'IA, avec leurs fonctionnalités et leurs responsabilités. Afin d'automatiser un bon nombre de ces fonctionnalités, les capacités de l'IA peuvent être exploitées à l'aide des méthodes étudiées dans la littérature. Nous avons ensuite discuté des défis et des problèmes ouverts et des hypothèses invalides dans la littérature existante pour concevoir et mettre en œuvre les fonctionnalités des composants extraits.

Cependant, les exigences fonctionnelles que nous avons identifiées pour chaque composante du RA sont basées exclusivement sur les connaissances que nous avons acquises grâce aux SLR sans que la perspective de la principale partie prenante, c'est-à-dire les recruteurs et les personnes interrogées, soit prise en compte. Afin d'éviter toute conclusion biaisée, nous avons mené une étude empirique pour raffiner objectivement les exigences fonctionnelles identifiées ainsi que pour obtenir de nouvelles connaissances sur les exigences fonctionnelles requises des composants de l'AR pour un système d'automatisation des entretiens d'embauche basé sur l'IA et basé sur des observations réelles.

Pour effectuer cette analyse empirique, nous avons appliqué la modélisation thématique sur un ensemble de données de 10 ans de questions d'utilisateurs sur le recrutement et les entretiens d'embauche qui ont été partagées sur le site Workplace de Stack Exchange. Nous avons ensuite étiqueté chaque sujet pour qu'il soit représentatif des exigences fonctionnelles atten-

dues des utilisateurs potentiels du système d'automatisation des entretiens d'embauche basé sur l'IA. Ensuite, en appliquant la technique d'inférence de sujet, nous avons lié chaque sujet à l'un des composants dérivés de l'architecture de référence proposée (RA). Enfin, en utilisant ces connaissances (c'est-à-dire l'étiquette des sujets et leurs composants correspondants) ainsi que les mesures que nous avons formulées à l'aide des données météorologiques disponibles dans l'ensemble de données, nous avons mesuré la popularité et le défi de chaque sujet et avons dérivé ce que le sujet peut nous apprendre sur les composant(s) correspondant(s) de l'architecture de référence.

Nous avons constaté que certains sujets et les fonctionnalités de leurs composants correspondants devraient être à la fois difficiles et intéressants à ajouter ou à prioriser lors de la conception du RA. Les exigences liées à la réduction des dépenses liées aux entretiens d'embauche font partie de ces fonctionnalités.

Nous avons également appris que la plupart des fonctionnalités intéressantes qui devraient être incluses/améliorées sont liées au composant responsable de l'établissement d'une interaction directe entre le système et la personne interrogée (par exemple, la reconnaissance de l'apparence physique, la détection d'indices comportementaux, le suivi contacts entre les parties). De même, bien que certaines fonctionnalités connexes de ce composant seraient également difficiles à inclure/améliorer, les fonctionnalités les plus difficiles à concevoir/améliorer sont liées à un autre composant qui est chargé d'analyser la discussion de l'entretien et d'évaluer le candidat.

ABSTRACT

Job interviews are a major activity in the recruitment process, yet they involve substantial manual effort and as such are an expensive and time-consuming task. In order to reduce this cost, organizations are increasingly considering the use of AI to automate their recruitment process, which brings the promise of more effective, less biased job interviews, while reducing the costs of the hiring process. While existing research and products have appeared focusing on automating individual steps of the job interview process, the overall architecture of a software system for job interview automation seems still daunting and the trade-offs in terms of functional requirements are unclear.

To this end, supported by a Systematic Literature Review (SLR), we propose and document the static and dynamic structure of a comprehensive reference architecture (RA), focusing on the functional viewpoint, for AI-based job interview automation systems. We identified four high-level components that need to be addressed in an AI-based job interview system, with their functionalities and responsibilities.Many of these functionalities can be can be automated with AI capabilities using the methods that are surveyed in the literature. We then discussed the challenges and open issue and invalid assumptions in the existing literature for designing and implementing the functionalities of the extracted components.

However, the functional requirements we identified for each component of the RA is based exclusively on knowledge we gained through the SLRs, lacking the perspective of the main stakeholder, i.e., the recruiters and interviewees. In order to avoid making any biased conclusion, we conducted an empirical study to refine the identified functional requirements objectively as well as to obtain new knowledge about additional required functional requirements of the RA's components for an AI-based job interview automation system based on real observations.

To do this empirical analysis, we applied topic modeling on a dataset of 10 years of user-contributed questions about recruitment and job interviews that have been shared in the Workplace site of Stack Exchange. We then, labeled each topic to be a representative of the expected functional requirements of the potential users of AI-based job interview automation system. Next, applying the topic inference technique, we linked each topic to any of the derived component of the proposed reference architecture (RA). Finally, using this knowledge (i.e., label of topics and their corresponding components) along with metrics we formulated using the available mete-data in the dataset , we measured how popular and challenging each topic is and derived what the topic can learn us about the corresponding component(s) of

the reference architecture.

We found that, some topics and the functionalities of their corresponding component(s) are expected to be both difficult and interesting to be added or to be prioritised while designing the RA. The requirements related to lowering job interview expenses is notable example.

We also learnt that most of the interesting functionalities that are expected to be included/improved are related to the component that is responsible for establishing direct interaction between the system and the interviewee (e.g., physical appearance recognition, behavioural clue detection, making follow-up contacts between the parties). Likewise, although some related functionalities of this component were also found to be challenging to be included/improved, the most difficult functionalities to be designed/improved are related to another component that is responsible for analysing the interview discussion and evaluating the candidate.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1 INTRODUCTION

In this study, we first focused on proposing a reference architecture for an AI-based Job interview system from the functional view point, supported by findings of a systematic literature review. Then, in order to refine our claimed knowledge about the identified functional requirements and recover any missing functional requirements, we explored the related concerns of roles explicitly involved in the recruitment and job interview pipeline, i.e., as the potential users of an AI-based Job interview system.

In the next section, we will first motivate why an AI-based Job interview system is required. Then, we will define the problems that will be addressed by a Reference Architecture for such a system. Finally, we discuss why understanding the relationship between the concerns of job market stakeholders and the proposed reference architecture is essential to improve the design of such system.

To structure our research, we formulated several RQs that will be addressed in the thesis. Next, we will explain the methodologies that we will follow to address the RQs. Finally, this chapter will finish by presenting the structure of this thesis.

1.1 Problem definition

1.1.1 The necessity of an AI-based Job interview system

The essence of an organization lies in the experience, skills and intellectual capabilities of its human resources [1]. Efficient recruitment processes allow organizations to succeed in achieving the desired objectives. According to [2], *"recruitment includes those practices and activities carried on by the organization with the primary purpose of identifying and attracting potential employees"*. These activities might include developing and advertising the job description, reviewing applicants' resumes and filtering out the ones best matched with the job description, scheduling and conducting screening and technical interviewees, the result evaluation and generating reports for the recruiters, offering the job to the selected candidate and finally his engagement and joining the team. (Figure 1.1). Potentially, all of these tasks can be revolutionized with artificial intelligence capabilities. As one of the essential tasks in the talent hiring pipeline, interviewing forms a direct bridge between candidates and recruiters in fitting the right person for the right job.

The recruitment interview, as the focus of this study, has been defined as *"A face-to-face interaction conducted to determine the qualifications of a given individual for a particular open*

position" [3]. Conventionally, it is performed through face-to-face meetings or phone/video calls and involves third parties, like HR staff and direct managers to do the interviews and analyze the result. The interviewees would be the applicants who have been previously shortlisted based on their resumes. Almost all organizations conduct interview sessions during applicant selection [4], since it is one of the most effective techniques to evaluate technical capabilities, problem-solving ability and communication skills.

However, like other practices in the conventional recruitment pipeline, the traditional human-driven interview process is costly, time-consuming and requires substantial effort. For example, as reported by [5], in the United States companies nowadays have to pay over 4000 dollars averagely per candidate for selecting a capable candidate, the process of job interview usually taking about 24 days. This process is also subject to potential biases due to the subjective nature of the process [6] and [7]. Especially since the different interviewers have different technical backgrounds and experiences, a biased or incomplete assessment of job applicants is not uncommon [8]. This situation is even more challenging for big companies that usually receive huge number of applicants for every job opening [9].

These challenges have led to the idea of "electronic recruitment" (e-recruiting) systems that combine online communication mechanisms (for interviews) with artificial intelligence in order to avoid adverse effects of human bias through system standardization while yielding a cost and time-effective approach. The recent advances in AI offer new ways to recruit talents while reducing human workloads [10], and can provide a competitive environment in a tough market of job for skills by providing a better talent management capability [11]. There are other motivations and advantages that convince companies to introduce an e-recruiting system, as surveyed in [12]. Also, [13] discussed how the technology acts as an enabler for performing recruitment more effectively, efficiently and scalably through cloud based services, decision making systems, distributed teams, and online talent management across the organization. The efficient use of e-recruitment is said to directly lead to a significant change in the traditional recruitment process [14].

One major pillar step in the e-recruitment process are the new communication platforms that have come up to substitute face-to-face interviews. For example, today it is a common practice for organizations to screen applicants through Skype interviews or asynchronous video interviewing (AVI) platforms [15], [16] and [17]. The pace of adoption of such technologies only has been expedited by the COVID-19 and pandemic situation expedites this trend; As a recent Gartner poll[1] showed, 48 percent of employees will likely work remotely at least part

[1]https://www.gartner.com/en/newsroom/press-releases/2020-04-14-gartner-hr-survey-reveals-41–of-employees-likely-to-

of the time after COVID-19 versus 30 percent before the pandemic. It means that organizations shift to more remote work operations and peruse to hire employees from different geographical locations to overcome lack of skills locally. On the other hand, job seekers also benefit from this trend; they can apply for their desired jobs while they do not need to leave their cities, and are guaranteed to perform their interviews in a safe (online) environment.
The second pillar of e-recruiting processes, AI, offers potential to support automation of large chunks of the job interview process. This includes predictive behavior technologies to understand human behavior and talent analytics, technologies and tools to extract and evaluate the data to automatically assess the applicant's hire-ability, and also techniques to visualize the interview sessions in a virtual environment to automate interview processes (e.g., [18] and [19]). Indeed, AI-based interview approaches can significantly revolutionize job interview practice by reducing manual tasks, the time invested into the process and third-party involvement, while preventing subjective evaluations [20]. This thesis focuses on the job-interview related steps of e-recruitment (Figure 1.1)

1.1.2 The necessity of proposing a Reference Architecture for an AI-based Job interview system

This study aims to derive the reference architecture for AI-based job interview systems using the knowledge provide through a systematic literature review (SLR). Nakagawa et al. define an RA for software systems as "an architecture that encompasses the knowledge about how to design concrete architectures for systems of a given application domain; therefore, it must address the business rules, architectural styles (sometimes also defined as architectural patterns that address quality attributes in the SRA), best practices of software development (for instance, architectural decisions, domain constraints, legislation, and standards), and the software elements that support development of systems for that domain. All of this must be supported by a unified, unambiguous, and widely understood domain terminology" [21]. This definition is in line with that of other work [22], and [23]. Reference architectures (RA) provide a frame of reference that helps to get an overview of a domain and provide a starting point and a template solution for an enterprise architecture effort for a domain. Following this approach, methods and decisions can be changed inside each one of those components as independent "development islands," without the need for a change in the other ones. An RA is a high-level abstraction of software system that must conform to the main functional requirements of the system's domain. An RA also describes its software elements and the relationships among them. The motivations behind preparing an RA are, to name but a few, improving interoperability of software systems, reducing the developments costs, improving communication among stakeholders, reducing risk due to previous experiences, promoting

PROCESS.JPG PROCESS.JPG PROCESS.JPG PROCESS.JPG

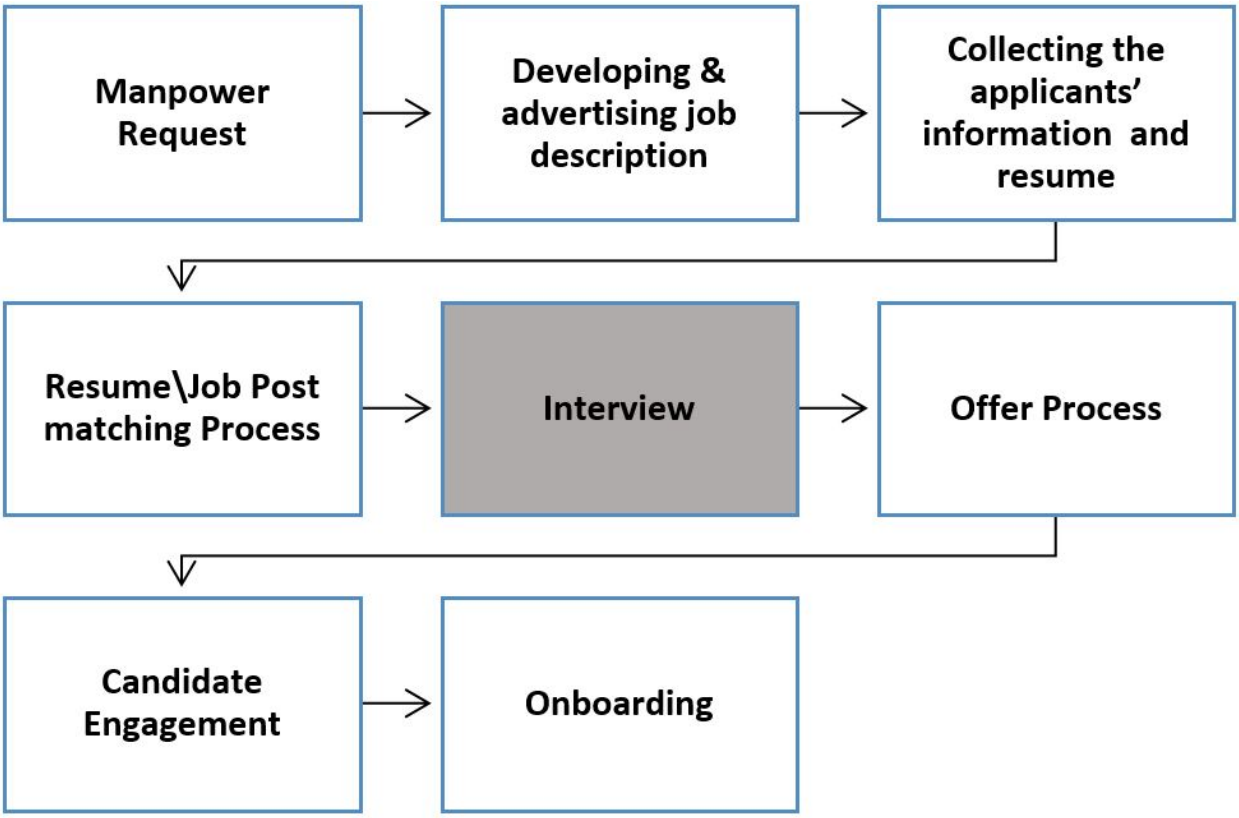

Figure 1.1 Recruitment/E-Recruitment Process

best practices, and reducing time-to-market [24].

Note that, there is a subtle but critical difference between reference architecture and a regular architecture. In fact, a reference architectures is used as a basis for creating a architectures at production level that are more specialised and focus on a specific software system in a domain [25]. However, reference architectures are basically high level software architectures where the structures and respective components and the interaction among them provide templates (often based on the generalization of a set of solutions) for concrete architectures for a particular domain. Reference architectures deal with the range of knowledge of an application domain, providing solutions and templates for a broader domain. Therefore it can be said that architectures are more specialised and detailed-oriented than reference architectures. Thus, essentially we need to first propose the reference architectures as a basis for architectures at the production level for a software system.

In our reference architecture we only focus on the static structure and dynamic aspect of the *functional viewpoint*. The functional viewpoint introduces the functional elements of the system and their responsibilities as well as its dynamic behaviour. This is similar to other work, for example [26] focused only on 'functional view' of the architecture description, which is the *"specification of the intended functions and their interactions necessary to achieve the desired behavior"*. This means that the RA focuses on the logical decomposition of a system into components and sub-components and the data-flows between them for their proposed model. Therefore, although during the SLRs the applied methods to automate the identified functionalities are surveyed, we do not provide a reference for the actual technical implementation of the functional requirements of the components in terms of required hardware or methods and techniques, since that is not at the same high level of abstraction as an RA.

1.1.3 The necessity of understanding the relationship between the concerns of job market stakeholders and the proposed reference architecture

Investigating people's opinion is a common approach that researchers in many domains follow to obtain a better insights of scientific topics based on reality, avoiding bias by the researcher. Particularly, in the job interview context, we need to use this approach to explore the concerns of the major roles involved in the interview process and link them to those derived components of our proposed reference architecture in order to both prioritize and possibly extend additional requirements of our RA . For this, we are interested in a data driven approach, analyzing discussions of recruitment stakeholders about their concerns and major issues involving the job interview process. This would avoid the disadvantages of applying other mechanisms for qualitative data collection such as survey and questionnaire (e.g., bi-

ased and dishonest answers or costs for setting up the experiments), and would provide access to an extensive amount of data across a long period to be analyzed.

Apart from linking the obtained knowledge to the functionalities of the derived components of our proposed reference architecture, the achieved results will also reveal the popular and challenging issues and concerns among stakeholders in this context. Based on these findings we can learn what would be the expectations of the end-users of an AI-based Job Interview Automation Systems; what components and which of their functionalities would be more appealing or difficult to design based on the discussion they had about the recruitment and in particular the job interview process. These outcomes should be considered while designing and implementing the corresponding components of an AI-based job interview system based on proposed reference architecture.

1.2 Objectives of the thesis (RQs)

In this study, we aim to propose a reference architecture (RA) for AI-based job interview automation systems. To meet this objective, we will derive its components' main sub-components and their functionalities. We will also explore and discuss the challenges in designing and developing different components of the proposed reference architecture.

1.2.1 Objectives of proposing Reference Architecture for an AI-based Job interview system

By creating this reference architecture, we aimed to answer the following two research questions:

RQ1: What are the high-level architectural components required of an AI-based job interview pipeline?

By answering this question the high-level components of AI-based job interview systems are extracted.

RQ2: What are the detailed architectural sub-components of an AI-based job interview pipeline?

To answer **RQ1**, a Systematic Literature Review of research publications is conducted to identify the major components required for an AI-based job interview system, in a systematic fashion. After deriving the high-level reference architecture, for each of the identified high-level architectural components a separate follow-up SLR in more detail are conducted to propose the reference architecture of each components (answer of **RQ2**). Through these

literature studies the AI-based and non AI-based methods and technologies that have been used in the literature to automate those functionalities are also identified. As we will see, not for all components AI techniques have been applied yet or found successful. Also, for each component, the invalid assumptions of prior studies and challenges in this context are analyzed to open future research windows. To our knowledge, this work is the first study that focuses on a comprehensive view of an AI-based job interview system. An overview of the approach is illustrated in Figure 3.1 in Chapter 3.

1.2.2 Objectives of understanding the relationship between the concerns of job market stakeholders and the proposed reference architecture

Through an empirical study, the following two research questions will be answered. We will conduct this empirical study by applying topic modeling and topic inference techniques to analyze the opinion of the major roles involved in the job recruitment process that we will collect from a popular on-line career-related forum. The findings will lead us to understand the concerns of the involving roles and link them to the derived components of our proposed reference architecture. We will also find the popular and challenging issues and concerns among stakeholders in this context, through data driven approaches. Based on these findings, we will discuss and conclude what would be the expectations of the actual users of an AI-based Job Interview Automation System, as well as what components and which of their functionalities would be more appealing or difficult to design based on the opinions that the potential end-users had shared about the recruitment, and in particular the job interview, process.

RQ3: What topics are discussed among stakeholders about the Recruitment and Job interview process? The goal of this research question is to validate if there is any link between the main concerns that different stakeholders (including job applicants and recruiters throughout the recruitment process) and the different components of the reference architecture proposed for automation of the job interview pipeline. To answer this question, we applied topic modeling technique on the posts that the analyzed roles have shared in the Workplace site of the Stack Exchange forum, from 2011 to 2021.

RQ4: Which component(s) of our reference architecture and which of its functional requirements are more interesting from the points of view of stakeholders (including job applicants and recruiters) throughout the recruitment process?

The goal of this question is to identify the functional requirements for each component that have the priority from stockholders' point of view. So that, one can then prioritize those when building the architecture's components. Also, by answering this question, one can learn which

are the required functionalities of each component that are missing in the proposed reference architecture. To answer this question, using 3 parts of meta data (i.e., View Count, Score and Favorite Count of posts), we defined 3 metrics as indicators of the popularity of a post to find out whether people are more attracted to certain shared posts than others. Then, having known the corresponding component of the proposed reference architecture for each topic (i.e., output of the topic inference analysis) as well as the labels that the labeling team have assigned to each topic as description of a functional or non-functional requirement, we can understand which component(s) and which of their functionalities would be more interesting from the public's points of view to include or improve first.

RQ5: Which component(s) of our reference architecture and which of its functionalities are more challenging to design from the points of view of stakeholders?

The goal of this question is to identify the topics that may be challenging for different stakeholders including job applicants and recruiters throughout the recruitment process to address. By answering this question, one can learn the functionalities (and hence which components) that require more effort when designing and implementing the components of the proposed reference architecture in order to satisfy the major stakeholders.

To answer this question, we defined two metrics using three parts of meta-data of posts (i.e., having or not-having the ID of an accepted answers (Boolean value) and creation and closing date of the post) and calculate them for each group of posts (i.e., topics). Again, having known the corresponding component of the proposed reference architecture for each topic (i.e., output of the topic inference analysis) as well as the labels that the labeling team have assigned to each topic as description of a functional requirement, we can identify which component(s) and which of their functionalities are harder to automate from the involving stakeholders point of views.

1.3 Thesis structure

This thesis is organized as follows: In chapter 2 the most related publications to our work are reviewed and categorised by topic. Next, in chapter 3, the reference architecture (RA) for an AI-based job interview system is proposed. Also, the systematic literature review (SLR) that is conducted to support this reference architecture is provided in this chapter. The methodology for conducting the SLR and documentation of the RA, is explained in that chapter too.

In chapter 4, we explain the methodology for conducting an empirical study on discussion posts about job recruitment in order to understand the concerns of the major roles involved

in recruitment and job interview, followed by establishing a link between these concerns to the derived components and their (non-)functionalities of our reference architecture.

Finally, this thesis is concluded in chapter 5, by providing a summery of this study and its limitations and opening windows for future studies.

CHAPTER 2 LITERATURE REVIEW

2.1 Applying reference architecture

Many published papers have proposed reference architectures for a variety of domains. For example, in the dialogue system domain, having briefly covered cognitive aspects of chatbots through 2 case studies, [27] proposed a reference architecture for chatbot applications across domains that correspond to our virtual interface component. In the same field, after exploring the gaps in the literature, [28] proposed a client-server architectural style framework particularly for their proposed chatbot for a dialogue composition named MyUBot. Likewise, [29] proposed a two-layer architectural framework of a multi-modal dialogue system.

In addition, in the IoT context, [30] did a short literature review, then explained the important components for information integration for smart city security profiling using a machine learning approach through an architectural framework. In the same context, in order to apply a reference architecture to describe IoT-based logistic information systems in agri-food supply chains, [31] first defined requirements for such systems by analyzing specific characteristics of the agri-food supply chain and specific application scenarios. Their architecture is abstracted from two case studies in different sub-sectors.

Furthermore, in health-related fields, [32] provided a reference architecture for enabling AI-based personalization and self-adaptation of mobile apps for e-Health. Furthermore, after a short analysis of the related works and identifying the gaps, [33] proposed a reference architecture for an intelligent platform to personalizing the patient management.

The proposed architectural framework for intelligent CO2 capturing systems by [34] is another illustration of the applicability of architectural modeling in different domains. They built the basis for their reference architecture through a brief review of the relevant literature. We followed a similar SLR approach as the above works, focusing on the domain of AI-based job interview systems.

2.2 Literature surveys to support the proposed models

In this section, we also present related literature survey papers and explain how our work distinguishes from them. These comprise papers that survey publications that could be applied to automating some of the phases of a job interview automation system, yet they are conducted in general terms or in domains other than the job interview context. Below we

provide a shortlist of them.

- Literature survey on questions generation
  There are several conducted efforts to generate questions from a given corpus automatically by applying different algorithms as surveyed in [35]. In addition, Rao et al. surveyed a list of techniques found in the literature for automatic multiple choice question (MCQ) generation from a text [36]. Likewise, [37] provided an extensive survey on the recent advancements in the application of NN techniques in question generation. Automatic question generation (AQG) is also applicable for education purposes. Thus, Kurdi et al. conducted a systematic review on AQG findings in the education area [38].

- Literature survey on virtual characters
  There are some literature surveys on the application of chat bots and avatars in other domains. For example, [39] studied the application of chat bots in the agriculture domain, or [40] provide a survey on existing chat bots, especially in a customer service context. Additionally, [41] did the same but in the psychotherapy domain. [42] and [43] also reviewed different models of chat bots regardless of the domain. Likewise, [44] provided a survey on the different motion and behavior techniques from avatars to unrestricted autonomous actors.

- Literature survey on evaluation of the performance of the components in our RA
  The performance of each component in the proposed RA needs to be evaluated after designing and implementing. Researchers have followed different approaches for evaluating the performance of their system that have similar functionalities with our components'. Here we review those approaches.

  For evaluating the question generation component, [45] reviewed evaluation methodologies for AQG in two categories: i) intrinsic evaluation methodology and ii) extrinsic evaluation methodology. Also, [46], [47], [37] and [48] provided several approaches and word overlap metrics to evaluate AQG systems and NLG (natural language generation), which are applicable for AQG systems too.

  For evaluating the virtual character component, several researchers had tried to measure the naturalness of chat bots by investigating chat bot performance with respect to different factors, including usability, naturalness and friendliness [49]. For instance, in [50], [51] and [52], authors proposed methods to evaluate chat bots. In addition, in section 2.5 of their work, [53] reviewed some user testing and evaluation tools for the virtual agents. Likewise, [54] and [55] introduced evaluation metrics for chat bot systems. Furthermore, [56] provided a comprehensive evaluation strategy with multiple

metrics designed for evaluating the virtual Agents, [57] also provides two other metrics to evaluate turn-taking dialogue scenarios than precision, recall, F1 score and accuracy metrics; *"average latency (time to take the turn after the user has finished) and false-cut in rate (the rate at which the user is interrupted during their turn)."*

Apart from focusing on other domains, each of these survey papers is concerned with only one aspect of an overall job interview system. However, these papers do suggest areas of future work, since one could apply ideas from those other domains to the job interview automation domain. To our knowledge, our work is the first comprehensive study that has surveyed the publications that specifically cover the required phases of an AI-based job interview system in a systematic fashion to provide the grounds for the proposed reference architecture.

2.3 Conducting Empirical studies in the job interview context

Researchers in the realm of software engineering have conducted empirical researches using quantitative and qualitative data to improve a software product and/or its development process [58]. For example, [59] conduct an empirical study to understand how software engineers think about energy consumption throughout the software production life cycle by analyzing the qualitative data collected from surveying practitioners. Also, [60] followed a similar approach to gain knowledge of the personality characteristics and emotional intelligence and work related preferences among software engineers.

Despite their significance, job interviews are rarely studied empirically in the scientific literature. For example, [61] performed an empirical study and found differences between recruiters' expectations from candidates in the interview evaluation phase; there is a mismatch between what the interviewers actually evaluate and what they really expect from a candidate. Likewise, another preliminary study by " [62] was conducted to investigate the cognitive load differences between public interview settings (i.e., on the whiteboard) and private solving problem interview settings (i.e., on paper) by analyzing the data collected by means of head-mounted eye-trackers. Their findings suggested that "the public setting pressures candidates into keeping shorter attention lengths and experiencing higher levels of cognitive load compared to solving the same problems on private setting". In another study, [63] through an empirical study learned the negative perceptions of software practitioners on the technical interviews in many terms. their findings led them to an *"inclusive hiring guidelines"* for technical (i.e., problem-solving) interviews.

2.4 Applying Topic Modeling to classify public views

One of the widely-used methodologies to analyze the collected textual data in empirical studies is using unsupervised NLP techniques, in particular topic modeling (also called topic detection or topic extraction). Topic modeling is a frequently used text-mining probabilistic method to discover hidden semantic structures or the abstract "topics" in a collection of documents.

This statistical technique is useful in organizing, summarizing and exploration of an extensive textual body by discovering hidden recurring semantic patterns that present across the collection of the textual documents, The extracted main topics (i.e., groups of documents with similar patterns) and detected structures offer us insights to understand existing themes in the large blocks of unstructured textual data.

In this method, a topic is a vector of word probabilities, and a document is a vector of topic probabilities. Thus, a topic that has the highest probability value is the most dominant topic for that document.

Many researchers have applied this technique; for instance, [64] built topic models on posts published about mobile development on the Stackoverfolw platform to learn the popular and difficult topics from the posters' view. Likewise, [65] proposed windowing the topic analysis. To do that, they applied topic modeling on commit messages of a software project to find topics that are about development tasks and, then tracked the system's evolution in these topics.

CHAPTER 3 REFERENCE ARCHITECTURE OF AI-BASED JOB INTERVIEW AUTOMATION SYSTEMS' BASED ON A SYSTEMATIC LITERATURE REVIEW

In this chapter, the proposed reference architecture of AI-based job interview automation systems focusing on the functional view point is first documented at a high-level in terms of both static and dynamic aspects. We then described briefly the high-level non-functional requirements of the system. Then, for each derived high-level component, its respective sub-components are discussed, as well as open issues, challenges in designing and developing them, and identify potential mismatches with existing research on job interview automation. Using the Methodology discussed in the next section, first a systematic literature review study is conducted to survey the publications in the last decade in the context of automation of the job interview process.

This chapter comprises the content of a paper that has been submitted in the Journal of Software and System in December 2021.

3.1 Methodology of doing SLR and proposing Ref

In this section we explain the approach that we adopted to answer the first two research questions.

3.1.1 Approach

An overview of the approach is illustrated in Figure 3.1. To address RQ.1, To address RQ1, we first conduct a systematic literature review to explore the existing works in order to extract the high-level components as well as the brief non-functional requirements relevant to the RA of an AI-based job interview system. This approach, does not rely on any prior knowledge to identify the high-level RA components that are required to be AI-based in a job interview pipeline. Then, we propose a high-level reference architecture and provide the corresponding functional viewpoint (Chapter 3). In order to complete this RA, we had to complement the high-level components identified from the SLR with two additional component to make the architecture complete and consistent.

To answer RQ2, for each component identified from the literature we propose a detailed reference architecture in which we document the respective sub-components in corresponding viewpoints.

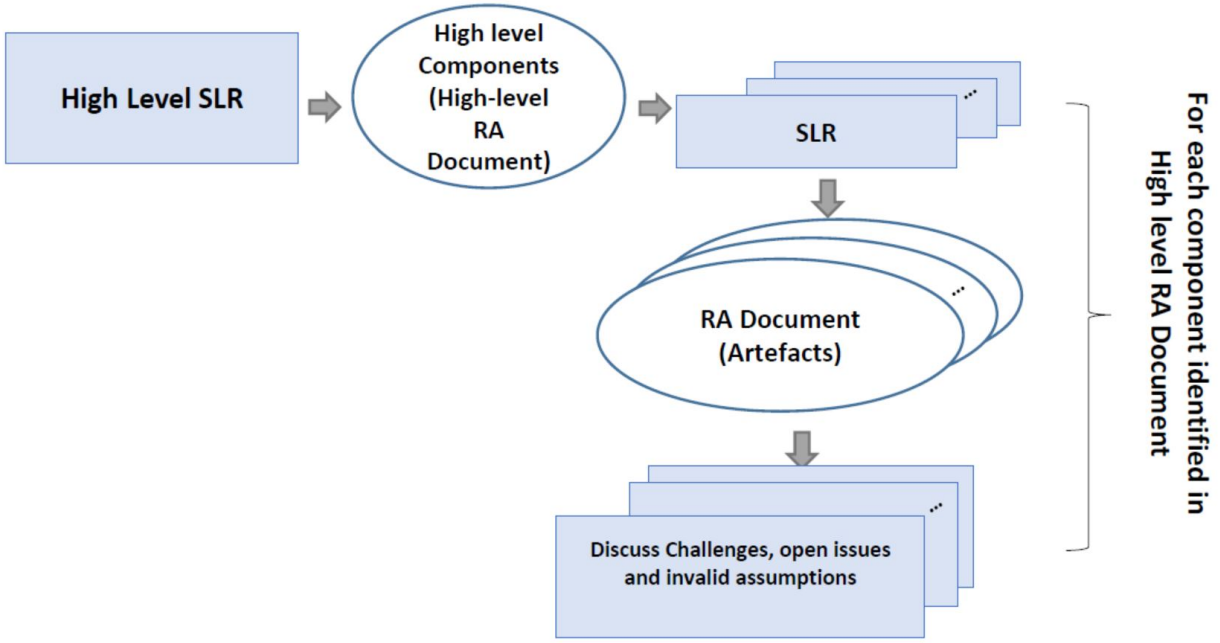

Figure 3.1 Overview of Methodology for proposing the reference architecture of AI-based job interview automation system

To do that, we conduct a separate systematic literature review, one per sub-component, in order to support our reference architecture by providing an overview of existing literature. This provides us a deeper insight into the main functionalities and responsibilities of each component. Also, during these literature reviews we tried to identify the methods, techniques and technologies that have been used thus far in the job interview context.

Note that, the reference architecture focuses on functional view point. Whether the architects of concrete e-recruitment systems decide that some components internally should be implemented using AI techniques, or that such AI technology requires specific AI pipelines from the operational point of view, this information should not be included in an RA. Instead, our discussion of the SLR results surveys AI technologies that have been applied before and that might be options for concrete architecture further down the line.

Table 3.2 shows the reviewed publications in the respective sub-sections. Finally, we discuss the challenges in designing and developing the derived sub-components as well as the assumptions that researcher have made but are not valid or applicable in real situations in existing works. These unresolved issues would open windows for future research studies.

3.1.2 Methodology of the Systematic Literature Reviews

Followed the guidelines proposed by [66] to do our systematic literature reviews at the high-level (overall system) and low-level (individual components). In this section, we explain our SLR methodology.

Data source

To do the SLR, we first collect relevant papers in two steps, i.e., database search and snowballing. In the first step, we decided to search for relevant papers on the This platform offers access to 12 engineering literature and patent databases to cover a wide range of trusted engineering sources including Ei Compendex which is the broadest and most complete engineering literature database available in the world."[1]. In addition, in order to create efficient search queries and synthesize the result set, it provides several exclusion and inclusion search parameters. It is also very flexible in choosing the desired publishing period, language, venues, and authors.

To find the job interview tasks required to be automated (i.e., RQ1), we collect the relative papers by formulating a *"high-level"* query, which is presented in Table 3.1. Given the recency of AI-based job interviews, we considered a publishing date between 2011 and 2021. We also set to search papers where the keywords of the query are in their subjects, titles, or abstracts; subjects refers to the tags or keywords on the papers.

Next, we perform a separate systematic literature review for each component identified when answering RQ1. In this step, we aimed at understanding the AI-based methods, techniques and technologies that have been used in the literature to automated the sub-components we propose in answer of the RQ2. To do these SLRs, for each of the identified high-level components we formulated a more specific query (see Table 3.1). The keywords for these queries are extracted from the papers that are collected though the high-level query. We will explain each query in its relative sub-sections in more detail. Once we collect the papers of either RQ1/RQ2 through database search, we aim to survey only papers that focus on job interview automation context. Therefore, we start to review their abstract and introductions to filter out the papers that do not deal with AI-based job interview pipelines. This explains the difference in numbers of collected papers before and after the filtering them.

[1] https://www.elsevier.com/solutions/engineering-village/content/compendex

Table 3.1 Queries for SLRs of RQ1 and RQ2.

Row	Purpose	Query	#Papers before filtering	#Papers after filtering
high-level	Extract the basic phases required to automate a job interview sessions (RQ1)	(”job interview” OR ”Interview coach*” OR ”HR Interview” OR ”Technical Interview” OR ”Screen interview”) AND (system OR automat* OR intellig* OR smart OR simulat*)	106	20
1	Papers related to the Predefined questions in the job interview context (RQ2)	(”job interview” OR ”Interview coach*” OR ”HR Interview” OR ”Technical Interview” OR ”Screen interview”) AND (question OR (“predefined question” OR “pre-defined question” OR greeting OR introduction OR general* OR generic OR database))	52	13
2	Papers for job interview Automatic Question Generation (RQ2)	(interview* AND (job OR recruit* OR HR)) AND (question) AND (screening OR base OR ”follow up” OR follow-up) AND (generat* OR system OR automat* OR intellig* OR smart OR simulat*)	79	14
3	Relevant papers for virtual interviewer development (RQ2)	(”job interview” OR ”Interview coach*” OR ”HR Interview” OR ”Technical Interview” OR ”Screen interview”) AND (”conversational agent*” OR ”conversational AI” OR Avatar OR chatbot OR virtual* OR ”Virtual Character*” OR ”visual interface” OR ”dialog Agent” OR ”virtual”)	46	23
4	Relevant papers for evaluation of the interviewee’s performance (RQ2)	(”job interview” OR ”Interview coach*” OR ”HR Interview” OR ”Technical Interview” OR ”Screen interview”) AND (metric OR scor* OR measur* OR predic* OR evaluat* OR assess OR analy* OR feedback OR feed-back Oreport OR performance OR capability OR employability OR hireability OR quality OR natural* OR effectiv* OR useful*)	106	7

Snowballing

However, as [67] pointed out, it would be difficult to formulate a perfect search query string that covers all relevant studies without resulting into a large number of irrelevant papers requiring manual filtering. Thus, we perform snowballing as a second step in order to collect more pertinent papers for answering both research questions. In the snowballing effort, we collect additional papers from the references of papers that have been previously found in the first search attempt [66]. In this way, we mitigate the risk of missing relevant papers.

Finally, Similar to the query-based search, we then started to review the papers to exclude those that are not in the job interview context and do not deal with the methods/technologies or functionalities that we aim to discuss in their respective sections, and keep the rest to be reviewed in our SLRs.

Analysing collected papers

In order to analyze the collected papers listed in 3.2 , we applied an inductive qualitative coding technique called open coding and three reading techniques (skimming, scanning and

detailed reading). We first skimmed and scanned the text and if applicable the tables, diagrams and figures of papers in order to get a general overview of the document and spot the sections that include our search keywords or phrases or any of their derivative forms. If we found a relevant keyword or phrase in specific sections of the paper, we read those sections thoroughly and in more depth to critically consider aspects of the text (i.e., detailed reading technique) to provide a summary of that part. We often found those in methodology and evaluation or experiment sections.

In order to group the papers, we categorized them inductively; i.e., one creates some inductive codes based on the qualitative data itself. Following this approach, we read a document and try to give a tag or label (i.e., codes) to each section, such as each sentence or paragraph, etc., and repeat this cycle until we have tagged all documents [68]. This inductive codes or labels (e.g. technology, technique, evaluation methods, tools, etc.) emerged during the analysis processes.

3.1.3 Architectural modeling

Software architecture represents the high-level structure of a software system abstraction. A software architecture must be able to assign the major functionalities to specific components, and satisfy the system's overall non-functional requirements. A software architecture description (AD) must describe its software elements, the properties of those elements, and the interaction among them. [24]. Since architects need to consider and document a variety of properties for their systems, ADs need to incorporate different "views" [69]. the main view is the functional view, in which one document the static structure and dynamic behaviour. We provide Reference Architecture to document this view for the AD, following the guidelines of [70]. According to [70], the functional view describes "the system's functional elements, their responsibilities, interfaces, and primary interactions" (Figure 3.2). Note, although during the literature reviews the applied techniques to automate the identified functionalities are surveyed, it is not the goal of a reference architecture to document information about required techniques, methods or hardware. Instead, we discuss the latter information as complementary to the reference architecture.

Other views (i.e., Concurrency, Development, Deployment and Operational), "concern the software development process, the environment into which the system will be deployed and how the system will be operated, administered, and supported when it is running in its production environment" [70]. Since this study focuses on abstraction of the system elements and their interaction and not on development, deployment or operation processes, similar information for other viewpoints should be defined in future research.

We derived all the required information in this step from the inductive codes that we had created during the analysis of the collected paper.

We use traditional box-and-arrow diagrams of the components to visualise decomposition of components and sub components. In this view, we also document the dynamic behaviour to clearly identify the parts of the system that can execute concurrently or in sequence and how this is coordinated and controlled [70]. UML sequence diagrams are used in this work to illustrate how and in what order different components and sub-components work together from interviewees and recruiters' point of views. The components and their connections in our reference architecture were extracted from those discussed in the surveyed papers, using our created inductive codes.

We also discuss the very high-level non-functional requirements (NFRs w.r.t. to the reference architecture.) for the entire system. Non-functional requirements, as an architectural requirement, are those quality attributes that serve as constraints or restrictions on the architecture of the system [71]. Architectural requirements typically cover system characteristics such as performance, availability and scalability. We mainly extract these requirements from standard guidelines for interaction design or those developed for designing job interviews.

3.2 High-level reference architecture

In this section we will derive and document the high-level reference architecture for job interview automation using the SLR methodology outlined in the previous section.

3.2.1 High-level SLR query

In order to extract the high-level components of AI-based job interview systems (RQ1), we performed our database search to collect the relevant papers with the following high-level query:

("job interview" OR "Interview coach" OR "HR Interview" OR "Technical Interview" OR "Screen interview") AND (system OR automat* OR intellig* OR smart OR simulat*)*

In this query we search for all papers that include "job interview automation" key words or any of its synonyms or derivatives.

This query produced 106 records. By manually reviewing the abstract and title of this query's result set, then applying snowballing, we review the papers to exclude those outside the context of any phase of job interview pipeline and keep the rest to be reviewed in our SLR, yielding 20 papers in total.

3.2.2 High-Level SLR results

Reviewing these collected papers revealed three high-level activities in an AI-based job interview system:

1. Interview Question Generation
   (a) Designing predefined questions
   (b) Automatically generating questions
2. Virtual interviewer interaction through a chatbot (textual), an avatar (audio-visual, virtual) or robot (audio-visual, in-person).
3. Evaluation of the interviewee and their given answers to support the recruiter's final decision.

Below, we discuss each activity at a high-level. Detailed discussions of the activities, their internal architecture and research literature, follows in RQ2.

Interview Question Generation

There are two general approaches for question generation in interview literature. The~first approach involves choosing the right question from a list of questions pre-defined by human recruiters. For example, [72], [73], [74] and [75] experimented with questions that are prepared before the interview session. Such questions include greeting questions, questions that investigate behavioral, ethical and ethical characteristics, capabilities and backgrounds of the candidate. These questions are usually general and applicable for all job domains, but could also be designed for each domain separately. Problem solving questions that are designed before the interview are good example of these domain-specific questions. For instance, [76] used coding problems solving questions which are specific to software engineering domain in order to understand stressful job interview settings.

The second approach involves researchers focusing on automatically generating questions offline based on available resources, such as resumes, the job posts and the information of the previous successful applications. For example, [77] generate questions offline before the interview session using neural network algorithms. Alternatively, the answers of the previous question(s) can be used to generate question during the interview (i.e., online question generation). For instance, [78], [72] and [79] followed this approach to generate follow-up questions using deep learning methods.

Virtual Interviewer Interaction

Developing a virtual interviewer interacting with job candidates is the second phase that we identified in the literature. This virtual character can be developed in textual form (i.e., chatbot) like the chatbot developed by [80], [81], [82] and [55]. Alternatively, this character can be developed in the an (audio-)visual form like [83] and [75] that used an avatar, or [84] and [72], who used a robot as virtual interviewer.

Interviewee Evaluation

Finally, we learned that another critical phase of automation of job interview context is the evaluation and ranking of interviewees based on the results of the interview session. This evaluation usually is performed by comparing the given answer with a list of accepted answers as done by [1] and [82]. However, non-verbal clues such as body gestures and facial expressions can also be integrated into evaluation results ([85]).

3.2.3 Reference Architecture

According to our findings in the SLR, as the answer of RQ1, this section proposes a high-level reference architecture for AI-based job interview systems. The two main use cases that the architecture implements are:

UC1: "recruiter performs interview with candidate"

UC2: "recruiter ranks the interviewed candidates"

The candidate is the interviewee whose resume has been previously matched with the job description and the recruiter is the one who is in charge of the recruitment process. In this section, we document overall static structure and dynamic behaviour of the functional view in corresponding diagrams as well as the high-level non-functional requirements for our reference architecture.

Functional view

This view includes five different functional elements (hereafter components). Three of these were identified in the SLR, while the other two either are required for architectural reasons (Interview Controller) or to fully realize both use cases of the system. For each component, we explain its main responsibilities. Figure 3.2 illustrates the overall static structure and figures 3.3 and 3.4 illustrate the high-level dynamic behaviours of the our proposed RA to

document the functional viewpoint. The two sequence diagrams show how and in what order the 5 different 5 components work together from the interviewee and recruiter's perspective, respectively. Later in this work, in answer to RQ2, we will delve deeper into each component and provide more detail about the interaction between the sub-components within each component.

1. **Interview Controller:**

   This is not a formal component identified from literature, but an essential component to connect the other components (front controller pattern in [86]). We include this component to address the architectural requirement to control the other components, i.e., to execute the right component's functionality at the right moment. Furthermore, this component is responsible for two important tasks for which no research literature could be found: (1) selecting the category of the two types of questions to be asked (i.e., predefined or automatically generated one) and their respective type and (2) about when to terminate the interview session.

   To select the type of questions, existing work such as [72] first asks the candidates all the predefined questions, before asking all the generated ones. However, in reality according to the situation, the interviewer might switch to either of these categories at any time during the discussion. Thus, one of the main challenges in developing this component is to intelligently define and monitor the situation or interview policies to decide on the type of the question to be asked, while the interview is running.

   To decide about termination of the interview, there are some works that pre-define a specific number of questions to be asked during the interview session (e.g. [72], [80] and [87]). However, we could not find any smarter criteria or methodology in the literature for deciding on termination or continuing the session. Therefore, defining smart termination criteria such as terminating the discussion based on the evaluation of the candidate is another major open issue for research on this component.

2. **Questioning:**

   This component is responsible to pick the proper question from the pool of predefined questions or to generate new questions to ask from the candidate given data such as a candidate's answer to the previous question, job post, candidate's resume, historical data of successful applicants, etc. This component also provides the predefined correct answer(s) for the corresponding predefined question from the dataset or provides other means for automatic validation of the quality of an answer to the closed question by the

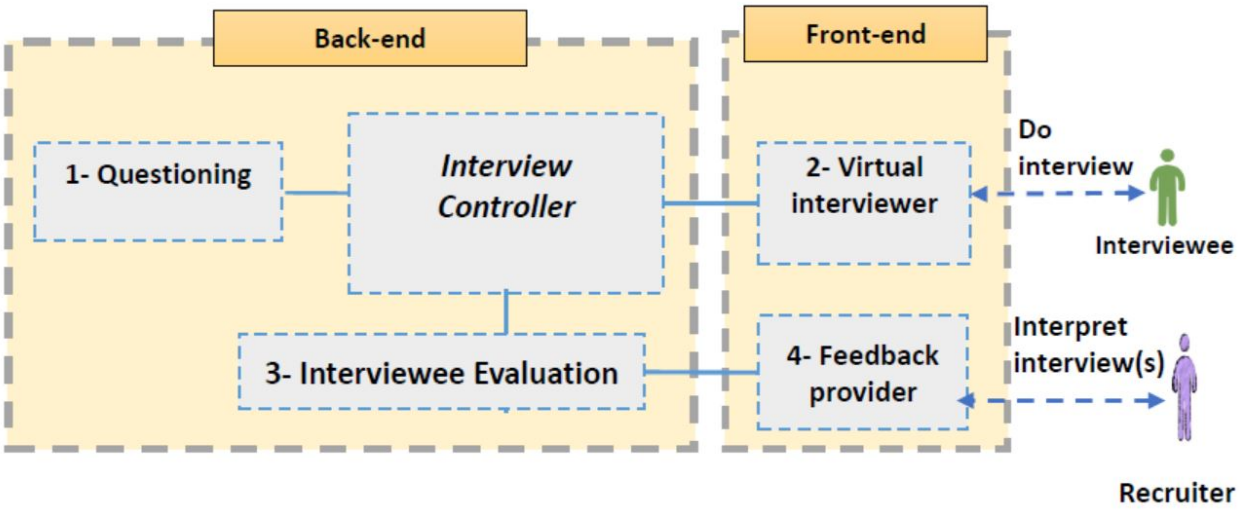

Figure 3.2 High-level functional view point of the proposed reference architecture for Automatic Job Interview (RQ1)

Interviewee Evaluation component. Artificial intelligence capabilities can be leveraged to automate generating questions and answers.

3. **Virtual Interviewer:**

   This component involves creating a virtual interactive interviewer in the form of a chatbot, an avatar, or a robot. This virtual character is the interface between the back-end system and the interviewee. In addition, this component captures the non-verbal cues and social skills of the candidate. AI-based methods can be used to provide a natural dialogue between the candidate and the virtual recruiter.

4. **Interviewee Evaluation:**

   In order to evaluate the candidate, this component processes the given answers. This evaluation can be done through methods for validating an answer against correct answers (as provided by the Questioning component) and analyzing the non-verbal cues and social behavior. The result of this non-verbal cues analysis is integrated with validation of the provided answer to enable more elaborate evaluation. Then, this integrated result is combined with that of all previous questions to calculate a cumulative evaluation result. In this way, the system can provide an instant and dynamic evaluation of the interview discussion, allowing the Interview Controller component to make a decision on termination or continuing the interview session. In order to evaluate the candidate it is possible to apply AI-techniques and follow an AI-based pipeline.

This component also manages the operational data available for later uses by accredited stakeholders. This data includes the computed cumulative evaluation result as well as the recorded interview discussion (i.e., all asked questions, given answers and any other types of non-verbal feed-backs given by the candidates).

5. **Feedback provider:**

   As [88] and [89] argue, exclusively relying on AI solutions to make decisions can lead to bias. Also, [90] discussed how e-recruiting should not rely exclusively on IT-based measures, but on a combination of both online and offline methods to attract, choose and recruit suitable candidates. However, we could not find any publication that focuses on designing or developing such a component in the context of AI-based hiring decision-making processes.

   Therefore, we believe that by including this component into our reference architecture we allow the system to involve human intelligence in the loop to make the most accurate decision about choosing the best candidate for a given job. By providing feedback reports and operational data to the recruiter and other stakeholders, they can make the final decision using human intelligence as well as the evaluation result of the AI-based interview system.

   This component can be designed as a recommendation system which recommends the list of best candidates to the recruiters based on the calculated interview session evaluation as well as other defined criteria. From this point of view, it can be said that this component can be designed using AI-based techniques.

High-level non-functional requirements

Similar to any other dialogue system, an AI-based job interview system should satisfy a number of important non-functional requirements (NFRs). Many NFRs for this system can be extracted from standard guidelines for interaction design such as [91] or those developed for designing employment interviews like [92]. However, the main NFRs generally deal with scalability constraints of the system in terms of number of concurrent job interview that can be run, accessibility and availability constraints, security requirements in terms of authentication, authorization of the users (i.e., HR staff, managers and recruiters, candidates and other stakeholders) as well as security constraints of the stored and transmitted data (i.e., encryption and integrity of contents of an in-transit messages). Likewise, ethical and legal implications should be considered in defining NFRs as [93] described. In addition, usability requirements of the system are another set of non-functional attributes of the system that

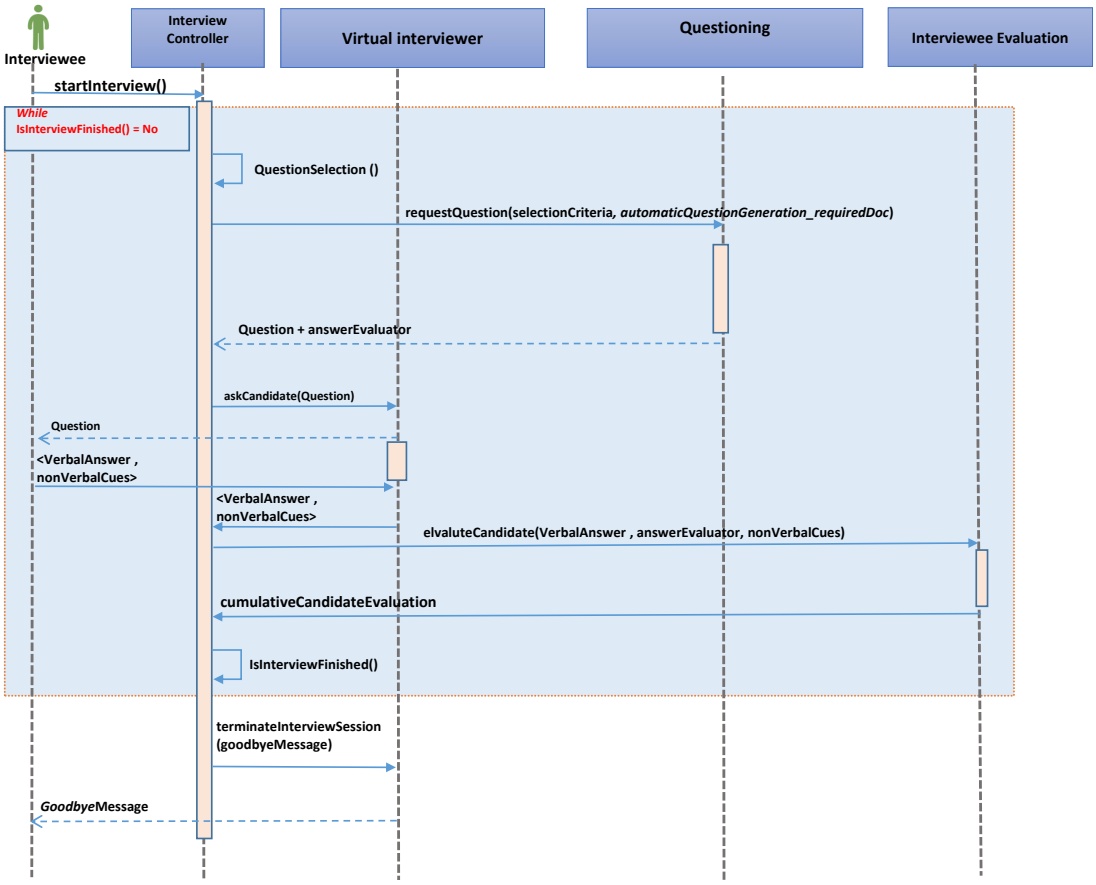

Figure 3.3 High-level sequence diagram of the proposed reference architecture from the interviewee's point of view (UC1;RQ1).

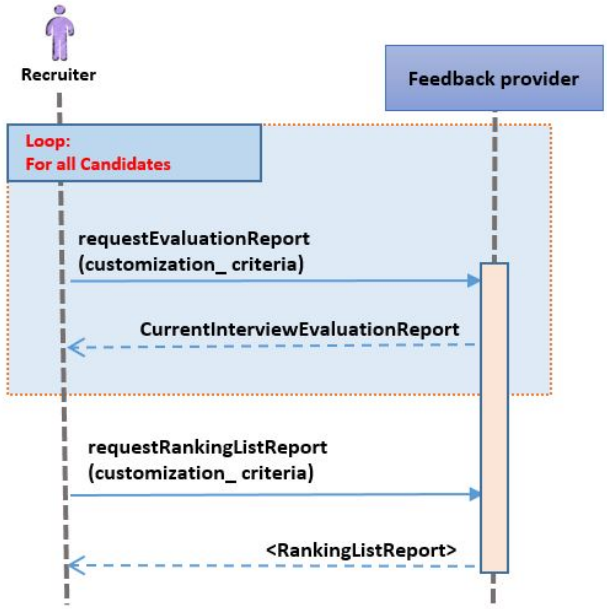

Figure 3.4 High-level sequence diagram of the proposed reference architecture from the recruiter's point of view (UC2;RQ1).

should be defined in future researche. Other major issues such as bias and fairness should be considered as additional NFRs of such systems, but are missing in the existing research on this domain.

In the next sections, to answer our second Research Question, we will propose the detailed reference architecture that we derive for each of the 4 main components (excluding the controller); we will document their sub-components and their functionalities and how they collaborate together to achieve those functionalities in functional view. Then, we will discuss the relevant papers found in the SLR, and link them to the sub-components identified in our reference architecture. Finally, we will discuss relevant challenges, the assumptions that researcher have made but are not valid or applicable in real situations and a decomposition into sub-components.

3.3 Questioning Component

Existing literature has divided interview questions in two categories: the predefined questions that are designed before the interview session, which just require being asked in the right order, and questions generated during the interview based on the interview context and available resources to react to the interviewee's answers. This component also provides a set

Table 3.2 Reviewed publications for the 4 high-level components identified from literature.

Reference	**Predefined Question Lookup**	**Automatic Question Generator**	**Virtual Interviewer**	**Interviewee evaluation**
Shen et al. (2018)		Yes		
Qin et al. (2019).		Yes		
Shi et al. (2020)		Yes		
Su et al. (2018)		Yes		
Du et al. (2017)		Yes		
Chali & Baghaee (2018)		Yes		
Mandasari (2019)		Yes		
Su et al. (2019)		Yes		
Inoue et al. (2020)	Yes	Yes	Yes	
Purohit et al. (2019)		Yes	Yes	
Shimizu et al. (2019)	Yes	Yes	Yes	
Zhou et al. (2019)	Yes		Yes	
Laiq & Dieste (2020)			Yes	
Cartis & Suciu (2019)			Yes	Yes
Sarosa et al. (2018)			Yes	Yes
Junus et al. (2014)			Yes	
Stanica et al. (2018)			Yes	
Cao (2020)			Yes	Yes
Hamdi et al. (2011)			Yes	
Cofino et al. (2017)			Yes	
Gebhard et al. (2014)			Yes	
Andrews et al. (2014)			Yes	
Stanica et al. (2018)	Yes		Yes	
Salvi et al. (2017)	Yes		Yes	
Hoque et al. (2013)	Yes		Yes	
Baur et al. (2013)	Yes		Yes	
Kumazaki et al. (2017)			Yes	
Kumazaki et al. (2019)			Yes	
Schneeberger et al. (2019)	Yes			
Langer et al. (2016),	Yes			
Behroozi & Parnin (2018)	Yes			
Guchait et al. (2014)	Yes			
Kasundi & Ganegoda (2019)				Yes
Maddumage et al. (2019)				Yes
Romadon et al. (2020)				Yes

of acceptable answers for closed questions.

3.3.1 Reference Architecture

We propose 4 sub-components for the Questioning component in our reference architecture, which is illustrated in figure 3.5). These include a Dispatcher sub-component, Predefined Questions Lookup sub-component, Automatic Question Generator sub-component and corresponding Automatic Answer Generator. We included the Dispatcher sub-component for coordination purposes and due to architectural requirements.

The architecture allows the multiple "Automatic Question Generator" sub-components and their corresponding "Automatic Answer Generator" sub-components for generating different categories of (generated question, corresponding answer evaluator for closed question) pairs. Examples of these questions categories are off-line and on-line ones. question generators, which are based on the techniques reviewed in section 3.3.3. Also, the Predefined questions/answer evaluator database stores all the predefined questions and corresponding answer evaluators for closed questions as will be reviewed in 3.3.2.

For the dynamic behaviour of the component, we provide the sequence diagram in figure 3.6. Once the Dispatcher receives a request from the Interview Controller component, depending on the questionType parameter, it either forwards the request to the Predefined Questions Lookup sub-component or to the Automatic Question Generator sub-component. The predefined Questions Lookup sub-component is responsible for selecting the appropriate Predefined Question and corresponding Answer Evaluator available in the Predefined questions/answer evaluator database, given the "selectionCriteria" parameter, which indicates the type or category of the predefined question (e.g. greeting, problem solving , etc.)

Alternatively, the automatic Question Generator sub-component might receive the request from the dispatcher along with required documents for automatic Question Generation (i.e., previous given answers for follow-up question generation of other textual corpus like resume, job posts and etc.). It then generates a question given the received "selectionCriteria" parameter which indicates the type or category of the question (e.g., offline or online), using AI-based techniques. This sub-component also sends the generated closed question to the Automatic Answer Generator sub-component to generate the corresponding answer evaluator for this type of questions, using AI-based techniques. In both situations, the looked-up predefined question or the generated ones along with their answer evaluators (if the question is of the closed type) are sent to the dispatcher sub-component to be forwarded to the Interview Controller component for further processes.

In the rest of this section, we will report the result of our SLRs, based on which we proposed the above reference architecture. In these SLRs, we review the existing works in the job interview context that aim to automate the functionalities of the sub-components of the Questioning component in our reference architecture. This provides more insights into the functionalities, e.g., the typical kinds of questions, the Questioning component should support.

3.3.2 Predefined Questions Lookup

In this section, we investigate the existing works in which predefined questions for job interviews have been developed, to link them to functionalities we considered in our reference architecture for this sub-component (Figures 3.5 and 3.6).

SLR approach

One of the first steps in conducting a job interview is designing the questions for the candidates. These questions can be predefined in advance to be the same for all applicants. Therefore, they must be generic enough to be asked from all participants for a specific position regardless of the inherent differences in their separate job interview sessions or their characteristics and backgrounds. Answers of these questions provide general information of the candidates like their strengths and weaknesses, experiences and qualifications. Alternatively, these questions might be domain-specific or even job-specific. The crucial issue to be considered while asking these questions is how to choose the right order of questions or even how to choose the right questions from the list (since not all questions should be asked all the time.)

In this section, we survey the publications that put their focus on designing predefined questions based on the following query (Table 3.1):

("job interview" OR "Interview coach" OR "HR Interview" OR "Technical Interview" OR "Screen interview") AND (question OR ("predefined question" OR "predefined question" OR greeting OR introduction OR general* OR generic OR database))*

This query produced 52 records. After manually reviewing the abstract and title of this query's result set, then applying snowballing method, we review the collected papers to exclude those that are not discuss managing the predefined questions for job interview and keep the rest to be reviewed in our SLR, yielding to 13 papers in total.

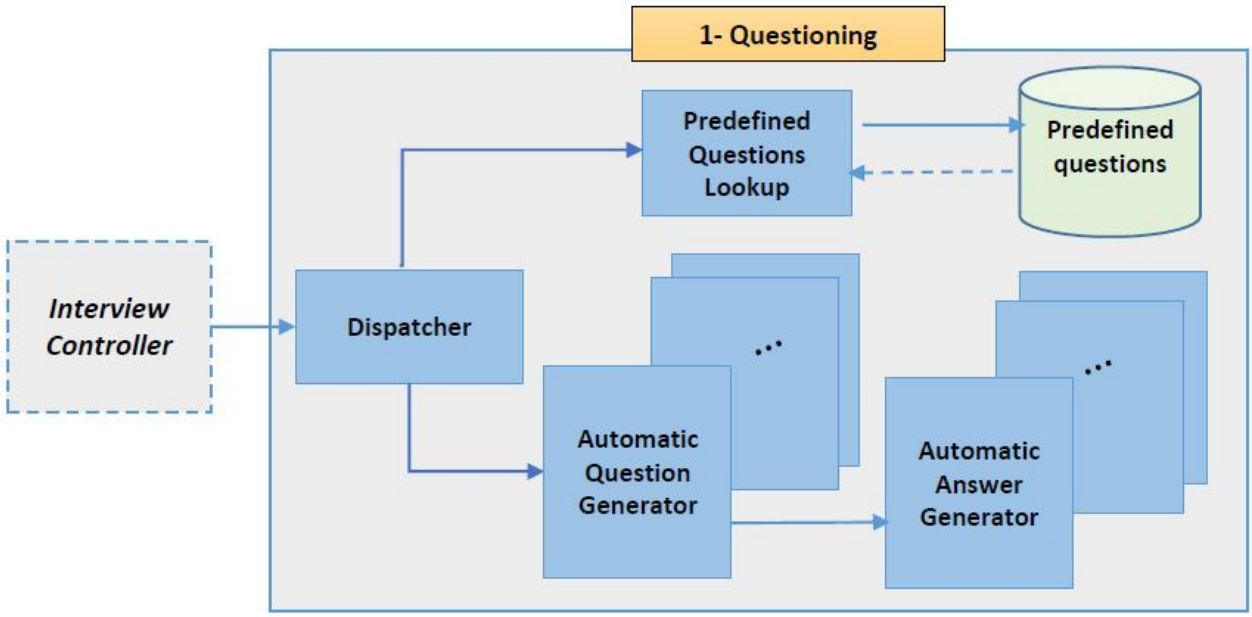

Figure 3.5 Functional view point of the questioning component in the proposed reference architecture (RQ2)

SLR results

Researchers designed many different questions and categories that are designed before the interview sessions.

Greeting or introduction questions: The first step in any job interview session is initializing the discussion between the job applicant and interviewer(s). This initialization usually starts with a *greeting or introduction*. This step might involve only the interviewee by just doing a self-introduction. For example, in [80], the implemented system asks the user to first present a self-introduction prior to asking more detailed and precise questions. A simulated job interview can also start with a welcoming and introduction discussion. For example, [94] and [95] asked the candidate to first provide a self-introduction.

Questions to explore the capabilities and competencies of the applicant: A second type of general questions that can be asked after the introduction questions, aims to *explore the capabilities and competencies of the applicant* for the position. For example, in an experiment conducted by [73], the interviewees answer questions about their resume and biographical, situative information as well as social questions such as the applicants' pro-activity, their strengths and weaknesses or questions about their organizing ability and critical faculties. Also, in the model proposed by [72], the system initially asks four base questions about four

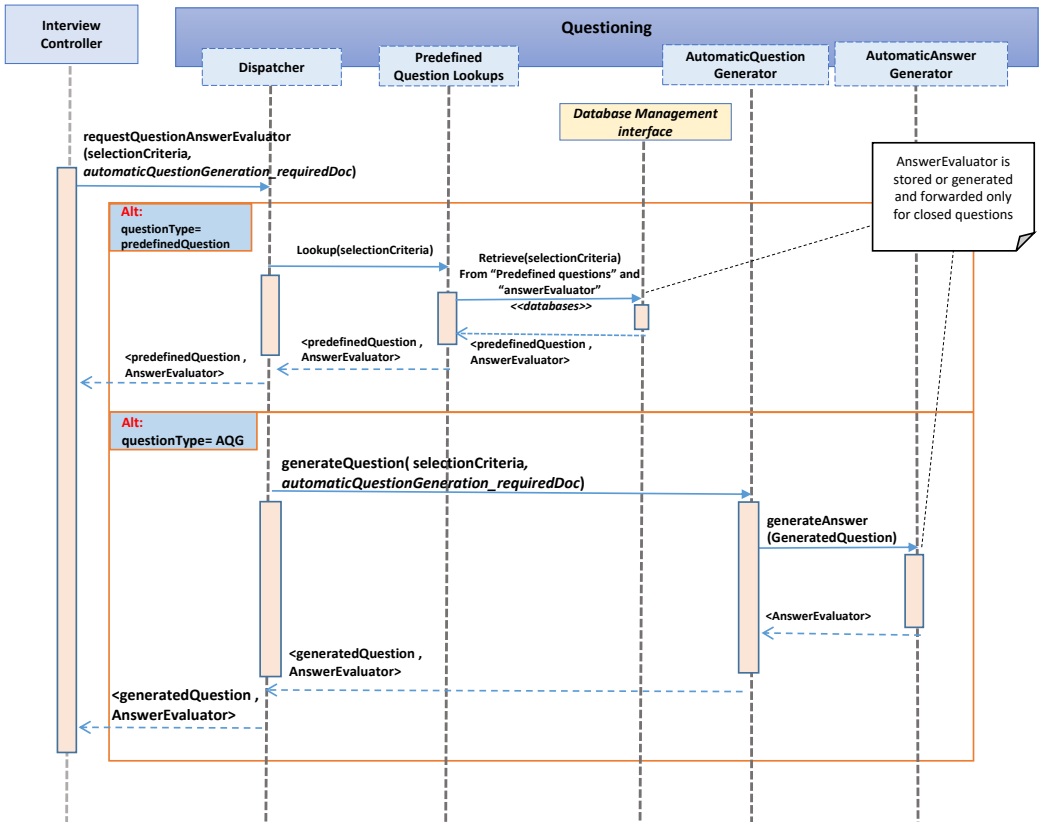

Figure 3.6 Sequence diagram of the questioning component in the proposed reference architecture (RQ2)

different topics ("What is the reason for applying?", "What are your strengths?", "What are your achievements?" and "What are your skills?").

Also, in another work by [74], interviewees participated in a mock interview to answer general interview questions that *explore the interviewees' resume, strengths, weaknesses* for the job, questions that explore the interviewees' biographical information and questions that assess their pro-activity and critical thinking. They also asked two situational questions to examine interviewees' organizing ability and persuasiveness. Likewise, in [87], authors developed 15 common generic interview questions such as "So, please tell me about yourself.", "Tell me about a time when you demonstrated leadership.", "Tell me about a time when you were working in a team and faced with a challenge. How did you solve that problem? ", "What is your weakness, and how do you plan to overcome it?" and "Why do you think we should hire you?" Moreover, the artificial interviewer developed by [80] asks a set of casual questions like "If you had a superpower, what would it be?"

Domain-specific questions: Questions can be designed in advance for each domain separately. For instance, in an effort done by [96], the system asks two sets of predefined questions; questions related to hard skills in the domains of project management and programming competencies as well as questions related to soft skills to assess to the interviewee's social competencies, communication skills and personality-related traits that required for this kind of positions. In another work, [75] after processing and extracting the relevant information such as skills, scores of various examinations, achievements, and degrees/certifications from the interviewee's resume, the system selects an appropriate difficulty level of some predefined questions based on the category of the interviewee and their applied specific position. Another type of predefined domain-specific questions are problems (usually domain-specific ones). For instance, in the experiment done by [76], they use the result of the interview sessions in which applicants were asked to solve coding problems solving ones.

There are some *commercial SaaS (Software-as-a-service) solutions* that simulate job interviews using predefined questions. For example, Hirevue [2] provides a video interviewing technology that helps recruiters to screen the candidates. It records the answers of job applicants to pre-set questions. Some examples are "What makes this position a good fit for you at this point in your career?", "Tell us how your experience and training have prepared you for this position."," What circumstances led you to apply for this position?"," Give us your understanding of our organization." and "What would you miss most/least about your current job?". [16] used this system to conduct interviews to study the impact of video interviewing on the recruitment process's effectiveness. Also, to understand applicants' perceptions

[2]https://www.hirevue.com/

of digital interviewing and its impact on the recruitment process's effectiveness, [16] used commercial SaaS (Software-as-a-service) solutions. They asked eleven open-ended industry-specific questions (i.e, hospitality management sector) . Tengai [3] is another interview solution that is developed to assess soft skills and personality traits through interviews and using a robot. Moreover, Recright [4] have produced a platform as a video recruitment tool. Likewise, Wonderlic test [5] helps companies to screen candidates through job-related questionnaires.

3.3.3 Automatic Question Generator

This section explores the conducted efforts on generating questions automatically and how they evaluated their generated questions in the job interview context.

SLR approach

Since the predefined questions are often generic ones across all domains/positions, real job interviews also require generating questions in real-time and according to the situation and available resources, and this should be integrated into a full-fledged job interview automation system. [97] defined Question Generation (QG) as *"automatically generating questions from various inputs such as raw text, database, or semantic representation".* Research on Automatic Question Generation (AQG) goes back to the 70's ([98]). In this section, we survey the publications that focus on Automatic Question Generation (AQG) in the job interview context.

To collect the corresponding papers for automatic question generation, we run the following query (Table 3.1):

(interview AND (job OR recruit* OR HR)) AND (question) AND (screening OR base OR "follow up" OR follow-up) AND (generat* OR system OR automat* OR intellig* OR smart OR simulat*)*

This query produced 79 records , which, after similar filtering as in 4.2.1, results in a final set of 14 papers.

SLR results

In the following, we present the results of offline and online AQG, respectively (Table 3.3). By offline question generation we mean generating questions before the interview session,

[3] https://www.tengai-unbiased.com
[4] https://www.recright.com/en/
[5] http://www.wonderlic.com/

given some textual input, while the questions that are generated online are created given the answer of the candidate to the previous question(s) of the interview session in real time.

Offline Question Generation

There are many papers on automatically generating questions offline (before the interview) from a given corpus, applying different approaches.

Rule-based Question Generation techniques. These techniques were one of the first attempts to meet the objective of automatically generating questions. The idea here is to use a knowledge base with documents to create hand-crafted rules. For example, [99] used language patterns and WordNet to generate multiple-choice questions and [100] proposed a rule-based model to generate questions. Although the precision of the questions that are generated by applying rule-based techniques is rather high, low scalability is a drawback in these systems, since the performance of these approaches heavily depends on the existence of well-designed rules, which requires substantial human effort and domain-specific expertise ([101] and [102]).

Neural Network-based Question Generation techniques. Introducing neural network techniques has improved question generation models in general and many researchers leverage this technique in AQG. For example, [103] and [104] applied Seq2Seq with an attention mechanism for question generation tasks and trained their model. Although many of these proposed models thus far are not necessarily designed to the job interview context, and the applied technique can be adopted in this context too. For instance, the applied techniques in automatic question generation for educational purposes ([38]) can be applicable in the job interview context. The textual course material is analogous to the job post or the applicant's resume documents.

However, there are some proposed NN models that focus on automatically generating proper questions given a corpus specific to in the job interview context. A good example of domain-specific questions that are generated automatically by the system is DuerQuiz, proposed by [77]. This system uses the content of the applicant's resume and the job posts and the information of the previous successful applications to extract the required job skills to create a skill-graph. The skill-graph includes the required skills that should be evaluated in the job interview. Using this graph, the system will then fill question sentence templates with the skills (i.e., nodes in the graph) through slot filling methods. Some example template sentences are *"Try to introduce the...", "Please introduce the characteristics of..."* and *"Please introduce several common feature selection algorithms for..."* . To implement this system, the authors applied bidirectional Long Short-Term Memory (LSTM) with a Conditional Random Field (CRF) layer (LSTM-CRF) neural network algorithms. In another effort proposed by [78], the system generates screening questions from the given job post. They proposed a deep

learning model called Job2Questions to detect intent from the text job description and rank the intent from the job post in order to generate the questions.

Online Question Generation

Although all the above models generate questions automatically, the given input is the existing textual input available before the interview session (i.e., the candidate's resume, job posts or information of previous successful applications for that position). In other words, the generated questions are independent of the applicant's response to the previous question(s) in that conversation and produced before the interview session, statically.

However, in a real human-driven job interview, recruiters usually ask some follow-up questions based on the given answer of the candidate to the previous question. Therefore, some researchers proposed models to generate follow-up questions dynamically during the conversation, which is applicable in interview conversation too. For example, [53] followed a *rule-based* approach and created rules for pattern detection in a user's answer on conversations with the topic "getting to know you". Leveraging the named entity, part of speech information, and the predicate-argument structures of the sentences, they proposed a template-based framework to generate follow-up questions. Also, [105] applied the same techniques to generate follow-up questions for an open-domain dialogue agent.

When it comes to the job interview context, there are only a few papers on generating follow-up questions. Existing research typically uses Neural Network-based techniques or NLP techniques to address this issue.

Neural Network-based techniques. One of the works that used NN models to generate follow-up questions is done by [79]. They proposed a follow-up question generation model for interview coaching (including job interviews) in Chinese. First, for sentence pattern generation, they used a clustering method, then applied the CNTN technique to select a target sentence in an interviewee's response. Then, a seq2seq model takes this sentence as input to generate the follow-up questions. Finally, to rank the generated questions, they used an n-gram language model. Likewise, [106] proposed a model in which they applied the CNTN model for selecting a key sentence from the candidate's answer. They populated a domain ontology, then used it to generate the follow-up questions. Also, in the second stage of the AI-based job interview system proposed by [72], the follow-up questions are generated based on keywords in the previous response. These keywords are first extracted from the given answer to the previous question using a neural network model. This NN model consists of a one-layer bidirectional long short-term memory (BLSTM) followed by a three-layer linear transformation with an output layer. Then, the system fills the extracted keywords in a predefined template to generate a follow-up question.

Natural Language Processing (NLP) techniques. On the other hand, some researchers applied Natural Language Processing (NLP) techniques to generate follow-up question. For example, [107] proposed an AQG model that after performing resume analysis, first prepares questions based on the job category for which the candidate applied. Then, using NLP and sentimental analysis, it generates follow-up questions based on the candidate's answers to the previous question. In another effort, [95] proposed an interactive question generator component for their system by applying NLP techniques. This component extracts nouns from the answers using a Japanese morphological analyzer. Then, it combines the nouns and five template sentences to generate questions and stochastically rank one of them as the best one. These template sentences are based on a Japanese corpus containing examples of job-interview questions.

Table 3.3 Publications on automatic Question Generation in the job interview Context

Reference	Static/ Dynamic	Dataset Language	Method	Experiment dataset	Performance	Limitations
Qin et al. (2019).	Static	English	LSTM-CRF	• Historical recruitment data • Interview questions data • Search query log data	• Skill Entity Extraction module: *Job posting*: Precision: 0.87, Recall: 0.80, F1: 0.83 *Resume*: Precision: 0.71, Recall: 0.71, F1: 0.71 • Skill Entity Filter module: Precision: 0.97, Recall: 0.99, F1: 0.94 • Skill Relation Extraction module: Precision: 0.82, Recall: 0.62, F1: 0.71	• Focused on required skills to formulate questions • Low performance of "Skill Entity Extraction" and "Skill Relation Extraction" modules. • Experiment of one job domain-specific dataset(low generalizability)
Shi et al. (2020)	Static	English	Deep learning	Job posts on LinkedIn	0.91	Generate question based on Job post; ignore Resume content in AQG
Su et al. (2018)	Dynamic	Chinese	CNTN	3390 follow-up question/answer pairs	• Accuracy: 0.88 • BLEU: 0.31	• Low performance of "pattern-based method with language model" (BLUE: 0.316) • Calculates the word similarity for word clustering just looking up the E-HowNet (not a comprehensive knowledgebase) • Low performance of "pattern-based method with language model" (BLUE: 0.316) • Calculates the word similarity for word clustering just looking up the E-HowNet (not a comprehensive knowledgebase)
Su et al. (2019)	Dynamic	Chinese	CNTN	• ConceptNet domain ontology • 1,754 ordinary and 1,262 follow-up questions	• Key sentence selection module: 0.81 • Ontology triple population module: 0.76 • Sentence similarity matching module: 0.92	• Experimenting and trained only with Chinese dataset • Low performance of "Ontology triple population" module (Accuracy: 76.59%)
Inoue et al. (2020)	Dynamic	Japanese	BLSTM	Human-Human dialogue	Precision: 0.63 Recall: 0.45 F1: 0.52	• Experimenting and trained only with Japanese dataset • Low performance of : Follow-up questions based on keyword extraction" module (F1-score: 52.7% , Precision: 63.1%, Recall : 45.2%)
Purohit et al. (2019)	Dynamic	English	NLP	N/A	N/A	No Experimenting
Shimizu et al. (2019)	Dynamic	Japanese	NLP	Mock job interviews	N/A	• Small training dataset (143 and 5 question template sentences • No AQG Experiment

3.3.4 Automatic Answer Generator

In our reference architecture, the Questioning component also provides a set of acceptable answers for closed questions; the correct answers for predefined questions is stored in a database and the answer for closed questions is automatically generated in the "Automatic Answer Generator" sub-component. Although there has been extensive research on Automatic Answer Generators context in other domains (e.g. [108] in e-commerce and [109] in education area), we could not find any literature focusing on the job interview automation area.

3.3.5 Challenges and Invalid assumption

In this section we discuss invalid assumptions that have been made in existing interview-related question generation work. We also provide research opportunities and the main challenges of the sub-components for this components.

Predefined Questions

Since it is not feasible to find a comprehensive and standard data set of questions and answers across all possible job domains and descriptions, a major challenge for this sub-component is to find a scalable way to populate and maintain such a database.

In addition, the predefined questions in the reviewed papers typically are assigned equally to all interviewees (e.g., [96] and [76], assuming that all candidates have equal background and knowledge) However, this assumption is too simplistic, since in reality different sets of questions should be pre-designed proportional to different levels of background and knowledge ([92]).

Moreover, in some papers some questions are designed in a way that the given answers are hard to interpret for a machine in order to score a candidate, or it is difficult to consider the correct answer(s) as a baseline to compare it with the candidate's response. In general, most of the papers assume that answers will be interpreted by humans, not an AI-based system (e.g., [72], [80], [74], [96] and [87]). Therefore, an important challenge is how to integrate humans in the AI-based system loop, and how to do that in the architecture of the system.

Furthermore, while most papers concern designing and asking questions in general without considering their difference in importance/role, there must be a methodology for identifying the proper question to be picked from the available pool of questions. Yet, we did not find any paper looking into this problem.

Automatic Question Generation

During this literature study, we identified a number of open questions related to automatic question generation. For example, researchers rely on the job interview data set that they collected themselves to train or evaluate their models. Similar to the previous subsection, such data sets are not available freely online, hence require substantial effort to gather, clean and maintain, providing again an important hurdle towards building and operating an AI-based job recommendation system.

This data set essentially requires to be as similar as possible to real job interview settings, leading to enhancing the reliability of the model. It also needs to be large-scale enough to provide an acceptable level of generalisability and usability of the system. Similar as in the previous subsection, such data sets are not available freely online, hence require substantial effort to gather, clean and maintain, providing again an important hurdle towards building and operating an AI-based job recommendation system.

Likewise, many researchers used the slot-filling method using question sentence templates that they have defined themselves without a standard, available corpus. These templates also should be defined.

In addition, in all the papers that we reviewed on online question generation in the job interview domain (e.g., [79], [106], [72], [107] and [95]) authors assumed that the next question can be predicted exclusively based on the answer of the previous question, allowing to generate follow-up questions. Yet, questions can (often should) be generated considering other documents (e.g., resume, job posts and etc.) that are fed into the system instantly and dynamically, along with the answer of the previous question. In real situations, the recruiter considers the whole discussion and possibly all earlier answers of the candidate to ask the next question.

Automatic Answer Generation

Although we only find a few works in literature that focus on automatic answer generation in the job interview context, the corresponding approaches in other domains might be applicable here too. However, a challenge in designing this functionality is again providing the required data for training the models. Therefore, while in the e-commerce domain, for instance, the products' description can be fed to the system to generate an answer for the customer's question, in the job interview context one needs access to given answers of the previous successful applicants, assuming that the job has been offered before and enough good answers received then.

For example, in a very limited domain, ontologies could be used to extract correct answers for a given question. [1] used such an ontology to retrieve all possible answers about Java knowledge. However, it is not easy to generate ontologies for every possible job domain, unless the system focuses on narrow set of domains. Another challenge is generating not only one but a set of acceptable answers for a given closed question with different weights. In addition, automatically generating responses for the follow-up questions is a challenging issue.

3.4 Virtual interface component

In any AI-based interview system, including a job interview system, human effort should be reduced as much as possible in all tasks. Hence, apart from delegating the task of coming up with questions to models, an automated job interview system should also to some extent, replace the human recruiter's presence during an interview by a virtual character, in order to minimize human involvement in interviewing the participants. In this way, recruiters benefit from reducing cost/time/energy. It also mitigates the risk of potential human bias due to the subjective nature of the in-person interview sessions for interviewees.

3.4.1 Reference Architecture

Some researchers like [85] argue that, along with verbal answer of the candidate, their non-verbal cues during the interview session should be captured to enable a better evaluation. Non-verbal cues refer to body language, facial expressions or even the pattern of the voice of the candidate, while a verbal answer is the transcribed answer. Therefore, in our proposed reference architecture we considered both the non-Verbal cue capturing and verbal answer capturing functionalities.

The static diagram for this component in figure 3.7 includes 4 sub-components: Dispatcher, NonVerbalCueCapturer, VerbalAnswerCapturer and Virtual interviewer Character. Similar to before, we include the Dispatcher sub-components for coordinating purposes and due to architectural requirements. The Virtual interviewer Character sub-component is an interface between the system and the candidate and can be developed in textual or graphical form using the available frameworks as reviewed in section 3.4.3. The NonVerbalCueCapturer and VerbalAnswerCapturer sub-components, on the other hand, extract the non-verbal cue and verbal answer from the raw answer of the candidate, respectively.

The sequence diagram (figure 3.8) shows the dynamic behaviour of this component. Once the Interview Controller component sends the request along with the question parameter to

the Dispatcher, the latter forwards it to the Virtual interviewer Character sub-component to ask the question from the candidate and receive their raw answer, which will then be forwarded to the Dispatcher. The Dispatcher sub-component, in turn, sends the raw answer to the NonVerbalCueCapturer and VerbalAnswerCapturer sub-components to extract the non verbal cue and verbal answer. Alternatively, if the Interview Controller component decides on terminating the interview session, it sends a terminateInterviewSession command to the Dispatcher sub-component to ask the Virtual interviewer Character finish the discussion with the interviewee.

3.4.2 SLR approach

For this component's SLR, we used the following query:

("job interview" OR "Interview coach" OR "HR Interview" OR "Technical Interview" OR "Screen interview") AND ("conversational agent*" OR "conversational AI" OR Avatar OR chat bot OR virtual* OR "Virtual Character*" OR "visual interface" OR "dialog Agent" OR "virtual")*

This query produced 46 records, which, after filtering resulted in 23 papers.

3.4.3 SLR results

There are several publications focusing on automation of the virtual interaction with candidate. There are several techniques using AI-based pipe lines in creating a virtual but a natural dialogue, such as natural language understanding (NLU) and natural language generation (NLG) and natural language processing (NLP), automatic speech and emotion recognition. Many of the surveyed publication in this section have leveraged some of these techniques.

Textual characters (Chatbots)

To engage in a natural conversation, chatbots that act as a virtual interviewer are commonplace for performing the job interview. [110] explored the influence of chatbots on the recruitment process and found that chatbots are helpful in improving performance of the recruiters. Many researchers leveraged chatbots to develop their proposed job interview models. For example, [80] proposed models to come up with a set of practical design suggestions for building effective chatbot interviewers. They evaluated their model using a web-based system that includes a chatbot interviewer driven by a human operator. In general, chatbots are designed to *capture only the verbal answer of the interviewees* while interacting with them, as explained next.

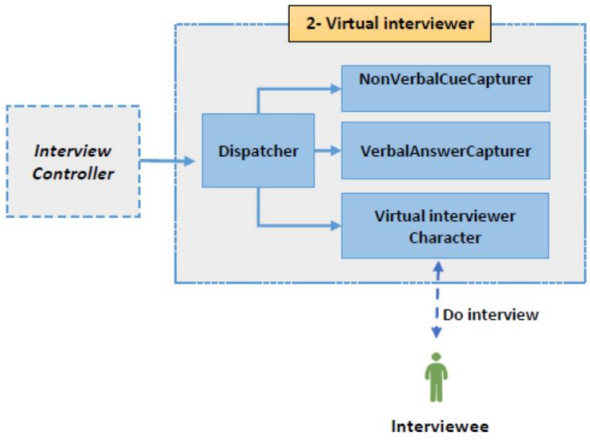

Figure 3.7 Functional view point of the Virtual interviewer component in the proposed reference architecture

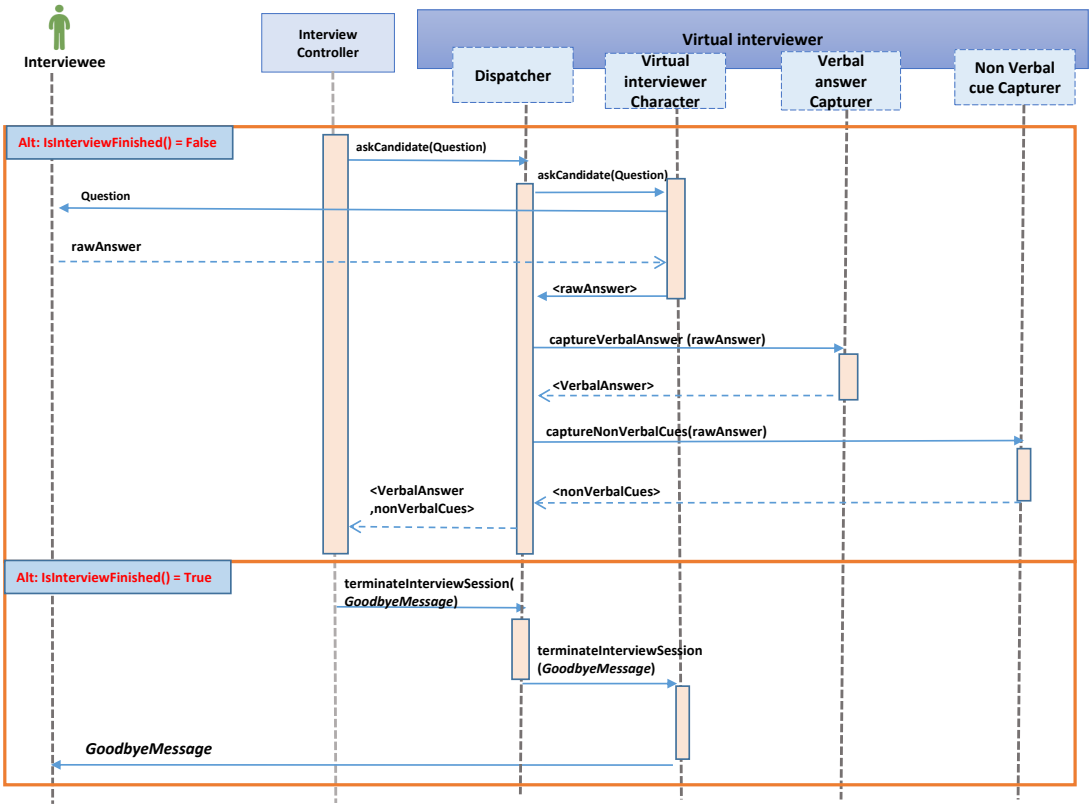

Figure 3.8 Sequence diagram of the Virtual Interviewer component in the proposed reference architecture

Commercial Technologies.

Some of the researchers used commercial technologies to develop their chatbots. For instance, in order to prepare requirements engineers for job interviews, [81] developed a chatbot interviewer. using IBM Watson technologies [6]. Likewise, the model proposed by [82], used IBM Watson technologies to conduct screening interviews using a chatbot through which the candidates fill a simple web form.

Open-source technologies.

However, many of the developed chatbots for job interview sessions used open-source technologies. For example, Sarosa et al. used an open-source interviewer bot named ALICE to help students practice speaking English appropriately for job interviews ([111]). ALICE uses pattern-matching techniques and is developed in AIML (Artificial Intelligence Markup Language), an open XML language. Also, Junus et al. used ALICE for the same purpose to develop their interviewer bot ([112]).

Likewise, [96] built a system to prepare software engineers for real job interviews. To develop their chatbot, they used the functionalities provided in Pandorabots, while for training the bot they again used AIML (Artificial Intelligence Markup Language) files. Pandorabots [7] is a free open-source-based website allowing developing and publishing chatbots on the web. Moreover, [55] proposed a model to prepare job applicants in the field of data science for job interviews. In order to implement the chat capabilities, they embed another open-source chatbot framework named 'Rasa', which includes two components: i) Rasa NLU which is a library for natural language understanding (NLU) and ii) Rasa Core. Likewise, JARO leveraged Dialogflow[8] application provided by Google. Google's Dialogflow provides the chatbot interface that can be integrated into a website. Authors integrated their chatbot into the website using webhooks ([107]).

Graphical characters

Compared with simpler text-only communication offered by chatbots, graphical job interviewers add a more personal dimension to interview sessions using open-source or commercial technologies, as discussed below. Potentially, graphical form of virtual characters are capable to detect the non-verbal cues of their users.

Avatars:

[6] https://www.ibm.com/watson/how-to-build-a-chatbot?
[7] https://home.pandorabots.com/home.html
[8] https://cloud.google.com/dialogflow

Some researchers leveraged avatars to visualize the interviewer's character in the simulated interview session. Most of the papers we reviewed below *capture both the verbal answers as well as non-verbal cues of the interviewees.*

For example, to improve the candidates' behavioral skills, [113] simulated a job interview session where an Embodied Conversational Agent (ECA) plays the role of a virtual recruiter and asks predefined contextual questions to measure the technical skills of the candidates. According to [114], ECA is a *Virtual characters serve as communication partners for the user realizing non-verbal and verbal interactions.* Their ECA supports body gestures and facial expressions. In order to assess the emotional state of the candidates, the ECA adapts the predefined scenario depending on the emotional state and behavior of the candidate. In this study different signals and input modalities are considered to collect the related data including Physiological Signals (i.e., facial expression, heartbeat-related information, discharge of neurons in the brain and the electrical resistance of the skin), speech (i.e., pitch, tone, speed), textual content and static and dynamic gestures. These data are collected using a Brain computer interface, Biofeedback sensor and Microphone and webcam.

Open source technologies.

Some of the researchers used open source technologies to develop avatars. For instance, [83] use a dialog system where a virtual human-like avatar plays the role of the interviewer. In order to develop their dialog framework, they utilized the multimodal HALEF dialog system, which is an Open-Source modular web-based multimodal dialog framework[9]. They modeled, textured, rigged, and animated the avatar in Blender, such that it features lip, head, arm, chest and leg movements to provide a natural appearance. They also used Papagayo studio [10] to synchronize the lip movements with the dialog. Although, the user's video and audio interactions are continuously streamed and recorded during the session, the system does not capture any non-verbal cues of the interviewee.

Likewise, [115] built two avatars with different characters (understanding and demanding) to explore the impact of the social behavior profiles of the interviewer on the job interviewees. To build the avatars, they used SEMAINE API, which is an open-source framework for creating emotion-oriented systems with virtual characters. Their avatar featured joint movements like the neck joint or the spine joint and 14 facial expressions; she is also able to perform lip-sync speech output. This framework records a user's social cues by a Microsoft Kinect device and a headset to be analyzed and reacting appropriately. They used the SSI framework ([116]) and NOnVerbal behavior Analyzer (NOVA) ([117]) to record and analyze various

[9] http://halef.org

[10] http://lostmarble.com/papagayo/

social signals. Different body and Facial features (i.e., Postures, gestures, head gaze, smiles, motion energy, overall activation) as well as Audio Features (i.e., Voice activity, intensity, loudness, pitch,audio energy, duration, pulses, periods, unvoiced frames,voice breaks, jitter, shimmer, harmonicity, speech rate) are detected in this work.

[118] developed a C++ game, through which candidates can practice different interview scenarios. Their game is based on the Cold Nebula game engine, which is an open-source game engine in C++. Through this game, candidates can interact with the system with a Microsoft XBox controller or keyboard input. During the mock interview, a non-player character (NPC) interviewer asks a multiple choice question and some follow up questions based on the selected answer and its related scenario. Despite this interaction, their system does not capture any non- verbal cues of the interviewee.

Unity, the cross-platform game engine[11], is another free-to-use option that some researchers used to develop an avatar as the virtual recruiter. For example, Stanicai et al. proposed a VR-Job application for training job interviewees purpose. Their system includes two simulated interview rooms with their corresponding avatars using Unity 3D ([96]). Also, additional devices are used in order to monitor the emotions of the interviewee; In order to identify stress, they monitor the heart rate by skin monitor sensors, fitness bracelets or even smartwatches.

Finally, [75] proposed a model useful for self-preparation for having a successful job interview. In their model, after a chatbot conducts an aptitude test, the interviewee participates in a face-to-face interview with a human model who asks questions via audio files. This avatar is created in Unity and has various gestures such as hand movement, lip synchronization feature and some other facial expressions. In this work, using Fisher face algorithm, a representation of emotion of the candidate is provided to evaluate their confidence.

Commercial Frameworks.

Other works use commercial frameworks. For instance, in an effort conducted by [94], an avatar plays the role of a recruiter in a simulated a job interview environment, with the purpose of training social skills required in a real situation. To develop their interactive model, they used a software framework that supports behavior control for avatars ([119]). In their model, using combination of sensors and software algorithms like SSI framework (by [116]) and Microsoft Kinect, they also record and analyze seven social cues and psychological signals of users, including Hand to face, Looking away, Postures (i.e., Arms crossed, Arms open, Hands behind head), Leaning forward/backward, Voice activity and Smile.

Likewise, to help people practice social interactions in face-to-face scenarios, including job

[11]https://unity.com/

interviews, [87] developed MACH (My Automated Conversation coacH) in the form of a 3D character that can see, hear and make decisions in a real-time manner. They used the Multimodal Affective Reactive Characters (MARC) platform ([120]) to develop the avatar. As well as speech recognition (i.e. verbal answer capturing), they captured some non-verbal cues; From the video of the user's face, they detect smiles and head movements (e.g., nods, shakes, tilts). In order to detect smiles they used the Shore Framework ([121]) and tracked the "between eyes" region for detecting head nod ([122]). They also preformed prosody analysis by recognizing pauses, loudness and pitch variation. There are also some commercial solutions that integrate an avatar for job interview training, such as Molly [12].

Robots:

Another type of virtual character are physical robots that are used to simulate in-person job interviews, using proprietary technology. For instance, [84] and [123] used an android robot to model a face to face mock job interview for individuals with an autism spectrum disorder. They used Actroid-F as a female humanoid robot with an appearance and voice like a real human. This robot featured some facial expressions like smiling, nodding, and brow movements and is operated with remote control. Likewise, [72] used an android robot interviewer, called ERICA. It is an autonomous conversational robot ([124]) leveraging 46 motors in her face and body, and she can simulate different facial and body expressions, gestures, and movements. Her voice is a text-to-speech system that is trained on a real voice of a Japanese actress. In order to capture verbal answers, they used a 16-channel microphone array for automatic speech recognition. Also, in order to track the subject's position for the robot, they compute an estimation of the sound source direction based on the multi-channel speech signals and using a Kinect v2 sensor. None of these researchers, however, captured automatically any non-verbal cue during the interview sessions in their work.

VR Environment:

Some researchers leveraged virtual reality to visualize the room of the interview session. For example, [125] developed their virtual environment as a typical manager's office using an open-source technology named Xtranormal [13]. The researchers have not captured any non-verbal cue during the interview sessions.

3.4.4 Challenges and Invalid assumptions

In some papers, authors assumed that the behavioural interaction between the candidate and recruiter is one-directional. In other words, the appearance and body language or gesture

[12]https://www.jobinterviewtraining.net/
[13]http://www.xtranormal.com/

of the developed virtual characters does not change based on the given feedback from the candidate (e.g. [82], [55], etc.)

Moreover, authors that developed any form of virtual job interviewer assumed that there is no difference between interviewing with a virtual character and a real human for the candidate; i.e., their given answers and behaviour clues are not biased. Also, defining important cues in evaluating the candidate and methods for capturing them in such a way that have not impact on the behavior of the candidate (e.g., increasing the stress level) is another open issue in developing this component. A multitude of different behaviours, gestures, visuals, etc. have been proposed, yet it is not clear which design parameters matter most, and how they impact the successful hiring of candidates.

3.5 Interviewee Evaluation component

This section is about the component responsible for the evaluation of the candidate's competencies for occupying the job post based on the interview discussion and results.

3.5.1 Reference Architecture

For similar reasons as the previous component, our reference architecture considers "VerbalAnswerAnalysis" and "Non-verbalCuesAnalysis" sub-components for analysing verbal answer and non-verbal-cues of the candidates, respectively. The "VerbalAnswerAnalysis" sub-components decide, based on the "open"/"closed" nature of the questions and other policies, what similarity measure will be used. For closed questions, it uses the generated answer of the Questioning component and for open questions it will use other means for doing similarity analysis.

Integration of the results of the verbal answer analysis and non-verbal cues analysis is another responsibility of this component, followed by computing the cumulative evaluation result. This evaluation is done by retrieving the computed cumulative result for all the previous questions available in the Operational database, through the "Interview Controller component". Both of theses functionalities are done in "ResultIntegration" sub-component. The static diagram for this component is shown in figure 3.9.

The sequence diagram in figure 3.10 shows the dynamic behaviour of how the sub-components interact with each other. First, the Interview Controller component sends a request to the InterviewEvaluationController to evaluate the candidate along with the Question, Verbal answer and answer evaluator and, non-verbal cues as the parameter of this request. The Question and the answer evaluator with the correct answer for the closed questions were

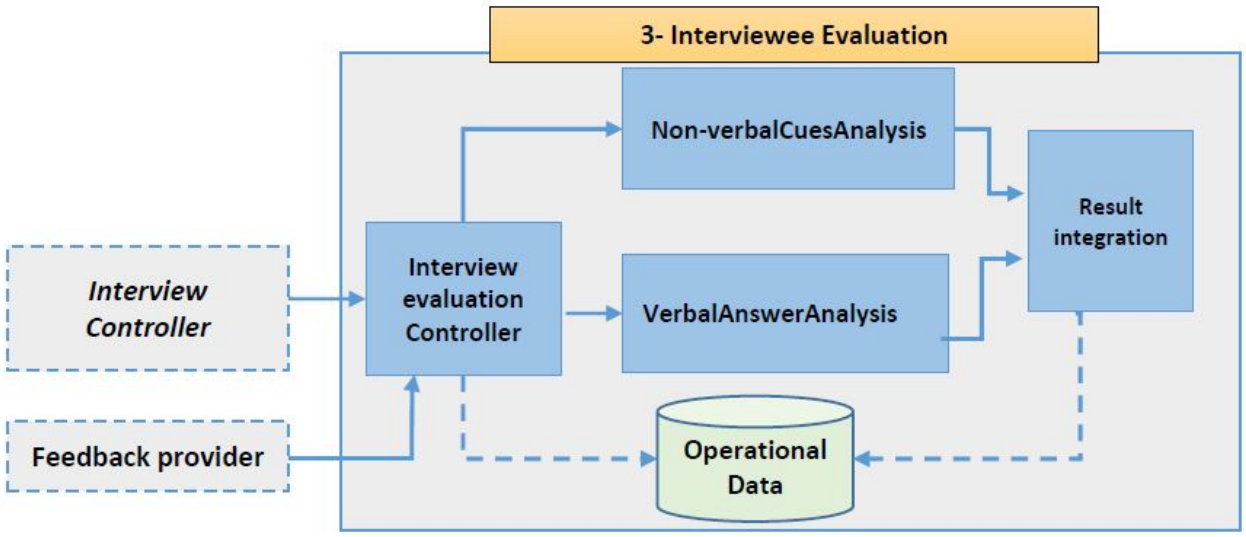

Figure 3.9 Functional view point of the Interviewee Evaluation component in the proposed reference architecture

obtained earlier from the Questing component, and the Verbal answer and non-verbal cues were obtained from the Virtual interviewer component.

Next, the InterviewEvaluationController sub-component stores the Question into the operational data database for further access by the recruiter and sends the verbalAnswer and AnswerEvaluator parameters to the Verbal Answer Analysis sub-component for similarity analysis, while the Non-Verbal Cues parameters is forwarded to the Non-verbal cues Analysis sub-component. After storing the result of the two analyses into the operational database they are forwarded to the ResultIntegration sub-component to calculate the cumulative result by retrieving the computed cumulative result for the previous questions available in the Operational database and storing the up to date cumulative result in the operational data database.

3.5.2 SLR approach

We first explore the literature on developing the functionalities of this component in the job interview context. We used:

search query: ("job interview" OR "Interview coach" OR "HR Interview" OR "Technical Interview" OR "Screen interview") AND (metric OR scor* OR measur* OR predic* OR evaluat* OR assess OR analy* OR feedback OR report OR performance OR capability OR employability OR hireability OR quality OR natural* OR effectiv* OR useful*)*

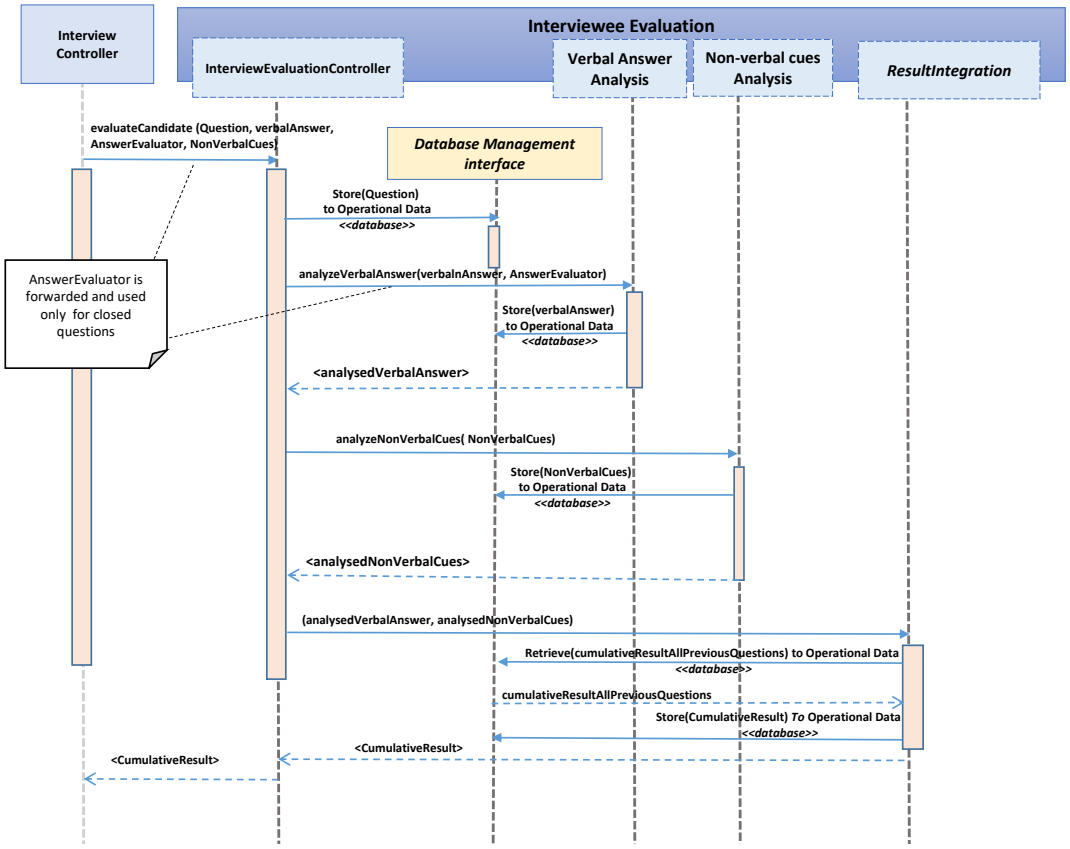

Figure 3.10 Sequence diagram of the Interviewee Evaluation component in the proposed reference architecture

This query produced 106 records. We filtered the papers to focus on AI-based evaluation of a candidate yielding only 7 papers. This drop in the number of papers is due to the second operand after the AND operation, since most of the returned papers are about the evaluation of their proposed models not the about evaluating the actual candidate's performance in the interview.

3.5.3 SLR results

There are few publications on automation of the evaluation of the candidate. There are several AI techniques to AI-based pipe lines to automate this functionality like classification of the candidate to prediction of their hire-ability based on syntactic, semantic and sentiment analysis of the given answers and ML classification algorithms. The surveyed publication in this sections have leveraged some of these techniques.

Analysis of verbal answers:

Most of the paper we reviewed concerned the analysis of the verbal answers of the candidate through applying different approaches, as follows.

Some researchers rely on *similarity analysis between the given answer and the desired ones*. For example, in the Java domain, [1] calculates semantic similarity, using WordNet to calculate the average similarity between the candidate's answer and the multiple correct answers for the same question. Their system retrieves these answers from a stored ontology of the technical (i.e., java) knowledge. Finally, it returns the average of the total marks as the candidate's score for the technical skills. Also, [75] performed a syntactic and semantic analysis to evaluate the given answers of the interviewee. They compared the words of the interviewee's answer syntactically with the correct answer. By creating a derivation tree, using Latent Semantic Analysis and converting the sentence to its logical form, they performed semantic comparison.

However, [82] took another approach; they evaluate the provided answers and score the interviewees with a sentiment analysis algorithm based on IBM Watson Personality Insights service. [55], on the other hand, are concerned with word co-occurrence rather than semantics analysis. Taking the idea of Jaccard similarity, their sentence similarity algorithm takes the word sets of two sentences and calculates the percentage of keywords that the respondent has covered in their answer. In this way they measured how many key concepts of the correct answer the interviewee could explain.

There are also some conducted efforts that used *machine learning techniques* to evaluate a candidate. For instance, [126] applied text classification methods on the transcribed audio

(i.e., verbal answers) of the interviewee to make a prediction of the hire-ability of the interviewee; then this automatically generated prediction is compared with the human experts' prediction, who do the final assessment of the candidate. In another effort by [111], authors applied the Naive Bayes Algorithm to classify the interview results into three categories including interest (interested, less interested, interested, very interested), potential (not skilled, less skilled, skilled, highly skilled), and talents (visual, psychomotor).

Analysis and integration of non-verbal cues:

Only a few researchers take the non-verbal cues into consideration when automatically assessing the interviewee. For example, while [75] evaluate the confidence level of the candidate by performing emotion detection and also separately analysing their verbal answers, they did not integrate the result of these two analyses to compute a final evaluation.

However, [85] argued that there are two perspectives to evaluate the interviewee in terms of the correctness of their given answer as well as the quality of the given answer. The quality of the answer is important since questions might have multiple correct answers or an answer that could be expressed in several other ways. They suggest to measure the confidence level of the respondent as representative of the quality of the answer, because they believed that if a candidate knows the answer he is more confident. They discussed how confidence can be measured using different metrics like eye contact, voice patterns, and hand gestures to evaluate the quality of the answer in terms of its correctness. They applied voice patterns for this objective along with semantic similarity calculation to compute the weighted average of the confidence score and semantic similarity score for overall answer evaluation. In this way they *integrated the result of analysis of given verbal answer with that of non-verbal cues* to make more elaborate evaluation.

3.5.4 Challenges and Invalid assumptions

In this section, we discuss the main challenges and unresolved issues in designing this component's sub-components or responsibilities in order to open research opportunities windows for future.

Operational Database

There must be a database that stores all the operational data required for candidate evaluation by the system or future access by human recruiters. This database should include the important data of the interview sessions of different candidates, such as the recorded

Table 3.4 Publications on virtual interviewer development in the job interview context

Reference	**Chat bot/ Avatar/ Robot/ Virtual Environment**	**Open Source Technology**	**Developmeny Interviewer Technology**	**Verbal anwser/Non-verbal cue capturing**	**Non-verbal cue capturing technologies**
Zhou et al. (2019)	Chat bot	N/A	N/A	Verbal anwser capturing	N/A
Laiq & Dieste (2020)	Chat bot	No	IBM's Watson	Verbal anwser capturing	N/A
Cartis & Suciu (2019)	Chat bot	No	IBM's Watson	Verbal anwser capturing	N/A
Sarosa et al. (2018)	Chat bot	Yes	ALICE framework	Verbal anwser capturing	N/A
Junus et al. (2014)	Chat bot	Yes	ALICE framework	Verbal anwser capturing	N/A
Stanica et al. (2018)	Chatbot/Virtual Environment	Yes	Pandorabots, Unity	Verbal anwser capturing	N/A
Cao (2020)	Chat bot	Yes	Rasa framework	Verbal anwser capturing	N/A
Purohit et al. (2019)	Chat bot	Yes	Google's Dialogflow	Verbal anwser capturing	N/A
Hamdi et al. (2011)	Avatar/ Virtual Environment	N/A	N/A	Capture Physiological Signals (i.e., facial expression, heartbeat-related information, discharge of neurons in the brain and the electrical resistance of the skin), speech (i.e., pitch, tone, speed)	Brain computer interface, Biofeedback sensor and Microphone and webcam.
Cofino et al. (2017)	Avatar	Yes	HALEF dialog system	Do not capture any non-verbal cues	N/A
Gebhard et al. (2014)	Avatar	Yes	SEMAINE API	Different body and Facial features (i.e., Postures, gestures, head gaze, smiles, motion energy, overall activation) as well as Audio Features (i.e., Voice activity, intensity, loudness, pitch,audio energy, duration, pulses, periods, unvoiced frames,voice breaks, jitter, shimmer, harmonicity, speech rate)	SSI framework and NOnVerbal behavior Analyzer(NOVA)
Andrews et al. (2014)	Avatar/ Virtual Environment	Yes	Cold Nebula game engine and MS XBox controller/keyboard input	Do not capture any non-verbal cues	N/A
Stanica et al. (2018)	Avatar/ Virtual Environment	Yes	Unity	Identify stress by monitoring the heart rate	skin monitor sensors, fitness bracelets or even smartwatches.
Salvi et al. (2017)	Chat bot and Avatar	Yes	Unity/ HAAR cascade algorithm	emtion recognition(i.e., Happiness, Fear, Sadness, Neutral, Surprise)	Fisher face algorithm
Hoque et al. (2013)	Avatar	No	MARC (Multimodal Affective and Reactive Character)	prosody analysis(i.e., pauses, loudness and pitch variation) and detect smiles and head movements (e.g., nods, shakes, tilts).	Shore Framework to detect smiles and tracked the "between eyes" region for detecting head nod
Baur et al. (2013)	Avatar	No	EMBR Engine	seven social cues and psychological signals of users, including Hand to face, Looking away, Postures (i.e., Arms crossed, Arms open, Hands behind head), Leaning forward/backward, Voice activity and Smile.	SSI framework and Microsoft Kinect,
Kumazaki et al. (2017)	Robot	No	Actroid-F Android robot	Do not capture any non-verbal cues	N/A
Kumazaki et al. (2019)	Robot	No	Actroid-F Android robot	Do not capture any non-verbal cues	N/A
Inoue et al. (2020)	Robot	No	Android robot(ERICA)	Do not capture any non-verbal cues	N/A
Villani et al. (2017)	Virtual Environment	Yes	Xtranormal technology	Do not capture any non-verbal cues	N/A

interview discussion including all asked questions, given answers and any other types of non-verbal feed-back given by the candidates. Type, format of this data as well as the duration of this storage in a way that satisfies both the security and usability measures of the system should be defined through a research study.

Verbal Answer Analysis

One open issue that no one addressed in the automation of the job interview is the standard criteria and concepts to evaluate the candidate. For instance, the authors that follow a similarity-based approach (e.g. [1], [1], [55] and etc.) assumed that there is only one correct answer for each question to be compared with the given answer, while in reality there might be multiple correct answers for a questions.

Moreover, there is no conducted effort for assigning weights to the responses to compute the overall evaluation value. In the papers that we reviewed, authors assumed that all responses should be weighed equally in the evaluation process (e.g. [1], [1] and [55]). In reality, some responses are more appropriate than others. For example, the more optimal solution for a problem-solving question should receive a higher score.

Furthermore, since the acceptable answers for similarity analysis can be provided only for closed questions, some policies must be defined for evaluating the answers given to open questions. Often, questions cannot be answered 100% in the allocated time. Therefore, an approach should be defined to see how a candidate approaches a complex problem, which is more important than the actual answer. To date, no papers have focused on this major issue.

Non-verbal Answer Analysis

Although some researchers like [85] included voice patterns as non-verbal cue, future research should address other influential non-verbal cues that might impact the candidate's evaluation in a real situation, such as body gesture, tone of voice and other emotional cues and their automated measurement criteria and methods, should be addressed in future research.

Final Evaluation Result Computation

In order to assess candidates' competencies, the result of comparison of their answers against correct answers should be integrated with the result of analysing their non-verbal cues during the interview session. [85] calculate weighted averages to integrate the results. Also, they only considered voice pattern classification as a non-verbal cue to reflect the confidence of the given answer. However, in reality multiple non-verbal cues influence the final evaluation of

the candidates, such as eye contact, body gestures and other emotional cues. Moreover, more complex approaches can be followed to combine the result of the two analyses; for example, it might involve assigning different weights to different types of non-verbal cues for different questions.

Another open issue is defining an incremental method through which the computed results for all questions can be combined to compute a cumulative evaluation result. This instant and dynamic evaluation computation is required to help make an automated decision on termination or continuing the of an interview.

Furthermore, in some proposed models such as [77] and [78], authors did not consider the rank of generated questions in terms of their importance, while, in real situations answering some question is more important than others. Thus, ranking the questions when computing the cumulative evaluation result requires more research. Likewise, in order to make a fair evaluation, further research should be conducted to identify the criteria that lead to bias introduced by data-driven artificial intelligence decision support system ([89] and [88]) and learn how to avoid them during automatically evaluation of the candidate.

3.6 Feed-back provider component

To our knowledge, there is no research on designing and implementing this component, whose goal is to report and summarize the job interview results to the recruiter or other stakeholders, in order to support the task of ranking the interviewed candidates.

This component would create several customized reports by summarizing and visualizing the evaluation computed by the system. Summarizing mostly refers to refining the data to extract useful information, and visualization refers to providing info-graphs. Reporting is one of the most effective ways to gain a thorough understanding of what the 'perfect employee' looks like from the perspective of the AI-based job interview system. These reports give the recruiters the overall process of interviewing the applicants and selecting the most suitable ones, while ranking them. These reports capture all the details about the interview sessions and candidates' records for a given job post at the fingertips of the recruiters. Also, the purpose of reporting is to be able to adjust the company's strategy in order to make recruitment more efficient or more productive.

Moreover, although most recruiters use the reports to adapt their interview process, others require such reports to prove they are adhering to rules and regulations. The public sector, for instance, often considered their Equality and Diversity summary to be one of their most important reports as the regulations are stricter for them than in other sectors (e.g., Canada's

law on Employment Equity Act [14].

On the other hand, we believe it is also essential to enable human-in-the-loop and involve human intelligence in evaluating the interviewees by providing views of the operational data to the recruiters. This allows to reduce the bias introduced by relying only on AI-based solutions as discussed by [88] and [89]. For example, an AI-based decision making system might decide that the given verbal answer of a question is in-appropriate, while it can be an appropriate answer from the recruiter's perspective. Moreover, the AI-based evaluations of answers given for open questions might not be as elaborate as the evaluation done by a human. A similar case might happen for non-verbal cues. For instance, while machines might recognise that a change in voice pattern is due to low confidence of the candidate in answering a question, in reality it might be because of an irrelevant change in the situation of the interviewee's room, which can be only noticed only by human.

As best practice, this component can play the role of a recommendation system which recommends the list of best candidates to the recruiters based on the calculated interview session evaluation as well as other defined criteria. AI technology could be used for that.

3.6.1 Reference Architecture

Based on the above functionalities, obtained through reasoning about our overall RA thus far, our reference architecture considers 4 sub-components for this component. Summarizing and visualizing the evaluation computed by the system is conducted in the "Summarization" sub-component. Then, the "ReportGeneration" sub-component generates customized reports for the recruiters based on their given filters and parameters for each candidate.

This candidate-level summarization is followed by candidate pool-level ranking: the "CandidatesRanker" sub-component helps the recruiters to have an overview of how each candidate is doing relative to other interviewees. Having the operational data of other interview sessions, this sub-component ranks the candidates relative to each other. The detailed static and dynamic aspects of the sub-components in this component is illustrated in figures 3.11) and 3.12, respectively.

3.6.2 Challenges

In this section, we discuss the main challenges that we envision for this component.

[14] https://laws-lois.justice.gc.ca/eng/acts/e-5.401/FullText.html

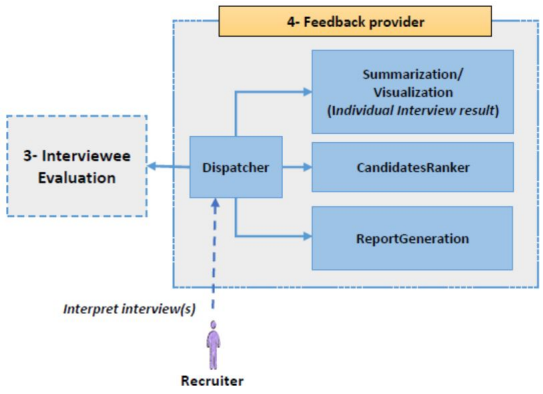

Figure 3.11 Functional view point of the Feedback Provider component in the proposed reference architecture

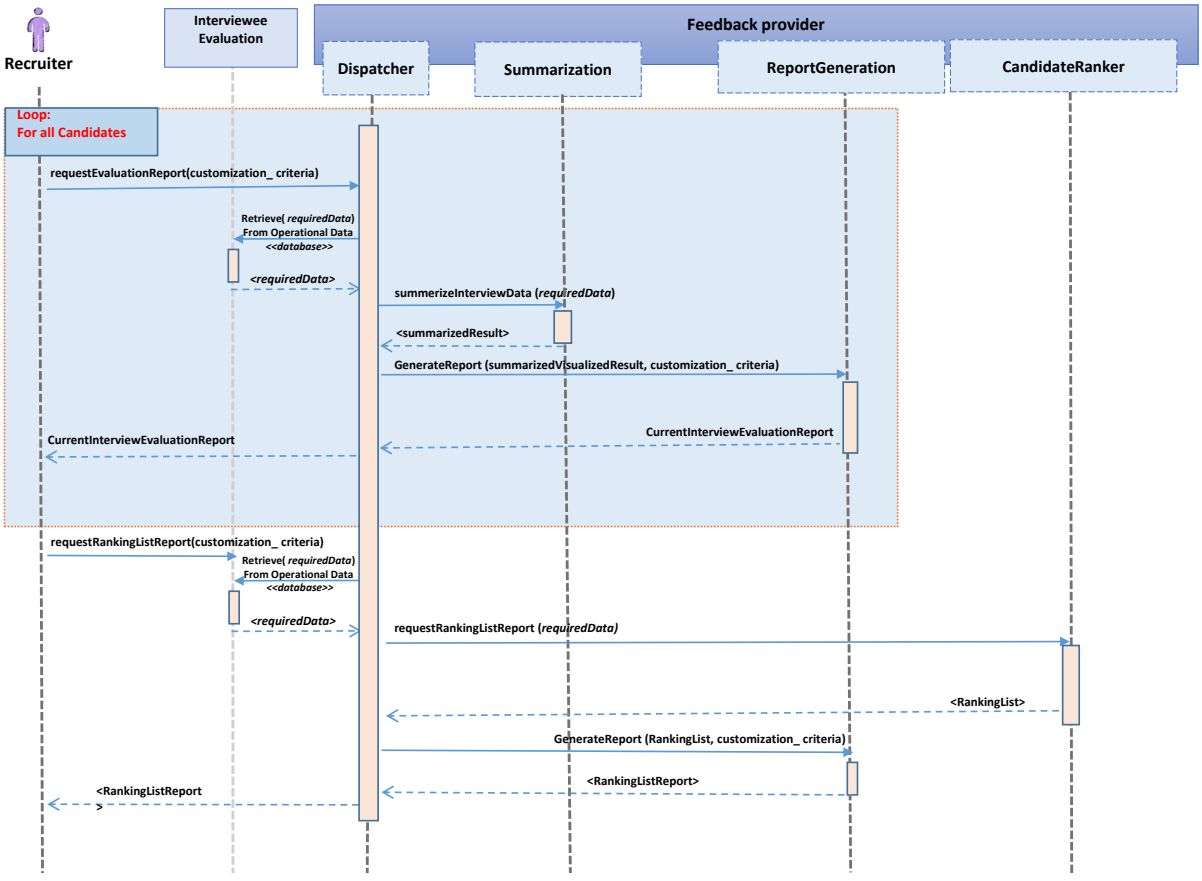

Figure 3.12 Sequence diagram of the Feedback Provider component in the proposed reference architecture

Report Generation

One of the main functionalities of this components is generating reports summarizing the candidates' performance. Defining the useful type (i.e., visual or textual reports) and required amount of provided information in a report from the recruiters' perspective is a challenge that needs to be addressed in future research.

Ranking Candidates

Having the operational data of other interview sessions, this component should give an overview of how each candidate is doing relative to other interviewees; it should provide a ranked list of all applicants for a position. Defining the relative criteria and algorithms for making such comparison is another challenge in designing this component.

3.7 Discussion

After systematically surveying the publications in the last decade in the context of automation of the job interview process, we derived the main practices that have been simulated to achieve an AI-based job interview system, to be included in the proposed reference architecture, including the questioning activity, virtual communication with the interviewee and evaluation of the applicant based on the result in the interview session of this pipeline. We also included a controller component for architectural requirements and a feed-back provider component in order to avoid bias introduced by relying only on AI technologies in human-computer interaction systems. Next, for each component we explored the literature for its sub-components in our proposed reference architecture; the methods, models, techniques and technologies that researchers proposed or applied to automate them. From the SLRs we, learned that AI-based methods can be applied in designing many of the functionalities of some of these components.

We also discussed the open issues to design and develop different components and their respective sub-components that are ignored in the literature and need to be addressed in developing an AI-based job interview system to highlight directions for future research. Some of these challenges refer to the need for collecting standard and large-scale datasets for populating and training the system. Other challenges indicate the need for research studies to propose appropriate methods to design/develop each component/sub-component from the recruiters' point of view. Our literature review indeed shows that there is a gap between the achievements and assumptions of the human resource science researchers and those of the researchers in the artificial intelligence field. That is, many of the surveyed proposed models

are not applicable in real situations due to invalid/inadequate assumptions of their authors and require further studies.

For instance, in reality several criteria, like the number of questions, duration of a session, evaluation of the candidate's give answers to previous questions, etc. involve making a decision on termination or continuing an interview session, while this aspect is ignored in the literature that we surveyed. Also, there must be methods to look up the appropriate questions in the predefined questions pool based on the situation of the discussion. Regarding the automatic question generation, a method that takes the whole discussion into consideration in order to generate the next follow-up question is lacking in existing works. Moreover, proposing a methodology for evaluation of the given answers to open questions and to the generated follow-up questions needs further studies. Likewise, ranking the questions in terms of their importance as well as ranking the given answers in terms of their correctness in the candidate's cumulative evaluation phase are ignored in the literature. Finally, identifying the non-verbal cues that influence the final evaluation of the candidates and proposing a method to analyse and integrate the analysis results into cumulative evaluation computation is another challenging issue.

Different parties can benefit from our work. For one, it provides several research opportunities for researchers. They can conduct research efforts to resolve the described challenges for each component. Also, our work provides opportunities for researches to bridge the existing gaps between human resource science and artificial intelligence science by surveying the stakeholders in the human resource domain to understand their requirements. Though such studies, they would learn the requirements of all stakeholders in this pipeline as well as other constraints and limitations that needs to be addressed in an AI-based system. A stakeholder is anyone who has an interest in the outcome of the AI-based job interview system including the hiring managers, business owners, human resources and even the external recruiters as well as the candidates themselves. Finally, companies that produce AI-based e-recruitment systems can leverage our proposed architecture to accelerate and improve the design of their systems. They can use the proposed RA as a base-line which provides an initial ideas of the essential components and functionalities needs to be included in an comprehensive job interview. Also they can leverage our findings in the SLRs to have a comprehensive overview of the available techniques to automate those functionalities.

3.8 Summary

This chapter proposed a comprehensive reference architecture (RA) for an AI-based job interview automation systems through architectural modeling technique. To do that, we

first conducted a systematic literature review (SLR) to provide a ground truth ground for extracting the high-level components and a list of very high level non-functional requirements of an AI-based job interview automation systems. Then, for each component, we conducted different SLRs to derive the corresponding main sub-components and their functionalities. We also discuss the challenges in designing and developing each extracted component, based on the knowledge we gained through the SLR phase (i.e., invalid assumptions and open issues).

However, this discussion uses surveyed literature's point of view, ignoring the actual stakeholders of AI-based interview systems, leading to introduce bias. To avoid this limitation, in the next chapter, we will conduct an empirical study to validate our gained knowledge in the current chapter and link the functional requirement of each component that we discussed in this chapter and those that would be highly expected from potential end-users from such systems. The findings of the empirical study will help us to refine the functional requirements of each component; for each component, we will identify which of the already considered functionality should be prioritized and which functionality is required to be added, this time from the view of the potential end-user of an AI-based job interview system.

CHAPTER 4 RELATIONSHIP BETWEEN CONCERNS OF JOB MARKET STAKEHOLDERS AND THE PROPOSED REFERENCE ARCHITECTURE

In this chapter, we will conduct an empirical study to explore the concerns of the major roles involved in recruitment and job interviews, categorized into different topics and linked these concerns to our reference architecture's components. We will also investigate both the hot and challenging topics. This leads us to understand what components and which of their functionalities of the proposed reference architecture that are linked to the derived topics based on data-driven approaches, would be more important or difficult to improve or required to be added as new features from the public's point of view. In this way, we will address the RQ3 to RQ5 of this thesis.

This chapter is part of a manuscript that is being prepared for submission.

4.1 Methodology of the empirical study

In this section, we explain the approach that we adopted to answer the last 3 research questions. The followed approach is illustrated in Figure 4.1.

4.1.1 Collect data set

We first download the **Posts** dataset from the available data dump of the Workplace site of Stack Exchange [1] from 2011 to 2021. This is an anonymized dump of all user-contributed content for Workplace on the Stack Exchange network. The metadata in Posts dataset include 'Id', 'PostTypeId', 'CreationDate', 'Score', 'ViewCount', 'Body', 'OwnerUserId', 'LastActivityDate', 'Title', 'Tags', 'CommentCount', 'FavoriteCount', 'ContentLicense', 'LastEditorUserId', 'LastEditDate', 'ParentId', 'ClosedDate', 'OwnerDisplayName', 'LastEditorDisplayName', 'CommunityOwnedDate', 'AnswerCount', and the ID of the accepted answer of a post if the post is a question and has an accepted answer. The downloaded dataset has a total of 201,971 posts shared from March 2011 to June 2021.

Among other similar discussion forums and social media in this context, this website is a community with a large number of active members who are distributed in different geographical locations and have shared an extensive number of posts until now, and also the available several meta data and tags for each of the shared career-related posts on this website is not

[1] https://workplace.stackexchange.com/

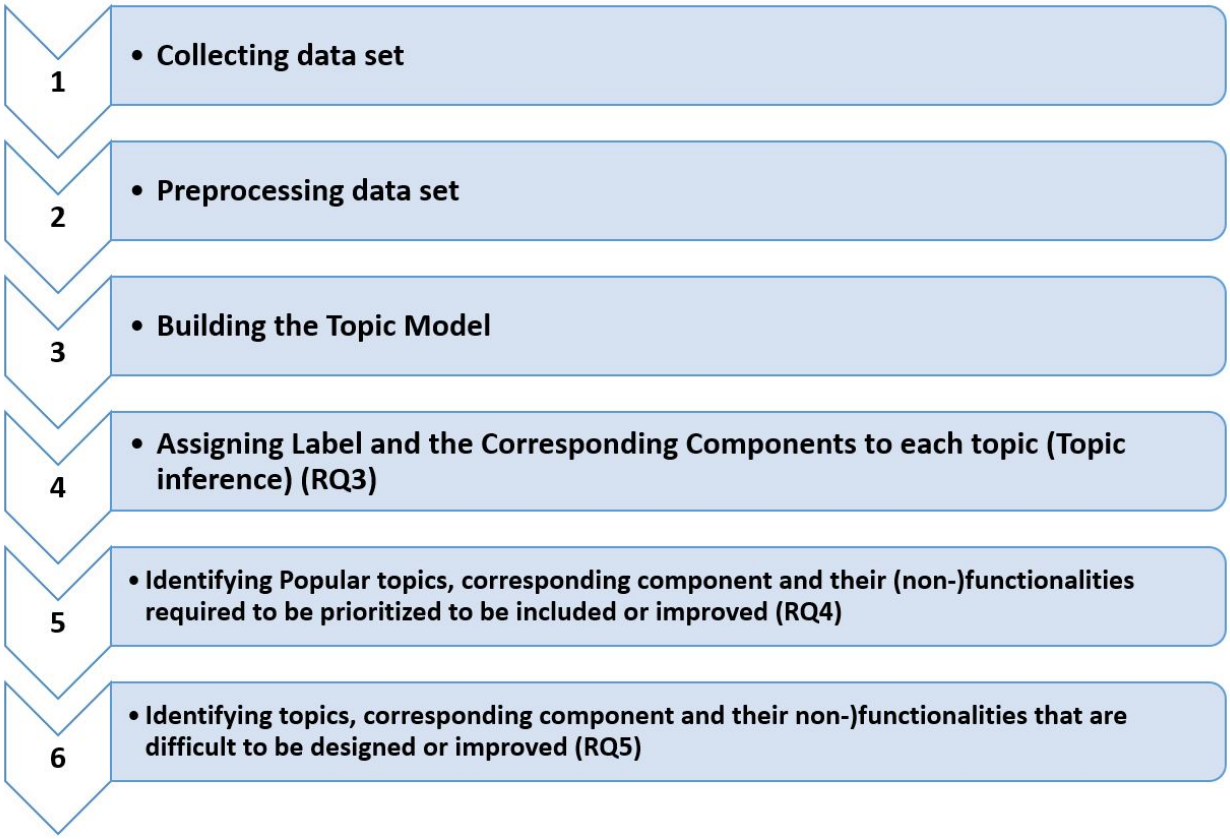

Figure 4.1 Overview of Empirical Study Methodology process

comparable with other similar question and answer forum. Further, given the significant popularity of Stack Exchange networks, we believe the shared post in Workplace site can be a good representative of the discussed issues among stakeholders in the job market.

From the collected dataset, we only keep those posts that have "interviewing" and/or "recruitment" tags, yielding 8016 post.

4.1.2 Data preprocessing

The next step is preprocessing the collected dataset for subsequent analysis. For every post, we first joined its title and body text to create one final body text. These created text documents still contain the HTML tags (for example, for presenting a paragraph, code snippet, URL and etc.), hence we removed all possible HTML tags (i.e., any code snippets surrounded by [HTML tag] .)$suchas < p >< /p > and < h >< /h >$.

We then removed the words identified as stop words [127]. These stop words are a set of commonly used words in any language, such as “the”, “a”, “an”, “in” and punctuation marks and non-alphabetical characters. These words should be ignored in modeling topics of a document. We also extend the list of words to be removed by adding the common words that are present in more than 80% of the documents (including ’job’, ’jobs’, ’company’, ’recruiters’, ’recruiter’, ’interview’, ’interviews’, ’employers’, ’employer’, ’interviewed’, ’interviewer’,’interviewers’). In this way, the resulting topics will focus more on those words related to the actual meaning of the documents (i.e., questions). Likewise, we removed letter accents and punctuation signs.

Moreover, we applied the stemming technique. Stemming is a process that reduces words to their stem, base, or root form. For example: “playing” is a word and its suffix is “ing”, hence if we remove “ing” from “playing” we will get the base word or root word, i.e., “play”.

Furthermore, we used n-grams in our analysis. An N-gram means a sequence of N words that occur frequently in the corpus. For example, “Medium blog” is a 2-gram (a bigram), “Write on Medium” is a 3-gram (trigram) and “Common Medium blog post” is a 4-gram. N-grams are a proven method to improve the quality of topic modeling [128]. Specifically, we implemented bi-grams and tri-grams which return 2 words and 3 words in sequence, respectively.

4.1.3 Topic Modelling

By following this technique, we extract the topics from the dataset of preprocessed posts by building several topic models using the MALLET toolkit [129] implements the Gibbs sampling

algorithms [130] for latent Dirichlet allocation LDA [131]. LDA is a widely-accepted method for fitting a topic model. It treats each document as a mixture of topics, and each topic as a set of frequently co-occurring words. This is achieved by categorizing the document into N topics after I iterations of grouping [132]. Since LDA is a probabilistic method, a topic would be a vector of word probabilities, and a document is a vector of topic probabilities. Thus, a topic that has the highest probability value is the most dominant topic for that document.

The number of topics (N) and the iterations grouping (I) are hyper-parameters that should be tuned to adjust the granularity of the discovered topics, thereby obtaining the optimal model [133]. In order to find the best number of topics N, we experimented with varying values of N and I. N ranges from 5 to 60 in increments of 5, while I varies from 500 to 3000 with increments of 500. These range settings are consistent with other similar studies such as [64], [133], and [134] that applied topic modeling on published posts in other contexts.

We also tuned two other two hyper-parameters: alpha and optimize interval. Alpha affects the sparsity of the topics, represents document-topic density [135]. In other words, it controls the prior distribution over topic weights in each document. Its defaults value is 1.0 number of topics prior; The higher the value of alpha, the more widely a topic will be distributed across the documents. Griffiths. et al chose alpha equal to 50 for a corpus of around 28K scientific documents and a vocabulary of 20K words [136]. However, we want the distribution of topics in each document to be sparse. That is, each document only represents a few topics, therefore alpha should be less than 1. We experimented with different ranges (i.e., [0.05,0.1,0.5,1,5,10]) of alpha to find which one suits our dataset.

The optimize interval hyper-parameter, on the other hand, turns on hyper-parameter optimization, which allows the model to better fit the data by allowing some topics to be more prominent than others. In this way, the model learns from itself as it goes along. According to documentations [129], optimization every 10 iterations captures a broad range of topics within the datasets while keeping them distinct from each other.

While tuning these hyper-parameters process, we build several models and choose the optimal model based on the highest *coherence score*, a measure that is used for assessing the quality of the learned topics, yielding meaningful relevant topics of medium level of granularity for the dataset. Topic coherence measures the relative distance between words within a topic. These measurements help to distinguish between topics that are semantically interpretable and topics that are artifacts of statistical inference [137].

4.1.4 Labeling topics

After building several LDA topic models and selecting the optimal models for the datasets, the next step is to manually assign a label to all obtained topics groupings (i.e., the output of running MALLET). The labels will be used to interpret and discuss the output of topic modeling and topic inference efforts. To be more specific, the labels will be utilized to describe existing or the required functionalities of an AI-based job interview system, as identified from discussions by major stakeholders of said system in the mined StackExchange posts from posters' perspective; we will explain the methodology to identify the corresponding component(s) in the next section.

Each topic document is a combination of keywords and each keyword contributes to its topic with a certain weight (i.e., probability value). Clearly, the higher probability value a keyword has, the higher contribution to that topic it has, and hence it has the higher impact on assigning the topic label. We used the most contributing keywords and corresponding probabilistic values as well as the top 10 to 15 most representative documents for each topic for the labeling purpose. To provide the right labels, this information was distributed between the labeling team (i.e., the author of this thesis and her two supervisors). The labeling team assigned labels after reaching an agreement based on discussions. The results of this topic labeling is part of the answer to **RQ3** of this thesis (i.e., defining functionalities).

4.1.5 Identifying the RA components the topics apply to

To do this task, we used topic inference which basically is testing the trained topic model on an unseen document. To meet this objective, we first created a text body for each component of our RA as unseen documents. To create these documents, we concatenated the title and abstract of all papers that we reviewed for each component in the Systematic Literature Review effort in chapter 3, which has been done to support our proposed reference architecture. Then, we put all of them (i.e., the concatenated titles and abstracts) into different documents separately for each component, as unseen documents to perform topic inference.

However, we could only follow this approach for the three components for which we were able to find the relative publications in the literature review (i.e., Questioning, Virtual Interviewer and Evaluation components). For the Feedback Provider component, we had to use the description that we provided for this component in our reference architecture as the unseen document for topic inference.

Then, using the trained topic model, we computed the probabilities of how likely these unseen

documents match to each topic. If the calculated probability of a topic to be mapped to a component is above 0.5, we linked that component to that topic. For example, if a component has 70% chance of mapping to topic A and 20% chance of mapping to topic B and 80% of mapping to topic C, we linked this component to topic A and C. It is also possible that a topic does not match to any component. In those cases, we do not consider it as a required functionality to be included in the RA. In this way, we analyzed the distribution of posts across the corresponding components. The achieved results along with the labels we assigned to each topic to represent the expected functionality from potential end-users of the system, will help us to answer RQ4 and RQ5. (Table 4.2).

4.1.6 Finding hot topics of discussion and corresponding component

To answer **RQ4**, we investigated which topics are more common among stakeholders in the interview pipeline, thereby their related functionalities required to be prioritized when designing or building an AI-based job interview system based on the RA of chapter 3 or needs to be added as a new feature to that system. To address this objective, we used three metrics that researchers applied in the literature: number of views [132], [138], [64], [139], [140], number of posts marked as favourite by users [132], [138], [139] and the post score [132], [141], [139]. The Score of a post is the total number of up-votes minus down-votes of that post. Evidently, a topic with a higher number of views, favorites counts and score is considered more popular.

4.1.7 Identifying challenging topics and their corresponding component

At this step, we identify which topics are found more challenging by posters and which are the related functionalities that would be hard to automate. To address this objective, we used parts of the available meta-data of posts to define two metrics. To define the average amount of time a question is open in terms of days, we used the creation date of the post, and the closing date of the post (see [142] and [64]). Also, some posts have an accepted answer that is referenced with the ID of the accepted answer, while this field might be empty for other posts, implying those posts were answered satisfyingly from the poster's point of view. Therefore, to define the second metric we calculate the percentage of questions in a topic without the accepted answers. This metric is also defined and used by [142], [138] and [64]. We believe that topics with the higher percentage of posts with an accepted answer and topics that have opened questions for a shorter time are less challenging. The findings will answer RQ5.

4.2 Results and Discussion

4.2.1 Post distribution across Job interview related topics

This section analyses the distribution of StackExchange posts across the topics (Table 4.2).

Table 4.1 Topic Labels and Corresponding Components

Topic No	Labels	Keywords
0	How to do follow-up contacts to the recruiters	email, call, time, send, day, contact, phone, follow, week, position
1	Dealing with job offers	offer, position, accept, apply, current, time, work, start, give, company
2	Candidate's physical appearance, emotional behaviors and social clues during interview sessions	question, answer, candidate, people, good, person, make, give, work, feel
3	Ethics in recruitment process	resume, contact, apply, position, work, send, information, find, candidate, reference
4	problem-solving questions	question, test, code, work, project, time, give, technical, problem, good
5	Impact of problems with the previous employer on candidate's evaluation	work, leave, current, time, month, manager, reason, boss, year, make
6	Impact of candidate's experience on candidate's evaluation	position, experience, work, role, team, skill, question, hire, apply, product
7	Resume content (experience and education)	work, experience, year, resume, internship, apply, position, study, degree, student
8	Payment and benefits	salary, offer, pay, position, give, make, work, employee, expect, question
9	Expenses for attending the in-person interview sessions	work, country, time, live, day, location, travel, pay, city, cost

After building several topic models we chose the optimal one with the highest coherence value equal to 58%. This model has 10 topics and the iterations grouping (I) for this model equals 2000. The alpha is 0.1 because Gensim calculates the symmetric value for alpha by dividing 1.0 by the number of topics in the model.

As figure 4.2 shows, the distribution of the posts among the 10 topics.

Evidently, most of the discussion happened on the questions about the topic labeled *"How to do follow-up contacts to the recruiters after the interview sessions"* (i.e., 15.56%). The post with the title of *"Unable To Reach Recruiter After Receiving An Invitation To Interview "*[2], which has the highest probability value (0.9728) to be grouped into this topic, is a good representative document.

Questions related to dealing with job offers (i.e., topic 1)and questions related to a candidate's physical appearance or emotional/social behavior during interview sessions (i.e., topic 2) were the next most dominant topic among all the discussions, respectively (roughly around 13% of total questions). For example, the most contributing post with the probability of belonging to the post on topic1 of 0.9767 was the post with the ID of 1203. This post is titled *"Should I get in touch with the recruiter with respect to accepting a job offer?"* [3]. Likewise, document 2459 is the most contributing one to the posts of topic 2 with 0.9769 probability. In this post, the posters asked: *"Will glasses make me look more professional?"*[4].

[2]https://workplace.stackexchange.com/questions/79817/unable-to-reach-recruiter-after-receiving-an-invitation-to-interview

[3]https://workplace.stackexchange.com/questions/55898/should-i-get-in-touch-with-the-recruiter-with-respect-to-accepting-a-job-offer

[4]https://workplace.stackexchange.com/questions/86438/will-glasses-make-me-look-more-professional

The fourth most dominant group of posts (10.83%) are grouped into the topic with questions related to ethics in the recruitment process (i.e., topic 3). A good example is the most contributing post (Prob. 0.9726), with as title *"How to avoid being cheated by some fake recruiters as a newbie?"*[5]. Here, the poster complains about being called by spammer headhunters susceptible to forgery.

The topic with questions related to problem-solving questions was the fifth dominant topic with 10.13% of the total number of posts. According to our dataset, the most contributing document to this topic is titled *"Mind went blank during interview coding test."* [6] with 0.9772 probability value.

Next is the topic that classified questions related to the impact of problems with a previous employer on a candidate's evaluation. This topic grouped slightly less than 10% of the posts, is making the sixth dominant topic across our whole dataset. For instance, in the post with ID 5478, the poster asked: *"I left my job because the company was failing financially and not meeting payroll, vendors, etc- how do I explain that in an interview?"*[7]. This is the most contributing document with the highest probability amongst others (i.e., 0.982).

The topic 7 on the impact of a job applicant's experience on their evaluation contains around 8% of all posts. The post with ID 3905 is a good example: This post titles: *"How to discuss my skills, methods and technologies That I used in my accomplished projects on my resume that is superior to the standard one required on the job specs?"*.

Next, topic 7 on posts related to resume content of the candidates grouped 7.47% of the posts.

Finally, only about 6% of the posts corresponded to topic 9 on concerns the expenses for attending in-person interview sessions or into topic 8 on questions about payment and benefits of a job. As a representative example of topic9, we chose post 2948 in which the poster asked *"Is it a reasonable expectation to ask the interviewer to cover the travel costs"*. For topic 8, post 4815 (prob. 0.977%) asked: *"Formula for computing the hourly rate for a software developer contractor?"*.

Figure 4.2 and Table 4.1 illustrate the distribution of posts across the 10 topics with corresponding labels.

[5]https://workplace.stackexchange.com/questions/79669/how-to-avoid-being-cheated-by-some-fake-recruiters-as-a-newbie

[6]https://workplace.stackexchange.com/questions/91885/mind-went-blank-during-interview-coding-test

[7]https://workplace.stackexchange.com/questions/81530/i-left-my-job-because-the-company-was-failing-financially-and-not-meeting-payrol

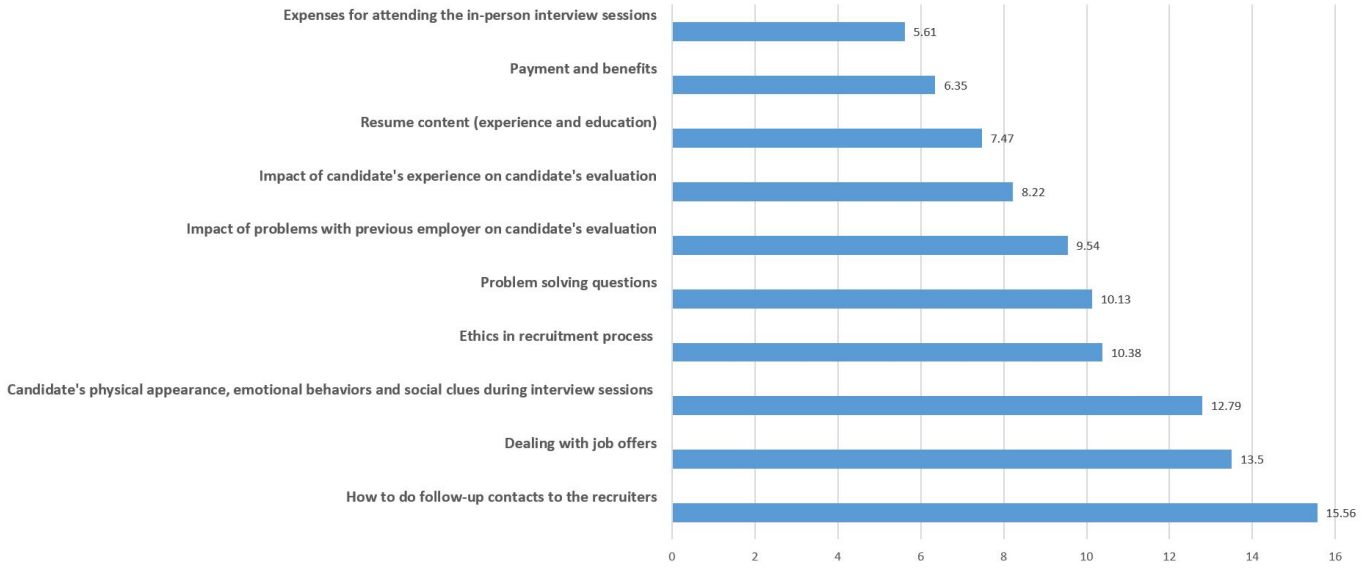

Figure 4.2 Topics and percentage of their Posts

4.2.2 Corresponding components of the reference architecture to the Job interview related topics

The topics identified in the previous section cover various aspects of the in-person interview process. Some, like the topic on expenses of attending job interviews, actually cover exactly the reason why e-recruitment systems (included AI-based job recruitment systems) have been conceived. Other topics provide concerns by job candidates and interviewers that potentially might be relevant for designers of AI-based job recruitment systems. As such, this section, in trying to answer RQ1, links each topic to the four RA components identified in chapter 3 in order to understand which topics (and hence StackExchange posts) might be relevant for the designer of which RA component.

As we explained in the Methodology section, we map topics to RA components using topic inference of the topic model trained previously on 4 documents representing each component. We linked a component to a topic if the calculated probability value is above 0.5. This means that a topic potentially could be mapped to more than one component, or even to no component at all.

As shown in table 4.2, 4 topics are linked one components. For example, topic 3 about ethics in recruitment, is only linked to the Evaluation component (Prob. 0.53). Likewise, evidently, topics that are about how to do follow-up contacts to the recruiters (i.e., topic 0), expenses for attending the in-person interview sessions (i.e., topic 9) and candidate's physical

appearance, emotional behaviors and social clues during interview sessions (i.e., topic 2), all are most probable to correspond to the Virtual Interviewer component (i.e., with over 60 percent of probability value). As these all deal with communication between the candidate and the interviewer, these mappings make sense.

Also, we found that one of the identified topics corresponds to two components. For example, the probabilities for topic 8 of matching to the Questioning and Evaluation components are 0.52 and 0.67, respectively. This makes sense, because the corresponding questions should be formulated by the Questioning component and the given answers should be analysed in the Evaluation component.

Moreover, 4 of the topics are linked to 3 components; For example, topic 7 about resume content is linked to the Questioning, Evaluation and Feedback provider components with probability values closed to 1 (i.e., 0.92, 0.89 and 0.73 respectively). This makes sense, because these 3 components are responsible to generate/look-up questions about resume content, evaluating the candidate with respect to their resume and reflecting the information of the resume in the final report, respectively.

Similarly, we found that the same components map strongly to topic 4 on problem-solving questions, with probability values over 0.5 (i.e., 0.72, 0.68 and 0.67, respectively). This also makes sense, because the same components are responsible to generate/look-up problem-solving question, evaluating the given answers of the candidate to those questions and reflecting the actual answer (i.e., not only the final result) in the final report.

Topic 6 on the impact of candidate's experience on their evaluation is more likely to be linked to the same three components as well. This makes sense too, as these components are responsible to generate/look-up questions with respect to the candidate's experience, evaluating their given answers to those questions and reflecting the results in the final report.

Finally, evidently, the group of posts that concern the impact of problems with a previous employer on a candidate's evaluation (i.e., topic 5) is more likely to map to the Virtual interviewer, Evaluation and Feedback Provider components (i.e., probability value above 0.5), in contrast to the Questioning component with probability value equals to 0.04.

Overall, we found that from the 10 extracted topics with some overlaps, 4 of them were linked to the Questioning component, 4 of them linked to the Virtual interviewer component, 4 of them linked to the Feedback Provider component and 6 of them were linked to the Evaluation component, suggesting that more concerns and requirements being discussed about the functionalities of the Evaluation component than others.

In the next sections, using the assigned labels to each topic (i.e, representative of functionality

requirements), we will discuss our findings on which of these functionalities are required to be prioritised or to be added while building the RA and are more challenging to develop or to add from the perspective of the stakeholders.

Table 4.2 Posts Distributions across Topics- Probability values of belonging 4 components to each topic (RQ3)

Topic No	% of Posts	Labels	Corresponding Components	Match prob. With Questioning Component	Match prob. with Virtual Interviewer Component	Match prob. with Evaluation Component	Match prob. with Feedback Provider Component
0	15.56	How to do follow-up contacts to the recruiters	**Virtual Interviewer**	0.15	**0.65**	0.33	0.12
1	13.5	Dealing with job offers	**No Corresponding Components**	0.08	0.08	0.09	0.16
2	12.79	Candidate's physical appearance, emotional behaviors and social clues during interview sessions	**Virtual Interviewer**	0.09	**0.91**	0.28	0.49
3	10.38	Ethics in recruitment process	**Evaluation**	0.41	0.03	**0.53**	0.34
4	10.13	problem-solving questions	**Questioning/Evaluation/Feedback provider**	**0.72**	0.32	**0.68**	**0.67**
5	9.54	Impact of problems with previous employer on candidate's evaluation	**Virtual interviewer /Evaluation/Feedback provider**	0.04	**0.54**	**0.58**	**0.53**
6	8.22	Impact of candidate's experience on candidate's evaluation	**Questioning/Evaluation,/Feedback provider**	**0.78**	0.17	**0.77**	**0.51**
7	7.47	Resume content (experience and education)	**Questioning/Evaluation/Feedback provider**	**0.92**	0.22	**0.89**	**0.73**
8	6.35	Payment and benefits	**Questioning/Evaluation**	**0.52**	0.09	**0.67**	0.36
9	5.61	Expenses for attending the in-person interview sessions	**Virtual Interviewer**	0.22	**0.61**	0.33	0.48

4.2.3 Popular discussed topics and the corresponding components

In order to answer **RQ.4**, as discussed in the Methodology section, we used 3 parts of meta data as indicators of the popularity of a post, i.e., View Count, Score and Favorite Count. We computed the average values of these meta data for all posts within each topic (Table 4.3). Since we also know for each topic the corresponding component of the proposed reference architecture, we can understand which component(s) have more functional requirement concerns being discussed by the major stakeholders of the system.

First, we found that questions related to a candidate's physical appearance, emotional behavior and social clues during interview sessions (i.e., topic 2) have the highest value across all three metrics. This suggests that its corresponding component, the Virtual Interviewer, should be designed and implemented and appropriately, taking into account these concerns. For example, the body language of the candidate would be expected to be captured elaborately by this component or this component is expected to capture the candidate's attire, possibly to be compared with the required interview sessions' dress code, later in the evaluation phase. Also, we concluded that designing the related functionalities that have already been considered into the RA should be prioritized by the designers.

If we solely consider the number of views, topic 0 on how to do follow-up contacts to the recruiters has the second most number of viewed questions, averagely. This topic is linked to the Virtual Interviewer component. This finding along with the assigned label to this topic is suggesting that posters (including the recruiters and candidates) would be interested in

adding some new functionalities to the Virtual Interviewer component that enable the users to keep the lines of communication open until the end of the recruitment process and to interact with each other (i.e., recruiter and the interviewee) even after the interview session.

On the other hand, if we only consider the average score of posts, we find that questions related to expenses for attending the in-person interview sessions have the second larger average score among other groups of questions. Therefore, it can be said that the whole concept of AI-based job interview system and in particular, the corresponding component to this topic (i.e., Virtual Interviewer) would be very effective to mitigate this type of concern among the stakeholders.

Finally, considering only the average favorite counts of posts, topic 4 on problem-solving questions turned out to have the second higher value for this metric. The label of this topic leads us to conclude that people would expect to prioritize the existing proposed functionalities of asking proper problem-solving questions in the Questioning component and are concerned about how the Evaluation component is assessing the answers given to such questions. Also, they would like to include specifically the asked problem-solving questions and the given answers in the evaluation report generated by the Feedback provider component, as a new feature for this component.

We have already considered features that make the operational data available for the recruiters by the Feedback provider component. However, evidently, including asked problem-solving questions and the given answers in the final report would be highly expected, compared to other elements of the operational data. Probably, because automatic evaluation only considers the final achieved results by the candidate, while including the whole given answer into the report would help the recruiter to consider the approach that the candidate has taken for solving the given problem as well. This would lead the users of the system to avoid bias introduced by relying only on AI solutions, instead involving human intelligence into the evaluation process.

Table 4.3 Hot discussed topics and corresponding components (RQ4)

Topic No	Labels	Corresponding Components	Avg View Count	Avg Score	Avg Favorite Count
0	How to do follow-up contacts to the recruiters	**Virtual Interviewer**	***4606.37***	4.42	0.47
1	Dealing with job offers	**No Corresponding Components**	3952.83	6.82	0.78
2	Candidate's physical appearance, emotional behaviors and social clues during interview sessions	**Virtual Interviewer**	**7647.19**	**16.79**	**2.40**
3	Ethics in recruitment process	**Evaluation**	3146.77	7.24	0.95
4	problem-solving questions	**Questioning/Evaluation/Feedback provider**	4033.81	11.27	***1.83***
5	Impact of problems with previous employer on candidate's evaluation	**Virtual interviewer /Evaluation/Feedback provider**	4419.58	12.93	1.68
6	Impact of candidate's experience on candidate's evaluation	**Questioning/Evaluation,/Feedback provider**	2295.10	8.56	0.99
7	Resume content (experience and education)	**Questioning/Evaluation/Feedback provider**	3508.77	7.86	1.30
8	Payment and benefits	**Questioning/Evaluation**	4604.47	10.32	1.48
9	Expenses for attending the in-person interview sessions	**Virtual Interviewer**	4390.63	***13.06***	1.11

4.2.4 Challenging discussed topics and corresponding component

In order to answer **RQ.5**, we used three parts of meta-data of posts including the presence of an accepted answer and the creation and closing date of the post. Using this information, we defined and calculated two metrics; i.e., the percentage of questions within a topic that have no accepted answers as well as the average amount of time (in terms of days) that questions within a topic are open (Table 4.4).

Our findings revealed that questions related to expenses for attending the in-person interview sessions have been open for longer periods of time than others. This is suggesting that these expenses are a big concern among the posters too . Thus, we concluded that e-recruitment can remove the need for on-site interviews entirely. Apart from shifting to e-recruitment, obviating the need for in-person interviews, the StackExchange posters also expect designers of such systems to ensure low entrance costs for participating in interviewing with a virtual character (i.e., hardware and software requirements for attending the interview session). This topic is found to be popular as well in the previous section, this can be seen as a new both popular and demanding functional requirement for the Virtual Interviewer component.

The second longest open questions are those on ethics in the recruitment process. The observations about different geographical locations of the posters from the "Location" field of the posters available in the "User" data set available in the data dump (which is associated with the "Post" data set through the "User Id" feature) suggests that posters have different cultures and backgrounds and due of such variation, most of the questions in this topic have never received a sufficiently satisfying answer from the original poster's view. This might be a justification for this finding, however, future work is needed to validate this hypothesis. This suggests that such variation in cultural background should be taken into consideration efficiently as an additional feature for the Evaluation components that is corresponding components to this topic, which can be said as a tricky subject.

On the other hand, (excluding the topic 1 which does not map to any of our components in the topic inference step), posts related to payment and benefits have the highest percentage of questions without an accepted answer (53.44%). This is might come from the fact that posters have different expectations based on their background, experience and costs (again possibly due to their different geographical locations) such that they cannot reach a consensus in terms of an accepted answer. Also, from the recruiters' point of view, payments would vary based on several criteria (i.e., available funds, local regulations, etc). Likewise, salary-related questions are weighted differently according to job domains. In other words, there is hardly a single answer that fits all questions. Therefore, we believe that the stakeholders would expect that the some new functionalities in corresponding components (i.e., Questioning and

Evaluation) should be added to the RA to consider such disparity between the candidates and job domains and recruiters' financial constraints while asking about the salary and evaluate the give answers.

Next, the posts in topic 5 (i.e., the impact of problems with the previous employer on candidate's evaluation) have the second highest percentage of posts without an accepted answer (52.16%). This means that posters find it hard to provide solutions or accept the proposed solution by others to such problems. This implies that posters would expect that the Virtual Interviewer component accurately captures the verbal/non-verbal answers when they are explaining their experience with their previous employer (we have already considered such feature for this component in the RA), that the Evaluation component considers the provided explanations into the assessment process efficiently and that such experiences would be expected to be included into the final report of the interview session to be judged by a human. These are found to be the new features for these two components of the RA.

4.3 Refined functional requirements for each component of the proposed RA

From the findings of the SLRs in the chapter 3 as well as that of the empirical study, we refined the functional requirements for each components as summarized below:

1. Questioning component
    - Designing predefined questions (* defining proper predefined problem-solving questions need to be prioritized, learnt from the empirical study).
    - Automatically generating questions.
    - Providing predefined or automatic generated means for validation of the quality of answers for predefined or automatic generated questions. respectively
2. Virtual interviewer
    - Performing Virtual interaction with the candidate.
    - Capturing the verbal given answers of the candidate.
    - Capturing the nonverbal cues and social skills of the candidate (* need to be prioritized, learnt from the empirical study).
    - Providing features to enable candidates to keep the lines of communication open until the end of the recruitment process (* new feature identified in the empirical study).

3. Interviewee Evaluation

    - Processing the given verbal and non-verbal answers to calculate a cumulative evaluation result
    - Considering differences in cultures and backgrounds between the candidates while evaluating them (* new feature identified in the empirical study)
    - Considering the provided explanations about the problems with the previous employer while evaluating candidates (* new feature identified in the empirical study)
    - Store and manages the operational data

4. Feedback provider

    - Providing the feedback reports includes the evaluation calculated by the system to the recruiter
    - Providing the operational data, in particular, the approaches are taken by the candidate to solve problem-solving questions (* new feature identified in the empirical study)
    - Providing the recruiter with any explanations about the problems with the previous employer in the final report of candidates (* new feature identified in the empirical study)

5. Interview Controller

    - Coordinating components to execute the right component's functionality at the right moment
    - Selecting the category of the two types of questions to be asked (predefined or automatically generated one).
    - Making decisions about when to terminate the interview session.

4.4 Summary

In this chapter, we aimed refine the identified functionalities for each component. To do this, we conduct an empirical study to investigate the opinions and concerns of the major roles involved in recruitment and job interviews, i.e., the job candidate and interviewers, being the potential end-users of such AI-based system. The findings lead us to discuss additional functionalities that need to be included in a such system as well as interesting/challenging

Table 4.4 Challenging discussed topics and corresponding components (RQ5)

Topic No	Labels	Corresponding Components	Avg.Openi Days	% Posts without Accepted Answer
0	How to do follow-up contacts to the recruiters	**Virtual Interviewer**	1119.87	51.00
1	Dealing with job offers	**No Corresponding Components**	1235.51	*53.97*
2	Candidate's physical appearance, emotional behaviors and social clues during interview sessions	**Virtual Interviewer**	1329.84	42.63
3	Ethics in recruitment process	**Evaluation**	***1434.50***	50.69
4	Problem-solving questions	**Questioning/Evaluation/Feedback provider**	1141.12	42.86
5	Impact of problems with previous employer on candidate's evaluation	**Virtual interviewer /Evaluation/Feedback provider**	1208.68	***52.16***
6	Impact of candidate's experience on candidate's evaluation	**Questioning/Evaluation,/Feedback provider**	1343.30	44.76
7	Resume content (experience and education)	**Questioning/Evaluation/Feedback provider**	1240.41	51.25
8	Payment and benefits	**Questioning/Evaluation**	1350.77	**53.44**
9	Expenses for attending the in-person interview sessions	**Virtual Interviewer**	**1471.28**	41.33

existing functionalities that need more analysis to design to give them more priority while designing the RA, all in the respective components of the RA in chapter 3.

We learnt that, while some topics and the functionalities of their corresponding component(s) are both important to include or improve, they are expected to be difficult to design.

For example, the topic that grouped posts about the physical appearance, emotional behaviors and social clues of the candidate during interview sessions, is found to be the hottest topic as it had the highest values for all of the 3 metrics we defined for RQ4. This finding leads us to conclude that for its corresponding component, the Virtual Interviewer, the issues discussed for this topic have a high priority to be considered in the design of this component. In other words, among the functionalities of this component, the posters value improving the automated recognition of the physical appearance of a candidate as well as emotional and social behaviors accurately and efficiently during the interview by this component, therefore, designers/architects should consider that very well.

Considering the average favourite counts metric, we found that the posts that were about problem-solving questions had the second highest value. This suggests that both asking and evaluating proper problem-solving questions are major functionalities of the Questioning and Evaluation components, respectively, that expected to be prioritized. This strengthens the need for these functionalities that we have proposed RA for these two components in the previous chapter and Furthermore, these posts also provide suggestions for the evaluation of answers by the Feedback provider component (i.e., third corresponding component to this topic). We reached this conclusion because the automatic evaluation usually only considers the achieved results for such questions while knowing how the candidate approaches a complex problem can be more important than the actual answer (as suggested in many of the posts). This would avoid the bias introduced by relying only on AI solutions by involving human

intelligence in the evaluation process.

Considering the last popularity metric, the number of views, questions that concern about following-up contacts with the recruiters had the second highest value. Most of the discussions were around the posts on this topic in our data set. This topic that are linked to Virtual Interviewer components, had the second most viewed questions. Therefore it can be concluded that the majority of potential users would be interested in including features to the Virtual Interviewer component that enable them to keep the lines of communication open until the end of the recruitment process in order to interact with each other even after the interview session. And that is a new required feature for this component that we had not considered in the RA.

Also, posts on payment and benefits have the first highest percentage of questions without an accepted answer (i.e., one of the metrics for identifying challenging topics and functionalities of their corresponding component). We believed that this might be because of the fact that posters have different expectations based on their diverse background and their job domains such that they hardly can accept an answer as the satisfied one. Also, from the recruiters' point of view, payments would vary based on their financial constraints. This suggests that the stakeholders would expect that the relevant functionalities in corresponding components (i.e., Questioning and Evaluation) consider such disparity between the candidates when asks/being asked about the expected salary and upon evaluation of given answers. We also did not consider asking questions on the salary and evaluation of the given answers in our proposed RA, thus is an additional required feature for these components that designers should take into account while building the architecture.

Finally, from findings of the SLRs in previous chapter as well as that of the empirical study, we refined and discussed the functional requirements for each component.

CHAPTER 5 CONCLUSION

5.1 Summary of Studies

In this study, we proposed a comprehensive reference architecture (RA) for an AI-based job interview automation system. We extracted the main components of this RA, their sub-components, their functionalities through a systematic literature review. We also explored and discussed the challenges in designing and developing different components of the proposed reference architecture, first based on the findings of the literature review study and then based on the findings of the analysis of real evidence.

To support our proposed reference architecture, we conducted a systematic literature review study to survey the publications in the last decade in the context of automation of the job interview process. Using a database search and snowballing approach to collect the relative papers and analyse them through open coding technique, we found the main practices that have been simulated to achieve an AI-based job interview system, namely the questioning activity, virtual communication with the interviewee and evaluation of the applicant based on the result in the interview session of this pipeline. Also included in the proposed RA are a controller component due to architectural requirements and a feedback provider component to involve the human intelligence and to avoid bias introduced by relying only on AI technologies in human-computer interaction systems (**RQ1**). Next, for each component we conducted a separate SLR study to extract the sub-components, functionalities and the methods, models, techniques and technologies that researchers proposed or applied to automate them (**RQ2**). From the SLRs, we learned that different AI-based methods can be applied in designing some of the identified functionalities of these components.

Moreover, based on the findings and gaps in the literature, we discussed the open issues to design and develop the derive components and their functionalities, leading to open research windows for future studies.

Many of the discussed challenges show the need for research studies to propose appropriate methods to design/develop each component/sub-component from the recruiters' point of view. The invalid assumptions suggest there is a gap between some of the achievements and assumptions of human resource researchers and those of the researchers in the artificial intelligence field, as several of the reviewed proposed models are not applicable in real situations due to invalid/inadequate assumptions of their authors. These include the open issues related to the evaluation of the integrated verbal and non-verbal answers of candidates,

based on the whole interview discussion, making a decision on termination or continuing an interview session, asking the right question at the right time and ranking questions. We also discussed how different parties including researchers and companies that produce AI-based e-recruitment systems, can benefit from our work.

After documenting the reference architecture of an AI-based job interview automation system, we validated our knowledge for defining the functional requirements through an empirical study of the concerns of the major roles involved in the recruitment/job interview pipeline, which can suggest missing functionalities or priorities for an AI-based job interview automation system. To meet this objective, we applied topic modeling to categorize the concerns of the stakeholders in the recruitment/job interview pipeline shared in the Workplace website of Stack Exchange in the last decade. We then labeled each topic to represent the expected functionality of an AI-based job interview automation system from the viewpoint of potential end-users (**RQ3**). Next, using topic inference we linked each topic to the components of the proposed reference architecture. Also, we measure the popularity and challengingness of each group of posts through several metrics we formulated using the available meta-data of posts.

Our findings reveal that there are some topics that provide functionalities that we have not included in our proposed RA in chapter 3 but are needed to be added into the RA as additional features. On the other hand, our findings also strengthen the need of some functional requirements of the components in our proposed RA from the viewpoints of the posters (i.e., potential users of the AI-based job interview system).

In addressing the **RQ4**, we found functionalities that enable the users to keep the lines of communication open until the end of the recruitment process and to interact with each other are popular and required to be added to the Virtual Interviewer component Also, we learned that the potential users of the system are more favoured toward designing the functionalities related to recognizing accurately and efficiently the physical appearance of interviewees as well as their emotional and social behaviors that have already been considered in the RA. Likewise, a new feature is found to be interesting to be included in the Feedback provider component to provide the recruiters in particular, with the actual answer of the candidate which includes the approaches they have taken toward solving the problem.

In addressing the **RQ5**, we learned that dissimilarities in salary expectations in different job domains, the disparity between candidates' backgrounds and variation in the recruiters' financial constraints should be considered in Questioning and Evaluation when generating questions about the expected salary and evaluation of given answers as new hard-to-design features in these two components. Furthermore, we have not considered the disparity in cultural backgrounds in proposing the functionalities in the Evaluation component. We argued

that the ethical issue turns to be a controversial topic possibly due to cultural backgrounds variation of posters located in different geographical locations; however, future work is needed to validate these hypotheses. Thus, we cautiously concluded that such variation required to be considered in this component's functionality as an additional hard-to-design feature.

And finally, we concluded that while e-recruitment can remove the need for in-person interviews completely, posters also expect the designers of e-recruitment systems and in particular, AI-based job interview systems, to ensure low costs for participating in interviewing with a virtual character (i.e., low hardware and software requirements for attending the interview session). Thus, we suggested it as a new requirement that is both popular and demanding for the Virtual Interviewer component of the proposed RA. This finding addresses both **RQ4** and **RQ5** of this research.

5.2 Limitations

5.2.1 Threats to validity of the Systematic Literature Review

The first threat is because of the choice of data base to collect the relevant papers; we referred to Engineering Village[1] database as the data source. This database provides access to 12 engineering literature and patent databases to cover a wide range of trusted engineering sources. However, there are other databases that include publications in the AI fields and job interview contexts, that we might miss some related papers published there. Also, we collected the papers that have been published in the last decade. Although there might be some relevant papers in the context of automation of the job interview process published before 2011, we decided to focus on the most recent ones. In formulating the search queries we included the most relevant keywords or phrases or any of their derivative forms, but it is possible that there are other keywords that we missed to include. To mitigate this limitation, we applied snowballing to re-cover any possible missed publications.

5.2.2 Threats to validity of proposed reference architecture for AI-based job interview systems

One of the most important limitation is that we extracted the high-level components and their sub-components as well as their respective functionalities using the knowledge that we gained through the SLR study. Grey literature such as government or policy reports and commercial documents are not included in the Systematic Literature Review, which may

[1] https://www.engineeringvillage.com/search/quick.url

bias our perspectives since scientific publications usually have not been under evaluation in real situations.

To mitigate this limitation, for each component we included the "Challenges and Invalid assumption" sections to discuss the assumptions that have been made in the reviewed publications but are not valid or applicable in real situations as well as the designing/implementation challenges. We even took a further step by conducting an empirical study to analyze the views of major roles involved role involved in the recruitment and job interview pipeline they have shared publicly during the last 10 years. This leads us to understand the expectations of potential users of an AI-based job automation system, in an objective fashion.

Furthermore, the number of publications that we reviewed was rather low for some of the components. This is because we only focused on the research efforts that proposed a solution/model for designing/implementing the functionalities of that component in the "job interview" context. Last but not least, the proposed reference architecture, including the required components are not validated.

5.2.3 Threats to validity of the study mapping the concerns of job market stakeholders and the proposed reference architecture

The first threat corresponds to collecting our dataset from the Workplace website site on StackExchange to investigate the concerns of job applicants and recruiters. While there might be other discussion forums and social media in this context, the Workplace website is a leading community with millions of active members in different geographical locations with extensive number of questions have been asked until now, and also several available meta data (including 'Id', 'AcceptedAnswerId', 'CreationDate','Score','ViewCount', 'Body','OwnerUserId', 'LastActivityDate', 'Title', 'Tags','AnswerCount', 'CommentCount', 'FavoriteCount', 'ContentLicense', 'LastEditorUserId', 'LastEditDate', 'ParentId','ClosedDate', 'OwnerDisplayName','LastEditorDisplayName','CommunityOwnedDate') as and tags (e.g., interviewing, professionalism, job-search , etc.) for each of the shared career-related posts is not comparable with other similar question and answer forum. Hence, given the significant popularity of Stack Exchange networks, we believe the Workplace site can be a representative source of the discussed issues among stakeholders in the job market.

The next threat refers to selecting only the posts with interviewing and/or recruitment tags from the whole data dump to be analyzed, while there might be other posts in a similar context. An alternative approach would be reading all posts manually to sift out irrelevant posts from the dataset. However, due to time constraints, we had to define criteria to select the most relevant posts.

Likewise, in order to have an insight of the expectations of the potential end-users of an AI-based job interview system, we relied on the descriptions the posters provided about the in-person fashion of the recruitment process. While they might have different opinions when it comes to an intelligent recruitment/job interview system., other techniques of data collection such as interviews, surveys and questionnaires could be applied to collect more related qualitative data. However, again time and budget constraints prevent us to use these techniques. Also, we believe while our approach avoids the disadvantages of applying other mechanisms for qualitative data collection such as biased and dishonest answers or costs for setting up the experiments, would provide access to an extensive amount of data across a long period to be analyzed.

Manual labeling of topics to represent the expected functionalities is another threat to the validity of the achieved results. This is because this phase of topic modeling studies is mostly interpretive in nature and there is no tool to automate the labeling task. To mitigate this threat, considering the most contributing keywords of each topic as well as manually reading through the top most representative documents for that topic, the authors discussed and assigned a label that appropriately maps to the topic. Similar mitigation solutions have been previously applied by [132], [139] and [143].

Furthermore, the range of values we selected to tune the hyper-parameters (i.e., number of topics, iterations grouping and alpha) to choose the optimal Mallet topic model from the built ones, is another threat to the validity of our achieved results. To avoid selecting these ranges subjectively, we followed the approaches that other researchers with similar objectives applied in the literature (e.g., [143], [133], [136]). In the data pre-processing step our choice of stop words could be biased and could affect the results. To mitigate this problem, we extended the list of words to be removed from the dataset by including the set of words that are presented in over 80% of the dataset in order to give more focus to the important words. Although this percentage can be tuned through examining the texts manually, it is a time-consuming task.

5.3 Future Research

In chapter 3, we included "Challenges and Invalid assumption" sections for each component to open future research opportunities. In these sections, we discussed in detail the open issues to design and develop different RA components and their respective sub-components that are ignored in the literature and need to be addressed in viable AI-based job interview systems. Some of these challenges refer to the need for collecting standard and large-scale datasets for populating and training the system. Other challenges indicate the need for research studies

to propose appropriate methods to design/develop each component/sub-component from the recruiters' point of view. Our literature review shows that many of the surveyed proposed models are not applicable in real situations due to invalid/inadequate assumptions of their authors and hence, require further studies.

For instance, in reality, several criteria, like the number of questions, duration of a session, evaluation of the candidate's given answers to previous questions, etc. involve making a decision on termination or continuing an interview session, while this aspect is ignored in the literature that we surveyed. Also, there must be methods to look up the appropriate questions in the predefined questions pool based on the context of the interview. Regarding the automatic question generation, a method that takes the whole discussion into consideration in order to generate the next follow-up question is lacking in existing works. Moreover, proposing a methodology for evaluation of the given answers to open questions and to the generated follow-up questions needs further studies. Likewise, ranking the questions in terms of their importance as well as ranking the given answers in terms of their correctness in the candidate's cumulative evaluation phase is ignored in the literature. Finally, identifying the non-verbal cues that influence the final evaluation of the candidates and proposing a method to analyze and integrate the analysis results into cumulative evaluation computation is another challenging issue.

Moreover, there are areas in the empirical study conducted in Chapter 4 that should be addressed in future research. Other techniques of qualitative data collection such as interviews, surveys and questionnaires should be applied to gather information from the stakeholder to have a better insight of the requirements of an AI-based job interview automation system from the real end-users' perspectives as well as validate our findings in this study. Analyzing the collected data, regardless of the data source, can be improved in the future to provide more detailed results (i.e., functional and non-functional requirements). For example, applying techniques that minimize the risk of bias and subjective conclusions should be applied in the future. This would lead to the design/implement an AI-based job interview automation system that meets the users expectations.

On top of that, although we conducted an empirical study to validate our knowledge about the functionalities that we considered for each component of the proposed RA, the reference architecture itself is not evaluated. Therefore, some future work ideas could be defining the quality attributes of the reference architecture, defining metrics and methods to measure those attributes as well performing the evaluation activities. This objective can be met by creation and study of a series of real system architectures derived from the reference architecture, as a similar approach followed by [144] in which their proposed reference architecture

is used for mapping two existing Farm Software Ecosystems.

REFERENCES

[1] J. Kasundi and G. Ganegoda, "Candidate recruitment based on automatic answer evaluation using wordnet," in *2019 International Research Conference on Smart Computing and Systems Engineering (SCSE)*. IEEE, 2019, pp. 29–37.

[2] A. E. Barber, *Recruiting employees: Individual and organizational perspectives*. Sage Publications, 1998.

[3] A. I. Huffcutt and S. S. Youngcourt, "Employment interviews," *Applied measurement: Industrial psychology in human resource management*, pp. 181–199, 2007.

[4] J. Diekmann and C. J. König, "Personality testing in personnel selection," *Employee Recruitment, Selection, and Assessment: Contemporary Issues for Theory and Practice; Psychology Press: London, UK*, p. 117, 2015.

[5] P. Karolina. (2018) 65+ recruitment stats HR pros must know in 2018. [Online]. Available: https://devskiller.com/65-recruitment-stats-hr-pros-must-know-2018/

[6] A. Jäckle, P. Lynn, J. Sinibaldi, and S. Tipping, "The effect of interviewer personality, skills and attitudes on respondent co-operation with face-to-face surveys," ISER Working Paper Series, Tech. Rep., 2011.

[7] R.-R. Yu, Y.-S. Liu, and M.-L. Yang, "Does interviewer personality matter for survey outcomes? evidence from a face-to-face panel study of taiwan," in *World Association for Public Opinion Research Annual Conference, Amsterdam, The Netherlands*, 2011.

[8] J. Li, M. X. Zhou, H. Yang, and G. Mark, "Confiding in and listening to virtual agents: The effect of personality," in *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, 2017, pp. 275–286.

[9] J. Annette, P. Lynn, J. Sinibaldi, S. Tipping *et al.*, "The effect of interviewer personality, skills and attitudes on respondent co-operation with face-to-face surveys," Institute for Social and Economic Research, Tech. Rep., 2011.

[10] S. K. Jha, S. Jha, and M. K. Gupta, "Leveraging artificial intelligence for effective recruitment and selection processes," in *International Conference on Communication, Computing and Electronics Systems*. Springer, 2020, pp. 287–293.

[11] S. Laumer, A. Eckhardt, and T. Weitzel, "Electronic human resources management in an e-business environment," *Journal of Electronic Commerce Research*, vol. 11, no. 4, p. 240, 2010.

[12] S. Lang, S. Laumer, C. Maier, and A. Eckhardt, "Drivers, challenges and consequences of e-recruiting: a literature review," in *Proceedings of the 49th SIGMIS annual conference on Computer personnel research*, 2011, pp. 26–35.

[13] J. Anthony, "Technology is an enabler not a replacement for HR functions," 2014.

[14] E. Parry and S. Tyson, "An analysis of the use and success of online recruitment methods in the uk," *Human Resource Management Journal*, vol. 18, no. 3, pp. 257–274, 2008.

[15] K. Shubham, E. P. Kleinlogel, A. Butera, M. S. Mast, and D. B. Jayagopi, "Conventional and non-conventional job interviewing methods: A comparative study in two countries," in *Proceedings of the 2020 International Conference on Multimodal Interaction*, 2020, pp. 620–624.

[16] P. Guchait, T. Ruetzler, J. Taylor, and N. Toldi, "Video interviewing: A potential selection tool for hospitality managers–a study to understand applicant perspective," *International Journal of Hospitality Management*, vol. 36, pp. 90–100, 2014.

[17] M. Langer, C. J. König, and K. Krause, "Examining digital interviews for personnel selection: Applicant reactions and interviewer ratings," *International journal of selection and assessment*, vol. 25, no. 4, pp. 371–382, 2017.

[18] A. B. Holm and L. Haahr, "E-recruitment and selection," in *e-HRM*. Routledge, 2018, pp. 172–195.

[19] U. C. Okolie and I. E. Irabor, "E-recruitment: practices, opportunities and challenges," *European Journal of Business and Management*, vol. 9, no. 11, pp. 116–122, 2017.

[20] I. Naim, M. I. Tanveer, D. Gildea, and M. E. Hoque, "Automated analysis and prediction of job interview performance," *IEEE Transactions on Affective Computing*, vol. 9, no. 2, pp. 191–204, 2016.

[21] E. Y. Nakagawa, P. Oliveira Antonino, and M. Becker, "Reference architecture and product line architecture: A subtle but critical difference," in *European conference on software architecture*. Springer, 2011, pp. 207–211.

[22] S. Angelov, P. Grefen, and D. Greefhorst, "A classification of software reference architectures: Analyzing their success and effectiveness," in *2009 Joint Working IEEE/IFIP Conference on Software Architecture & European Conference on Software Architecture*. IEEE, 2009, pp. 141–150.

[23] L. Bass, P. Clements, and R. Kazman, *Software architecture in practice*. Addison-Wesley Professional, 2003.

[24] S. Martinez-Fernandez, P. S. M. Dos Santos, C. P. Ayala, X. Franch, and G. H. Travassos, "Aggregating empirical evidence about the benefits and drawbacks of software reference architectures," in *2015 ACM/IEEE international symposium on empirical software engineering and measurement (ESEM)*. IEEE, 2015, pp. 1–10.

[25] P. Clements, D. Garlan, R. Little, R. Nord, and J. Stafford, "Documenting software architectures: views and beyond," in *25th International Conference on Software Engineering, 2003. Proceedings.* IEEE, 2003, pp. 740–741.

[26] S. Behere and M. Törngren, "A functional reference architecture for autonomous driving," *Information and Software Technology*, vol. 73, pp. 136–150, 2016.

[27] A. Di Prospero, N. Norouzi, M. Fokaefs, and M. Litoiu, "Chatbots as assistants: an architectural framework," in *Proceedings of the 27th Annual International Conference on Computer Science and Software Engineering*, 2017, pp. 76–86.

[28] M. Osorio, C. Zepeda, and J. Carballido, "Myubot: Towards an artificial intelligence agent system chat-bot for well-being and mental health. accepted to appear in proceedings artificial intelligence for health," in *PersonaLized MedIcine aNd WEllbeing Workshop at ECAI 2020*, 2020.

[29] M. Fuchs, P. Hejda, and P. Slavỳk, "Architecture of multi-modal dialogue system," in *International Workshop on Text, Speech and Dialogue*. Springer, 2000, pp. 433–438.

[30] A. Abid, A. Abbas, A. Khelifi, M. S. Farooq, R. Iqbal, and U. Farooq, "An architectural framework for information integration using machine learning approaches for smart city security profiling," *International Journal of Distributed Sensor Networks*, vol. 16, no. 10, p. 1550147720965473, 2020.

[31] C. Verdouw, R. M. Robbemond, T. Verwaart, J. Wolfert, and A. J. Beulens, "A reference architecture for iot-based logistic information systems in agri-food supply chains," *Enterprise information systems*, vol. 12, no. 7, pp. 755–779, 2018.

[32] E. M. Grua, M. De Sanctis, and P. Lago, "A reference architecture for personalized and self-adaptive e-health apps," in *European Conference on Software Architecture*. Springer, 2020, pp. 195–209.

[33] A. Elgammal and B. J. Krämer, "A reference architecture for smart digital platform for personalized prevention and patient management." in *Next-Gen Digital Services*, 2021, pp. 88–99.

[34] C. Luo and C. W. Chan, "An architectural framework for developing intelligent systems for the co 2 capture process," in *CCECE 2010*. IEEE, 2010, pp. 1–4.

[35] S. Soni, P. Kumar, and A. Saha, "Automatic question generation: A systematic review," in *International Conference on Advances in Engineering Science Management & Technology (ICAESMT)-2019, Uttaranchal University, Dehradun, India*, 2019.

[36] D. R. Ch and S. K. Saha, "Automatic multiple choice question generation from text: A survey," *IEEE Transactions on Learning Technologies*, 2018.

[37] L. Pan, W. Lei, T.-S. Chua, and M.-Y. Kan, "Recent advances in neural question generation," *arXiv preprint arXiv:1905.08949*, 2019.

[38] G. Kurdi, J. Leo, B. Parsia, U. Sattler, and S. Al-Emari, "A systematic review of automatic question generation for educational purposes," *International Journal of Artificial Intelligence in Education*, vol. 30, no. 1, pp. 121–204, 2020.

[39] P. Y. Niranjan, V. S. Rajpurohit, and R. Malgi, "A survey on chat-bot system for agriculture domain," in *2019 1st International Conference on Advances in Information Technology (ICAIT)*. IEEE, 2019, pp. 99–103.

[40] M. Nuruzzaman and O. K. Hussain, "A survey on chatbot implementation in customer service industry through deep neural networks," in *2018 IEEE 15th International Conference on e-Business Engineering (ICEBE)*. IEEE, 2018, pp. 54–61.

[41] B. Xu and Z. Zhuang, "Survey on psychotherapy chatbots," *Concurrency and Computation: Practice and Experience*, p. e6170, 2020.

[42] B. Borah, D. Pathak, P. Sarmah, B. Som, and S. Nandi, "Survey of textbased chatbot in perspective of recent technologies," in *International Conference on Computational Intelligence, Communications, and Business Analytics*. Springer, 2018, pp. 84–96.

[43] E. H. Almansor and F. K. Hussain, "Survey on intelligent chatbots: State-of-the-art and future research directions," in *Conference on Complex, Intelligent, and Software Intensive Systems*. Springer, 2019, pp. 534–543.

[44] A. Pina, E. Cerezo, and F. J. Serón, "Computer animation: from avatars to unrestricted autonomous actors (a survey on replication and modelling mechanisms)," *Computers & Graphics*, vol. 24, no. 2, pp. 297–311, 2000.

[45] J. Amidei, P. Piwek, and A. Willis, "Evaluation methodologies in automatic question generation," pp. 307—-317, 2018.

[46] A. Gatt and E. Krahmer, "Survey of the state of the art in natural language generation: Core tasks, applications and evaluation," *Journal of Artificial Intelligence Research*, vol. 61, pp. 65–170, 2018.

[47] P. Nema and M. M. Khapra, "Towards a better metric for evaluating question generation systems," *arXiv preprint arXiv:1808.10192*, 2018.

[48] D. Gkatzia and S. Mahamood, "A snapshot of nlg evaluation practices 2005-2014," in *Proceedings of the 15th European Workshop on Natural Language Generation (ENLG)*, 2015, pp. 57–60.

[49] M. Amilon, "Chatbot with common-sense database," 2015.

[50] V. Hung, M. Elvir, A. Gonzalez, and R. DeMara, "Towards a method for evaluating naturalness in conversational dialog systems," in *2009 IEEE international conference on systems, man and cybernetics*. IEEE, 2009, pp. 1236–1241.

[51] W. Maroengsit, T. Piyakulpinyo, K. Phonyiam, S. Pongnumkul, P. Chaovalit, and T. Theeramunkong, "A survey on evaluation methods for chatbots," in *Proceedings of the 2019 7th International Conference on Information and Education Technology*, 2019, pp. 111–119.

[52] A. Atiyah, S. Jusoh, and F. Alghanim, "Evaluation of the naturalness of chatbot applications," in *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*. IEEE, 2019, pp. 359–365.

[53] M. Lundell Vinkler and P. Yu, "Conversational chatbots with memory-based question and answer generation," 2020. [Online]. Available: https://devskiller.com/65-recruitment-stats-hr-pros-must-know-2018/

[54] B. A. Shawar and E. Atwell, "Different measurement metrics to evaluate a chatbot system," in *Proceedings of the workshop on bridging the gap: Academic and industrial research in dialog technologies*, 2007, pp. 89–96.

[55] Y. Cao, "Testbot: A chatbot-based interactive interview preparation application," 2020.

[56] A. Venkatesh, C. Khatri, A. Ram, F. Guo, R. Gabriel, A. Nagar, R. Prasad, M. Cheng, B. Hedayatnia, A. Metallinou *et al.*, "On evaluating and comparing conversational agents," *arXiv preprint arXiv:1801.03625*, vol. 4, pp. 60–68, 2018.

[57] D. Lala, K. Inoue, and T. Kawahara, "Evaluation of real-time deep learning turn-taking models for multiple dialogue scenarios," in *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, 2018, pp. 78–86.

[58] D. Jones, *Evidence-based Software Engineering based on the publicly available data.* Knowledge Software, Ltd, 2020.

[59] I. Manotas, C. Bird, R. Zhang, D. Shepherd, C. Jaspan, C. Sadowski, L. Pollock, and J. Clause, "An empirical study of practitioners' perspectives on green software engineering," in *2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE).* IEEE, 2016, pp. 237–248.

[60] M. V. Kosti, R. Feldt, and L. Angelis, "Personality, emotional intelligence and work preferences in software engineering: An empirical study," *Information and Software Technology*, vol. 56, no. 8, pp. 973–990, 2014.

[61] D. Ford, T. Barik, L. Rand-Pickett, and C. Parnin, "The tech-talk balance: what technical interviewers expect from technical candidates," in *2017 IEEE/ACM 10th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE).* IEEE, 2017, pp. 43–48.

[62] M. Behroozi, A. Lui, I. Moore, D. Ford, and C. Parnin, "Dazed: measuring the cognitive load of solving technical interview problems at the whiteboard," in *Proceedings of the 40th International Conference on Software Engineering: New Ideas and Emerging Results*, 2018, pp. 93–96.

[63] M. Behroozi, C. Parnin, and T. Barik, "Hiring is broken: What do developers say about technical interviews?" in *2019 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC).* IEEE, 2019, pp. 1–9.

[64] C. Rosen and E. Shihab, "What are mobile developers asking about? a large scale study using stack overflow," *Empirical Software Engineering*, vol. 21, no. 3, pp. 1192–1223, 2016.

[65] A. Hindle, M. W. Godfrey, and R. C. Holt, "What's hot and what's not: Windowed developer topic analysis," in *2009 IEEE international conference on software maintenance*. IEEE, 2009, pp. 339–348.

[66] S. Keele *et al.*, "Guidelines for performing systematic literature reviews in software engineering," Technical report, Ver. 2.3 EBSE Technical Report. EBSE, Tech. Rep., 2007.

[67] C. Wohlin, "Guidelines for snowballing in systematic literature studies and a replication in software engineering," in *Proceedings of the 18th international conference on evaluation and assessment in software engineering*, 2014, pp. 1–10.

[68] D. R. Thomas, "A general inductive approach for analyzing qualitative evaluation data," *American journal of evaluation*, vol. 27, no. 2, pp. 237–246, 2006.

[69] P. B. Kruchten, "The 4+ 1 view model of architecture," *IEEE software*, vol. 12, no. 6, pp. 42–50, 1995.

[70] N. Rozanski and E. Woods, *Software systems architecture: working with stakeholders using viewpoints and perspectives*. Addison-Wesley, 2012.

[71] L. Chung and J. C. S. do Prado Leite, "On non-functional requirements in software engineering," in *Conceptual modeling: Foundations and applications*. Springer, 2009, pp. 363–379.

[72] K. Inoue, K. Hara, D. Lala, K. Yamamoto, S. Nakamura, K. Takanashi, and T. Kawahara, "Job interviewer android with elaborate follow-up question generation," in *Proceedings of the 2020 International Conference on Multimodal Interaction*, 2020, pp. 324–332.

[73] T. Schneeberger, A. Hirsch, C. König, and P. Gebhard, "Impact of virtual environment design on the assessment of virtual agents," in *Proceedings of the 19th ACM International Conference on Intelligent Virtual Agents*, 2019, pp. 148–150.

[74] M. Langer, C. J. König, P. Gebhard, and E. André, "Dear computer, teach me manners: Testing virtual employment interview training," *International Journal of Selection and Assessment*, vol. 24, no. 4, pp. 312–323, 2016.

[75] V. Salvi, A. Vasanwalla, N. Aute, and A. Joshi, "Virtual simulation of technical interviews," in *2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA)*. IEEE, 2017, pp. 1–6.

[76] M. Behroozi and C. Parnin, "Can we predict stressful technical interview settings through eye-tracking?" in *Proceedings of the Workshop on Eye Movements in Programming*, 2018, pp. 1–5.

[77] C. Qin, H. Zhu, C. Zhu, T. Xu, F. Zhuang, C. Ma, J. Zhang, and H. Xiong, "Duerquiz: A personalized question recommender system for intelligent job interview," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 2165–2173.

[78] B. Shi, S. Li, J. Yang, M. E. Kazdagli, and Q. He, "Learning to ask screening questions for job postings," *arXiv preprint arXiv:2004.14969*, 2020.

[79] M.-H. Su, C.-H. Wu, K.-Y. Huang, Q.-B. Hong, and H.-H. Huang, "Follow-up question generation using pattern-based seq2seq with a small corpus for interview coaching." in *INTERSPEECH*, 2018, pp. 1006–1010.

[80] M. X. Zhou, C. Wang, G. Mark, H. Yang, and K. Xu, "Building real-world chatbot interviewers: Lessons from a wizard-of-oz field study." in *IUI Workshops*, 2019.

[81] M. Laiq and O. Dieste, "Chatbot-based interview simulator: A feasible approach to train novice requirements engineers," in *2020 10th International Workshop on Requirements Engineering Education and Training (REET)*. IEEE, 2020, pp. 1–8.

[82] A.-I. Carțiș and D. M. Suciu, "Chatbots as a job candidate evaluation tool," in *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*. Springer, 2019, pp. 189–193.

[83] K. Cofino, V. Ramanarayanan, P. Lange, D. Pautler, D. Suendermann-Oeft, and K. Evanini, "A modular, multimodal open-source virtual interviewer dialog agent," in *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, 2017, pp. 520–521.

[84] H. Kumazaki, Z. Warren, T. Muramatsu, Y. Yoshikawa, Y. Matsumoto, M. Miyao, M. Nakano, S. Mizushima, Y. Wakita, H. Ishiguro *et al.*, "A pilot study for robot appearance preferences among high-functioning individuals with autism spectrum disorder: Implications for therapeutic use," *PloS one*, vol. 12, no. 10, p. e0186581, 2017.

[85] C. Maddumage, D. Senevirathne, I. Gayashan, T. Shehan, and S. Sumathipala, "Intelligent recruitment system," in *2019 IEEE 5th International Conference for Convergence in Technology (I2CT)*. IEEE, 2019, pp. 1–6.

[86] M. Fowler, *Patterns of Enterprise Application Architecture: Pattern Enterpr Applica Arch*. Addison-Wesley, 2012.

[87] M. Hoque, M. Courgeon, J.-C. Martin, B. Mutlu, and R. W. Picard, "Mach: My automated conversation coach," in *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, 2013, pp. 697–706.

[88] A. Balayn, C. Lofi, and G.-J. Houben, "Managing bias and unfairness in data for decision support: a survey of machine learning and data engineering approaches to identify and mitigate bias and unfairness within data management and analytics systems," *The VLDB Journal*, pp. 1–30, 2021.

[89] E. Ntoutsi, P. Fafalios, U. Gadiraju, V. Iosifidis, W. Nejdl, M.-E. Vidal, S. Ruggieri, F. Turini, S. Papadopoulos, E. Krasanakis *et al.*, "Bias in data-driven artificial intelligence systems—an introductory survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 10, no. 3, p. e1356, 2020.

[90] T. Weitzel, A. Eckhardt, and S. Laumer, "A framework for recruiting it talent: Lessons from siemens." *MIS Quarterly Executive*, vol. 8, no. 4, 2009.

[91] S. Amershi, D. Weld, M. Vorvoreanu, A. Fourney, B. Nushi, P. Collisson, J. Suh, S. Iqbal, P. N. Bennett, K. Inkpen *et al.*, "Guidelines for human-ai interaction," in *Proceedings of the 2019 chi conference on human factors in computing systems*, 2019, pp. 1–13.

[92] J. Levashina, C. J. Hartwell, F. P. Morgeson, and M. A. Campion, "The structured employment interview: Narrative and quantitative review of the research literature," *Personnel Psychology*, vol. 67, no. 1, pp. 241–293, 2014.

[93] C. Fernández and A. Fernández, "Ethical and legal implications of ai recruiting software," *Ercim News*, vol. 116, pp. 22–23, 2019.

[94] T. Baur, I. Damian, P. Gebhard, K. Porayska-Pomsta, and E. André, "A job interview simulation: Social cue-based interaction with a virtual character," in *2013 International Conference on Social Computing*. IEEE, 2013, pp. 220–227.

[95] S. Shimizu, N. Jincho, and H. Kikuchi, "Influence of interactive questions on the sense of presence and anxiety in a virtual-reality job-interview simulation," in *Proceedings of the 2019 3rd International Conference on Virtual and Augmented Reality Simulations*, 2019, pp. 1–5.

[96] I. Stanica, M.-I. Dascalu, C. N. Bodea, and A. D. B. Moldoveanu, "VR job interview simulator: where virtual reality meets artificial intelligence for education," in *2018 Zooming innovation in consumer technologies conference (ZINC)*. IEEE, 2018, pp. 9–12.

[97] V. Rus, Z. Cai, and A. Graesser, "Question generation: Example of a multi-year evaluation campaign," *Proc WS on the QGSTEC*, 2008.

[98] M. Al-Yahya, "Ontology-based multiple choice question generation," *The Scientific World Journal*, vol. 2014, 2014.

[99] R. Mitkov *et al.*, "Computer-aided generation of multiple-choice tests," in *Proceedings of the HLT-NAACL 03 workshop on Building educational applications using natural language processing*, 2003, pp. 17–22.

[100] M. Heilman and N. A. Smith, "Question generation via overgenerating transformations and ranking," Carnegie-Mellon Univ Pittsburgh pa language technologies insT, Tech. Rep., 2009.

[101] J. D. Williams, "Web-style ranking and slu combination for dialog state tracking," in *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, 2014, pp. 282–291.

[102] H. Chen, X. Liu, D. Yin, and J. Tang, "A survey on dialogue systems: Recent advances and new frontiers," *Acm Sigkdd Explorations Newsletter*, vol. 19, no. 2, pp. 25–35, 2017.

[103] X. Du, J. Shao, and C. Cardie, "Learning to ask: Neural question generation for reading comprehension," *arXiv preprint arXiv:1705.00106*, 2017.

[104] Y. Chali and T. Baghaee, "Automatic opinion question generation," in *Proceedings of the 11th International Conference on Natural Language Generation*, 2018, pp. 152–158.

[105] Y. Mandasari, "Follow-up question generation," Master's thesis, University of Twente, 2019.

[106] M.-H. Su, C.-H. Wu, and Y. Chang, "Follow-up question generation using neural tensor network-based domain ontology population in an interview coaching system." in *INTERSPEECH*, 2019, pp. 4185–4189.

[107] J. Purohit, A. Bagwe, R. Mehta, O. Mangaonkar, and E. George, "Natural language processing based jaro-the interviewing chatbot," in *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*. IEEE, 2019, pp. 134–136.

[108] S. Gao, X. Chen, Z. Ren, D. Zhao, and R. Yan, "Meaningful answer generation of e-commerce question-answering," *ACM Transactions on Information Systems (TOIS)*, vol. 39, no. 2, pp. 1–26, 2021.

[109] A. Agarwal, N. Sachdeva, R. K. Yadav, V. Udandarao, V. Mittal, A. Gupta, and A. Mathur, "Eduqa: Educational domain question answering system using conceptual network mapping," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 8137–8141.

[110] N. Nawaz and A. M. Gomes, "Artificial intelligence chatbots are new recruiters," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 10, no. 9, 2019.

[111] M. Sarosa, M. Junus, M. U. Hoesny, Z. Sari, and M. Fatnuriyah, "Classification technique of interviewer-bot result using naive bayes and phrase reinforcement algorithms," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 13, no. 02, pp. 33–47, 2018.

[112] M. Junus, M. Sarosa, M. Fatnuriyah, M. U. Hoesny, and Z. Sari, "Interviewer bot design to help student learning english for job interview," *IC-ITECHS*, vol. 1, pp. 45–50, 2014.

[113] H. Hamdi, P. Richard, A. Suteau, and M. Saleh, "A multi-modal virtual environment to train for job interview." in *PECCS*, 2011, pp. 551–556.

[114] M. Rehm, "Developing enculturated agents: Pitfalls and strategies," in *Handbook of research on culturally-aware information technology: Perspectives and models*. IGI Global, 2011, pp. 362–386.

[115] P. Gebhard, T. Baur, I. Damian, G. Mehlmann, J. Wagner, and E. André, "Exploring interaction strategies for virtual characters to induce stress in simulated job interviews," in *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, 2014, pp. 661–668.

[116] J. Wagner, F. Lingenfelser, T. Baur, I. Damian, F. Kistler, and E. André, "The social signal interpretation (ssi) framework: multimodal signal processing and recognition in real-time," in *Proceedings of the 21st ACM international conference on Multimedia*, 2013, pp. 831–834.

[117] T. Baur, I. Damian, F. Lingenfelser, J. Wagner, and E. André, "Nova: Automated analysis of nonverbal signals in social interactions," in *International Workshop on Human Behavior Understanding*. Springer, 2013, pp. 160–171.

[118] H. Andrews, M. Bramar, L. Lupinski, and B. Kapralos, "A serious game for interview preparation," in *2014 IEEE Games Media Entertainment*. IEEE, 2014, pp. 1–2.

[119] A. Heloir and M. Kipp, "Real-time animation of interactive agents: Specification and realization," *Applied Artificial Intelligence*, vol. 24, no. 6, pp. 510–529, 2010.

[120] M. Courgeon, S. Buisine, and J.-C. Martin, "Impact of expressive wrinkles on perception of a virtual character's facial expressions of emotions," in *International Workshop on Intelligent Virtual Agents*. Springer, 2009, pp. 201–214.

[121] B. Froba and A. Ernst, "Face detection with the modified census transform," in *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.* IEEE, 2004, pp. 91–96.

[122] S. Kawato and J. Ohya, "Real-time detection of nodding and head-shaking by directly detecting and tracking the" between-eyes"," in *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*. IEEE, 2000, pp. 40–45.

[123] H. Kumazaki, T. Muramatsu, Y. Yoshikawa, B. A. Corbett, Y. Matsumoto, H. Higashida, T. Yuhi, H. Ishiguro, M. Mimura, and M. Kikuchi, "Job interview training targeting nonverbal communication using an android robot for individuals with autism spectrum disorder," *Autism*, vol. 23, no. 6, pp. 1586–1595, 2019, pMID: 30795694. [Online]. Available: https://doi.org/10.1177/1362361319827134

[124] D. F. Glas, T. Minato, C. T. Ishi, T. Kawahara, and H. Ishiguro, "Erica: The erato intelligent conversational android," in *2016 25th IEEE International symposium on robot and human interactive communication (RO-MAN)*. IEEE, 2016, pp. 22–29.

[125] D. Villani, C. Rotasperti, P. Cipresso, S. Triberti, C. Carissoli, and G. Riva, "Assessing the emotional state of job applicants through a virtual reality simulation: A psychophysiological study," in *eHealth 360*. Springer, 2017, pp. 119–126.

[126] A. W. Romadon, K. M. Lhaksmana, I. Kurniawan, and D. Richasdy, "Analyzing tf-idf and word embedding for implementing automation in job interview grading," in *2020 8th International Conference on Information and Communication Technology (ICoICT)*. IEEE, 2020, pp. 1–4.

[127] H. Joshi, J. Pareek, R. Patel, and K. Chauhan, "To stop or not to stop—experiments on stopword elimination for information retrieval of gujarati text documents," in *2012 Nirma university international conference on engineering (NUiCONE)*. IEEE, 2012, pp. 1–4.

[128] C.-M. Tan, Y.-F. Wang, and C.-D. Lee, "The use of bigrams to enhance text categorization," *Information processing & management*, vol. 38, no. 4, pp. 529–546, 2002.

[129] A. K. McCallum, "A Machine Learning for Language Toolkit," http://mallet.cs.umass.edu, 2002, [Online; accessed 19-Sept-2021].

[130] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Transactions on pattern analysis and machine intelligence*, no. 6, pp. 721–741, 1984.

[131] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[132] M. Bagherzadeh and R. Khatchadourian, "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*, 2019, pp. 432–442.

[133] A. Barua, S. W. Thomas, and A. E. Hassan, "What are developers talking about? an analysis of topics and trends in stack overflow," *Empirical Software Engineering*, vol. 19, no. 3, pp. 619–654, 2014.

[134] A. Panichella, B. Dit, R. Oliveto, M. Di Penta, D. Poshynanyk, and A. De Lucia, "How to effectively use topic models for software engineering tasks? an approach based on genetic algorithms," in *2013 35th International Conference on Software Engineering (ICSE)*. IEEE, 2013, pp. 522–531.

[135] H. Wallach, D. Mimno, and A. McCallum, "Rethinking lda: Why priors matter," *Advances in neural information processing systems*, vol. 22, pp. 1973–1981, 2009.

[136] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *Proceedings of the National academy of Sciences*, vol. 101, no. suppl 1, pp. 5228–5235, 2004.

[137] M. Röder, A. Both, and A. Hinneburg, "Exploring the space of topic coherence measures," in *Proceedings of the eighth ACM international conference on Web search and data mining*, 2015, pp. 399–408.

[138] X.-L. Yang, D. Lo, X. Xia, Z.-Y. Wan, and J.-L. Sun, "What security questions do developers ask? a large-scale study of stack overflow posts," *Journal of Computer Science and Technology*, vol. 31, no. 5, pp. 910–924, 2016.

[139] K. Bajaj, K. Pattabiraman, and A. Mesbah, "Mining questions asked by web developers," in *Proceedings of the 11th Working Conference on Mining Software Repositories*, 2014, pp. 112–121.

[140] S. Nadi, S. Krüger, M. Mezini, and E. Bodden, "Jumping through hoops: Why do java developers struggle with cryptography apis?" in *Proceedings of the 38th International Conference on Software Engineering*, 2016, pp. 935–946.

[141] G. Pinto, W. Torres, and F. Castor, "A study on the most popular questions about concurrent programming," in *Proceedings of the 6th Workshop on Evaluation and Usability of Programming Languages and Tools*, 2015, pp. 39–46.

[142] C. Treude, O. Barzilay, and M.-A. Storey, "How do programmers ask and answer questions on the web?(nier track)," in *Proceedings of the 33rd international conference on software engineering*, 2011, pp. 804–807.

[143] S. Ahmed and M. Bagherzadeh, "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*, 2018, pp. 1–10.

[144] J. W. Kruize, J. Wolfert, H. Scholten, C. Verdouw, A. Kassahun, and A. J. Beulens, "A reference architecture for farm software ecosystems," *Computers and Electronics in Agriculture*, vol. 125, pp. 12–28, 2016.