|  |  |
| --- | --- |
| School of Electronic Engineering and Computer Science | **Interim**  **Programme of study:**  Digital & Technology Solutions (Software Engineering)  **Project Title:**  **AI Based Automated Resume Screening**  **Supervisor:**  Keshav Bhandari  **Student Name:**  Kenzo Dubreuil  Date: 28th November 2024 |
| Final Year  Undergraduate Project 2024/25 |
| QMLogo |

Abstract

This project explores the development of an intelligent system for automated resume screening, with a focus on aligning job descriptions to relevant candidate profiles. Unlike traditional machine learning approaches, which depend heavily on large volumes of labeled data to train supervised models, this work tackles the problem from a fundamentally different angle. In real-world hiring contexts, such annotated datasets — where resumes are explicitly labeled for relevance to job roles — are rarely available. This lack of labels poses a major challenge for building effective and scalable solutions using standard supervised learning techniques.

To address this gap, the system is designed to learn without the need for labeled training data. Instead, it leverages modern language models and representation learning techniques to simulate the ranking process and measure the relevance between resumes and job descriptions. The result is a system capable of prioritizing candidates based on their suitability to a given role, even in the absence of human-annotated examples. The model is evaluated both quantitatively, through benchmark comparisons, and qualitatively, through human assessments, to ensure that it aligns with real-world hiring expectations. This approach offers a more practical, generalizable alternative to existing resume ranking systems that rely on scarce and often biased labeled datasets.

**C**

**ontents**

[Chapter 1: Introduction 12](#_Toc197978504)

[1.1 Background 12](#_Toc197978505)

[1.2 Problem Statement 13](#_Toc197978506)

[1.3 Aim 14](#_Toc197978507)

[1.4 Objectives 14](#_Toc197978508)

[1.5 Significance of the Project 15](#_Toc197978509)

[1.6 Scope and Deliverables 16](#_Toc197978510)

[1.7 Report Structure 16](#_Toc197978511)

[Chapter 2: Literature Review 18](#_Toc197978512)

[2.1 Overview 18](#_Toc197978513)

[2.2 Existing Tools in Automated Resume Screening 18](#_Toc197978514)

[2.2.1 Applicant Tracking Systems (ATS) 19](#_Toc197978515)

[2.2.2 AI-Powered Recruitment Tools 19](#_Toc197978516)

[2.2.3 NLP-Based Resume Screening Tools 20](#_Toc197978517)

[2.2.4 Hybrid Systems 21](#_Toc197978518)

[2.3 Analysis of Existing Tools 22](#_Toc197978519)

[2.4 Conclusion 23](#_Toc197978520)

[Chapter 3: Research Methodology and Primary Research 25](#_Toc197978521)

[3.1 Overview 25](#_Toc197978522)

[3.2 Research Design 25](#_Toc197978523)

[3.3 Data Collection 25](#_Toc197978524)

[3.3.1 Survey Design 25](#_Toc197978525)

[3.3.2 Recruitment of Participants 26](#_Toc197978526)

[3.3.3 Ethical Considerations 26](#_Toc197978527)

[3.4 Survey Administration 26](#_Toc197978528)

[3.5 Data Analysis 26](#_Toc197978529)

[3.6 Results & Key Findings 26](#_Toc197978530)

[3.6.1 Recruitment Challenges 27](#_Toc197978531)

[3.6.2 Expectations from Automation 28](#_Toc197978532)

[3.6.3 Perceptions on Bias 29](#_Toc197978533)

[3.6.4 Feedback for System Design 29](#_Toc197978534)

[3.6.5 Key Themes Identified 30](#_Toc197978535)

[3.7 Risk Assessment 30](#_Toc197978536)

[3.7.1 Ethical Risks 30](#_Toc197978537)

[3.7.2 Technical Risks 31](#_Toc197978538)

[3.7.3 Operational Risks 31](#_Toc197978539)

[3.7.4 Research-Related Risks 32](#_Toc197978540)

[3.7.5 Conclusion 32](#_Toc197978541)

[3.8 Conclusion 32](#_Toc197978542)

[3.9 Appendices 33](#_Toc197978543)

[Chapter 4: Design 34](#_Toc197978544)

[4.1 Overview 34](#_Toc197978545)

[4.2 System Architecture 34](#_Toc197978546)

[4.3 Data Flow and Processing 35](#_Toc197978547)

[4.4 Design Considerations 35](#_Toc197978548)

[4.5 Conclusion 36](#_Toc197978549)

[Chapter 5: Implementation 37](#_Toc197978550)

[5.1 Overview 37](#_Toc197978551)

[5.2 Dataset Description 37](#_Toc197978552)

[5.3 Dual Encoder Architecture 38](#_Toc197978553)

[5.4 Contrastive Learning with MarginMSE Loss 39](#_Toc197978554)

[5.5 Model Pipeline 39](#_Toc197978555)

[5.5.1 Training Pipeline 40](#_Toc197978556)

[5.5.2 Inference Pipeline 41](#_Toc197978557)

[5.6 Visual Overview 42](#_Toc197978558)

[5.6.1 Training Pipeline 42](#_Toc197978559)

[5.6.2 Inference Pipeline 43](#_Toc197978560)

[5.7 Tools and Libraries Used 43](#_Toc197978561)

[Chapter 6: Evaluation 45](#_Toc197978562)

[6.1 Objective Metric Evaluation 45](#_Toc197978563)

[6.2 Baseline Comparison 46](#_Toc197978564)

[6.3 Human Evaluation 46](#_Toc197978565)

[6.4 Conclusion 47](#_Toc197978566)

[Chapter 7: Discussion and Future Work 48](#_Toc197978567)

[7.1 Key Findings 48](#_Toc197978568)

[7.2 Limitations 48](#_Toc197978569)

[7.3 Future Work 49](#_Toc197978570)

[Chapter 8: Conclusion 50](#_Toc197978571)

[References 52](#_Toc197978572)

[Appendix A: Survey Questionnaire 53](#_Toc197978573)

[Appendix B: Email Invitation to Participants 55](#_Toc197978574)

[Appendix C: Participant Consent Form 56](#_Toc197978575)

[Appendix D: Participation Informatio Sheet 58](#_Toc197978576)

[Appendix E: Ethics Approval Letter 61](#_Toc197978577)

# Introduction

## 1.1 Background

Recruitment is one of the most crucial functions within any organization, directly influencing its ability to attract, assess, and retain talent. At the heart of the hiring process lies resume screening — the initial step where recruiters evaluate candidate applications to determine which ones merit further consideration. Despite being foundational to hiring decisions, resume screening remains predominantly manual and highly time-consuming. Recruiters often spend only a few seconds on each resume, which increases the likelihood of missing strong candidates due to superficial evaluations, inconsistent judgments, or cognitive overload.

Although recent years have seen increasing interest in applying artificial intelligence (AI) to recruitment — particularly in areas such as candidate sourcing, chatbots, and automated interview assessments — the domain of resume screening has not kept pace with innovation. The most common solutions currently on the market rely heavily on keyword matching or predefined heuristics, which may filter out qualified candidates who don’t use conventional phrasing or formatting. Furthermore, many traditional machine learning approaches used in recruitment depend on large, labeled datasets. However, in real-world hiring settings, such labels — like "good hire" or "qualified match" — are rarely available at the scale required to train supervised models.

This project addresses this gap with a fresh, technically innovative approach: the development of a machine learning model for resume ranking that does not require human-labeled training data. Instead of relying on supervised classification or keyword-based filtering, this system uses contrastive learning and semantic embeddings to rank resumes by their relevance to a job description. The model is trained using automatically generated pseudo-queries and hard-negative mining, allowing it to learn nuanced distinctions between relevant and irrelevant resumes in a self-supervised manner. This leverages the power of modern language models such as GPT and transformer-based encoders to understand the meaning and context of both job descriptions and resumes — a capability that keyword-based systems fundamentally lack.

This unsupervised or weakly supervised approach is particularly novel because it mirrors the subjective and context-dependent nature of hiring decisions while offering scalability and flexibility across roles and industries. It also opens the door to evaluating resumes in a more context-aware and holistic manner, accounting for implicit qualifications, transferable skills, and soft competencies embedded in natural language.

While the system is not primarily designed to mitigate hiring bias, its use of objective language understanding and ranking — as opposed to manual heuristics or opaque black-box scoring — may help reduce certain sources of inconsistency and unfairness that arise in manual screening. However, the core contribution of this project lies in its methodological innovation: demonstrating that effective resume ranking can be achieved using recent advances in representation learning and natural language understanding, even in the absence of labeled training data.

By applying this approach to real-world resume and job description data, this project not only delivers a functional prototype but also contributes to the growing field of AI for recruitment — showcasing how modern machine learning techniques can solve practical, high-stakes problems that don’t conform to the assumptions of traditional supervised learning paradigms.

## 1.2 Problem Statement

The recruitment process, particularly the initial screening of resumes, remains one of the most time-consuming and inefficient stages of hiring. Recruiters are often faced with hundreds of applications for a single role, and despite the high stakes involved, the tools and methods used to perform resume screening have changed very little in recent decades. Manual evaluation remains the norm, leading to a number of persistent challenges.

1. **Scalability and Efficiency**: Recruiters typically spend only a few seconds reviewing each resume (LinkedIn, 2018), which limits their ability to assess candidates holistically. With high application volumes becoming the norm, manual screening is no longer scalable.
2. **Lack of Contextual Matching**: Most existing automated screening tools rely on keyword matching or rigid heuristics. These methods often fail to capture nuanced relationships between a job description and a candidate’s qualifications, experiences, or transferable skills. This results in the exclusion of potentially strong candidates who do not use conventional phrasing or formats.
3. **Dependence on Labeled Data**: Traditional machine learning methods require large volumes of labeled training data to make effective predictions — data which is rarely available in recruitment contexts. Labels such as “suitable” or “hired” are subjective, inconsistently applied, and often not recorded at all. This limits the applicability of supervised models.
4. **Limited Adaptability**: Many existing solutions are built around predefined roles, industries, or taxonomies, making them difficult to generalize across different job categories or organizational needs. In contrast, the variability of real-world job descriptions and resumes calls for a more flexible, adaptive approach.
5. **Redundancy of Existing Tools**: While AI has been used for resume parsing or applicant tracking, relatively few systems attempt true semantic understanding of resumes and job descriptions. Current tools lack the capacity to go beyond surface-level matches and assess the deeper relevance between an applicant’s background and a given role.

This project seeks to address these limitations by introducing a fundamentally different approach to resume screening. Instead of relying on labeled datasets or rigid rule-based systems, it leverages recent advances in contrastive learning and large language models to build a scalable, self-supervised system. This system ranks resumes based on semantic relevance to job descriptions without requiring any manual annotations. It introduces a flexible and generalizable architecture that can work across domains, enabling a more context-aware, efficient, and intelligent approach to resume screening.

While improving fairness is a potential benefit, the core innovation of this work lies in demonstrating that high-quality resume ranking can be achieved using machine learning techniques that do not rely on traditional supervised learning assumptions. This shift in paradigm is central to the project's contribution.

## 1.3 Aim

The aim of this project is to develop an automated resume screening system that leverages machine learning and natural language processing (NLP) to intelligently rank candidates based on their semantic fit to a given job description. Unlike traditional approaches that rely on manually labeled data or surface-level keyword matching, this system applies a contrastive learning framework and transformer-based language models to learn meaningful representations of resumes and job postings.

The model is trained to differentiate between relevant and irrelevant resume-job pairs using unlabeled data, allowing it to generalize across roles and industries without human supervision. The final system generates ranked candidate lists based on contextual understanding rather than rigid rule-based filtering. This approach aims to enhance the speed, scalability, and precision of resume screening, providing recruiters with a powerful tool to identify high-potential candidates more effectively.

While reducing bias may be a secondary benefit, the primary innovation lies in demonstrating a label-free, generalizable architecture for resume ranking — a method that addresses the limitations of both manual screening and conventional supervised ML approaches in real-world hiring scenarios.

## 1.4 Objectives

To fulfill the project’s aim, the following objectives were defined:

1. **Design a Label-Free Training Framework**: Develop a machine learning pipeline that does not rely on manually labeled data, instead using contrastive learning and heuristic sampling to simulate supervision. This enables scalable and domain-agnostic learning from real-world resume corpora.
2. **Leverage Transformer-Based Language Models**: Integrate pre-trained language models such as BERT or Sentence-BERT to extract deep contextual representations of both resumes and job descriptions, enabling nuanced semantic comparisons beyond keyword matching.
3. **Implement a Dual Encoder Architecture**: Train a dual encoder model capable of independently encoding resumes and job descriptions into a shared vector space, allowing efficient similarity computation and large-scale retrieval.
4. **Evaluate Against a Strong Baseline**: Benchmark the system’s performance against a widely used baseline model (all-MiniLM-L6-v2) using both automated metrics (Recall@K, MRR) and human judgment rankings (Kendall Tau, Spearman Rho) across multiple job categories.
5. **Validate Generalization Through Human Feedback**: Collect human rankings of candidate-job relevance and compare them with the model’s output to assess alignment, realism, and practical utility in simulated recruitment tasks.
6. **Highlight Opportunities for Future Research**: Identify future directions such as larger-scale evaluation, fine-grained fairness auditing, and real-time integration with HR systems to transition from experimental prototype to production-ready tool.

## 1.5 Significance of the Project

This project introduces a novel approach to automated resume screening that diverges from conventional supervised learning paradigms. Unlike traditional systems that require extensive labeled data or rely on brittle keyword matching, the proposed method leverages unsupervised and contrastive learning techniques to rank candidate resumes against job descriptions without predefined relevance labels. This makes the approach highly scalable, adaptable across industries, and particularly suited for real-world applications where labeled datasets are scarce or impractical to obtain.

* **For Recruiters:** The system significantly reduces manual screening effort by intelligently surfacing the most relevant candidates based on semantic fit rather than surface-level attributes. It provides a foundation for integrating AI-driven decision support into recruitment pipelines, especially in high-volume application scenarios.
* **For the Field of Machine Learning in HR:** This work contributes to the emerging body of research focused on applying representation learning to textual matching problems in human resources. It demonstrates the feasibility of building effective retrieval models without explicit supervision, highlighting an underexplored path for future development.
* **For Fairness and Efficiency:** While not explicitly built as a fairness tool, the model’s design—focusing on skill and semantic relevance—can help reduce superficial biases inherent in manual review and keyword filters. It shifts the screening process toward deeper content-based evaluation, laying the groundwork for more equitable outcomes.

Overall, this project is significant not only for its practical utility in recruitment but also for its methodological innovation in applying modern NLP and retrieval techniques to a long-standing real-world challenge.

## 1.6 Scope and Deliverables

This project focuses exclusively on the **automation of the initial resume screening phase** through advanced machine learning and natural language processing techniques. It does not include the development of a full user-facing application, but instead emphasizes building and validating the underlying recommendation engine.

The primary deliverables are:

1. **A Machine Learning–Based Ranking System**: A dual-encoder architecture trained using contrastive learning techniques (specifically margin-based triplet loss) to rank resumes based on semantic alignment with job descriptions, without requiring explicit human-labeled relevance data.
2. **Evaluation Framework and Results**: A comprehensive evaluation of the system, using both objective metrics (e.g., Recall@k, MRR, NDCG) and human-in-the-loop validation. This includes comparisons to strong baselines such as all-MiniLM-L6-v2, as well as correlation-based comparisons between human and model rankings.
3. **Reproducible Scripts and Model Artifacts**: All training, inference, and evaluation scripts used in the project, packaged for reproducibility and future experimentation or deployment. This includes datasets, model checkpoints, and evaluation results.

Although limited in scope to resume ranking, the system is designed to be modular and extensible. It can be adapted for broader use in candidate sourcing, profile matching, or integration into end-to-end recruitment platforms with minimal architectural changes.

## 1.7 Report Structure

This final report is organized into eight chapters, each contributing to a comprehensive overview of the project and its development:

* **Chapter 1 – Introduction:** Presents the background, motivation, problem statement, aim, objectives, and significance of the project, setting the foundation for the work undertaken.
* **Chapter 2 – Literature Review:** Reviews existing research and tools in automated resume screening, machine learning in recruitment, and natural language processing, identifying key limitations and opportunities this project addresses.
* **Chapter 3 – Research Methodology and Primary Research:** Describes the methodologies employed, including data acquisition, the rationale behind model selection, and insights from preliminary investigations or user research.
* **Chapter 4 – Design:** Details the conceptual design of the system, including architectural components, system workflows, and functional requirements.
* **Chapter 5 – Implementation:** Explains how the system was developed, highlighting the tools, frameworks, and models used (e.g., contrastive learning, GPT-based query generation), as well as implementation decisions.
* **Chapter 6 – Evaluation:** Presents a thorough evaluation of the system through both quantitative metrics (e.g., MRR, Recall@k) and qualitative human judgment comparison, and discusses its performance relative to a strong baseline.
* **Chapter 7 – Discussion and Future Work:** Interprets the results, reflects on challenges encountered, and outlines possible improvements and extensions, including larger-scale evaluation and bias testing.
* **Chapter 8 – Conclusion:** Summarizes the project's outcomes, contributions, and relevance to the broader context of AI-driven recruitment.

# Literature Review

## Overview

In recent years, Artificial Intelligence (AI) and Natural Language Processing (NLP) have transformed how organizations approach information-heavy tasks, and recruitment is no exception. Resume screening—the critical first step in candidate evaluation—has traditionally been time-consuming, error-prone, and inconsistent. AI-powered solutions have emerged to address these shortcomings by enabling systems to process large volumes of unstructured data, extract relevant information, and assess compatibility between job requirements and candidate profiles.

While a number of commercial and academic tools have been developed to automate aspects of the hiring process, most rely on superficial keyword matching or static rule-based filtering. These approaches often fail to understand the nuanced relationship between a resume and a job description, especially when skills are phrased differently or when candidates demonstrate potential through non-traditional backgrounds.

The central innovation of this project lies in adopting a machine learning approach that does not require manually labeled training data. Traditional supervised learning methods assume the availability of relevance labels (i.e., this resume is a good match for this job), which are rarely present in real-world hiring data. This project overcomes that challenge by framing the problem as a semantic similarity task and leveraging contrastive learning with margin-based loss. By using job categories and synthetic queries to simulate realistic job descriptions, the system learns to rank resumes in a meaningful way without explicit labels.

This chapter provides a review of the literature and tools in the field of automated resume screening, with a particular emphasis on the technological approaches, their limitations, and the emerging techniques that have informed the design of this project.

## Existing Tools in Automated Resume Screening

Automated resume screening has become a focal point in the recruitment technology ecosystem, especially as companies face increasing volumes of applicants for each position. Broadly, the landscape of automated screening tools can be categorized into four key types: Applicant Tracking Systems (ATS), AI-powered recruitment tools, NLP-based screening systems, and hybrid solutions. This section provides an in-depth overview of each, highlighting both their capabilities and limitations in the context of modern hiring needs.

### Applicant Tracking Systems (ATS)

ATS platforms are among the earliest tools developed to aid the recruitment process. Their core function is to collect, store, and manage applicant data while facilitating initial resume filtering based on structured metadata and keyword search criteria.

Strengths:

* **Scalability:** ATS systems such as Taleo, Workday, and Greenhouse can process large volumes of applications efficiently, making them particularly valuable for enterprise-level organizations [9].
* **Integration Capabilities:** These tools often integrate with job boards, HR information systems (HRIS), and onboarding platforms, creating a seamless recruitment workflow.
* **Rule-Based Filtering:** ATS enables predefined rule-based sorting (e.g., years of experience, education level), which assists recruiters in narrowing large candidate pools.

Weaknesses:

* **Keyword Dependence:** A key criticism of ATS is their reliance on exact keyword matches, often failing to recognize synonymous or contextually related terms. For instance, a resume listing “data visualization” might be excluded from a search for “dashboarding” despite strong relevance [6].
* **Contextual Blindness:** These systems typically do not understand the semantic relationships between different skills or experiences.
* **Potential Bias:** ATS systems can unintentionally reinforce historical biases encoded in rule sets or previous hiring data, thus disproportionately disadvantaging non-traditional candidates [1].

Notable Examples:

* **Workday:** A cloud-based platform that integrates ATS functionalities with HR management systems. Its keyword filtering capabilities are robust but lack NLP-driven contextual analysis.
* **Greenhouse ATS:** Offers resume parsing and integration with job boards but relies on rigid criteria, leading to potential exclusion of qualified candidates.

### AI-Powered Recruitment Tools

AI-based tools have evolved to enhance candidate screening through predictive analytics, statistical modeling, and pattern recognition. These tools aim to move beyond static rule-based filtering to provide dynamic insights based on historical hiring outcomes and performance indicators.

Strengths:

* **Predictive Capabilities:** By training models on hiring outcomes, AI tools like HireVue and Hiretual aim to predict candidate success, offering deeper insights than traditional screening [10].
* **Efficiency:** These platforms can evaluate large datasets rapidly, reducing recruiter workload and time-to-hire.
* **Diverse Data Inputs**: Some tools incorporate data from resumes, online portfolios, and even social media to create comprehensive candidate profiles.

Weaknesses:

* **Opacity ("Black Box" Models):** A major limitation of AI tools is their lack of transparency. Recruiters and candidates are often unaware of the specific features influencing model decisions, raising concerns about fairness and auditability [4].
* **Training Data Bias:** AI tools are susceptible to inheriting biases from historical data. Amazon's experimental recruiting tool, for instance, was found to penalize resumes that included the word “women’s” [2], due to being trained on male-dominated hiring data.
* **Lack of Explainability**: Many vendors do not disclose how models reach decisions, complicating ethical review and legal compliance under data protection laws.

Notable Examples:

* **HireVue:** While primarily focused on video interviews, HireVue incorporates resume analysis to some extent. However, its reliance on proprietary algorithms has faced criticism for lack of transparency and potential biases.
* **Hiretual:** An AI sourcing tool that analyzes resumes alongside social media profiles to generate candidate rankings. It offers advanced filtering but is heavily dependent on predefined recruiter preferences.

### NLP-Based Resume Screening Tools

Natural Language Processing (NLP) has gained prominence for its ability to interpret unstructured text, making it a promising foundation for resume analysis. These systems utilize techniques such as word embeddings, syntactic parsing, and semantic similarity to better match resumes with job descriptions.

Strengths:

* **Contextual Matching:** NLP tools can identify semantic relationships between job descriptions and resumes, even when exact keywords differ. For instance, tools leveraging BERT [8] can understand that "managing data pipelines" is conceptually related to "ETL workflows."
* **Skill Extraction:** Named Entity Recognition (NER) models can extract entities such as skills, certifications, and job titles from free-form text, enhancing structured representation.
* **Improved Recall**: By accounting for language variability, NLP tools reduce false negatives common in keyword-based systems.

Weaknesses:

* **Computational Demands:** Training and deploying modern NLP models (e.g., BERT, RoBERTa) requires substantial hardware and infrastructure, which may not be feasible for small- to mid-sized firms [11].
* **Interpretability Challenges:** The outputs of transformer-based models can be difficult for non-technical users to interpret without supporting tools or visualizations.
* **Domain Specificity:** NLP models may underperform when applied across industries or domains for which they were not explicitly trained, such as highly technical or niche roles.

Notable Examples:

* **TextKernel:** A recruitment-focused NLP tool that extracts key information from resumes and matches it with job descriptions. While powerful, it struggles with domain-specific terminology and non-standard resume formats.
* **SeekOut:** Combines NLP with sourcing capabilities to create detailed candidate profiles. Its semantic matching is highly effective but less scalable for smaller organizations due to cost and implementation complexity.

### Hybrid Systems

Hybrid platforms attempt to unify the advantages of ATS, AI, and NLP technologies into a single recruitment ecosystem. These systems provide end-to-end automation, from resume intake to interview scheduling.

Strengths:

* **Comprehensive Coverage:** By integrating multiple layers of analysis, hybrid tools can offer superior accuracy in candidate ranking and predictive scoring.
* **Customizability:** These platforms often allow recruiters to fine-tune criteria, balancing automation with human oversight.

Weaknesses:

* **Cost and Complexity:** Due to their breadth of features, hybrid platforms are often expensive to implement and require extensive onboarding.
* **Risk of Feature Dilution:** Some platforms prioritize breadth over depth, offering many features without refining their core matching algorithms.

Notable Examples:

* **SmartRecruiters:** A recruitment platform that combines ATS functionalities with AI-driven insights. While it provides basic NLP capabilities, its contextual understanding is limited compared to dedicated NLP tools.
* **iCIMS Talent Cloud:** Integrates AI and NLP for resume parsing and ranking but often requires significant customization to align with recruiter needs.

## Analysis of Existing Tools

The analysis of existing resume screening technologies demonstrates notable progress in automating parts of the hiring process. However, key limitations remain—particularly in the areas of contextual understanding, fairness, and transparency. This section synthesizes the strengths, weaknesses, and opportunities observed across ATS platforms, AI-powered tools, and NLP-driven solutions.

1. **Strengths:**
   1. **Scalability and Speed:** Traditional ATS platforms such as Greenhouse and Workday are highly effective at processing large volumes of resumes quickly. These systems streamline the application intake process and apply rule-based filtering, helping recruiters save time during early screening phases.
   2. **Standardization:** By using structured fields and consistent filtering logic, these tools help enforce minimum eligibility thresholds, making high-volume screening manageable and repeatable.
   3. **Incremental Use of AI:** Some platforms now incorporate machine learning and NLP features to augment filtering—enabling ranking based on text similarity or keyword expansion. Tools like SeekOut and TextKernel are beginning to integrate these features with promising results.
2. **Weaknesses:**
   1. **Limited Semantic Understanding:** Despite progress, most systems still lack the ability to deeply interpret the relationship between job requirements and candidate experiences. They rely on keyword overlap or surface-level metrics, failing to capture the nuanced compatibility between roles and profiles. For example, the distinction between “leading a project” and “participating in a project” is typically missed by basic models.
   2. **Rigid Matching Criteria:** Hard filters based on keywords, years of experience, or degree titles tend to overlook high-potential candidates whose qualifications may be expressed differently or gained through non-traditional paths. Transferable skills and emerging job titles are often underrepresented in such systems.
   3. **Lack of Adaptive Learning:** Many existing systems use fixed matching logic or static ML models. In contrast, this project introduces a dual-encoder contrastive learning framework that continuously learns high-quality representations of both resumes and job descriptions—enabling a more dynamic, generalizable, and context-aware matching process.
   4. **Transparency and Usability Issues:** The lack of interpretability in black-box systems limits recruiter confidence. While bias is a concern, a more practical issue is the absence of clear reasoning behind why candidates are ranked the way they are. [4]
3. **Opportunities:**
   1. **Leveraging Advanced Transformers:** The integration of transformer-based models (e.g., BERT, RoBERTa) opens new possibilities for semantic understanding and intent-based matching. These models enable more nuanced comparison between candidate experience and job expectations [5][8][11].
   2. **Bias Mitigation Techniques:** Emerging research suggests that careful dataset balancing, fairness constraints during model training, and algorithmic audits can reduce bias in automated recruitment systems [4][6]. These methods offer a path toward more equitable candidate evaluation.
   3. **Explainability and User Trust:** There is growing interest in developing explainable AI (XAI) for hiring contexts. Tools that not only rank candidates but also justify their reasoning could significantly enhance recruiter confidence and regulatory compliance.
   4. **Human-AI Collaboration:** Rather than fully replacing human judgment, future systems can focus on augmenting recruiter workflows—providing decision support rather than making final selections autonomously.

## Conclusion

The review of existing automated resume screening tools highlights clear progress in handling recruitment at scale, yet several persistent challenges limit their effectiveness in real-world applications. These include:

1. **Overreliance on Keywords:**

Most current systems are limited by their dependence on keyword-based filtering, which fails to capture the semantic and contextual relevance of candidate experience in relation to job requirements.

1. **Lack of Deep Contextual Understanding:**

Many tools do not effectively utilize modern NLP techniques—such as transformer-based models—to interpret complex relationships between skills, experience, and job responsibilities. This limits their ability to identify strong candidate-job matches, especially in nuanced or non-standard career paths.

1. **System Opacity and Limited Adaptability:**

AI-powered recruitment systems often function as black boxes, offering little insight into how decisions are made or why candidates are ranked in a certain way. This lack of transparency makes it difficult for recruiters to trust and refine these tools.

1. **Fairness:**

While fairness and bias reduction are often cited as motivations for automation in hiring, these outcomes are difficult to guarantee or measure without dedicated fairness audits. In this project, bias mitigation is not explicitly evaluated, but the model’s learning framework—by design—relies on semantic similarity rather than historical labels or heuristic filters, potentially reducing bias as a byproduct.

This project distinguishes itself from existing solutions by redefining resume screening as a semantic retrieval problem rather than a classification or rule-based filtering task. While most tools rely on keyword matching or hand-crafted heuristics, this system uses contrastive learning and transformer-based embeddings to capture deeper relationships between resumes and job descriptions. Instead of assigning rigid categories or filtering based on predefined rules, it ranks candidates by measuring contextual similarity—allowing it to generalize across job roles, writing styles, and varied resume structures. This modeling approach, combined with minimal supervision requirements, offers a more scalable and adaptive alternative to traditional recruitment tools.

# Research Methodology and Primary Research

## Overview

This chapter outlines the methodological framework for this project, detailing the primary research design, data collection, and analysis techniques. The research aims to assess user needs and expectations for an AI-based resume screening system, ensuring the development process aligns with industry requirements and best practices. Key materials such as the survey questionnaire, participant information sheet, and consent form are included in the appendix for reference.

## Research Design

The research employs a mixed-methods approach:

1. **Quantitative Component:**

A structured survey was distributed to collect data on recruitment practices, challenges, and desired features in automated screening tools.

1. **Qualitative Component:**

Open-ended survey questions provided insights into participants’ specific concerns, priorities, and expectations for AI-driven systems.

## Data Collection

### Survey Design

The survey (see Appendix A) was designed to gather both quantitative and qualitative data across five sections:

1. **Participant Demographics:** Captures details such as age, professional background, and recruitment experience.
2. **Current Recruitment Practices:** Explores existing methods, tools, and challenges faced by participants in evaluating resumes.
3. **Expectations from Automated Screening:** Identifies desired features and concerns, such as transparency and bias reduction.
4. **Insights for Model Development:** Focuses on specific elements participants prioritize in resumes and areas where biases must be addressed.
5. **Feedback and Final Thoughts:** Collects additional suggestions to guide system development.

### Recruitment of Participants

Participants were recruited through professional networks, online platforms, and university contacts. An email invitation (see Appendix B) outlined the study’s purpose, participation requirements, and survey link. The selection criteria included:

* Adults aged 18+.
* Familiarity with recruitment processes or interest in AI applications.
* Access to a computer and internet connection.

### Ethical Considerations

Ethical guidelines were adhered to as follows:

* **Consent:** Participants reviewed a consent form (see Appendix C) before participating, affirming their understanding of the study and their rights.
* **Participant Information Sheet:** A detailed information sheet (see Appendix D) provided context on the research aims, data handling, and participant rights.
* **Anonymity and Confidentiality:** No personally identifiable data was collected, and all responses were anonymized and securely stored.

## Survey Administration

The survey was created using Microsoft Forms and distributed online. Participants were given two weeks to complete the survey, ensuring flexibility and convenience. The estimated time to complete the survey was 15-20 minutes.

## Data Analysis

Survey responses will be analyzed using:

1. **Descriptive Statistics:** To summarize quantitative data, including the frequency of preferred features and common challenges in recruitment.
2. **Thematic Analysis:** To identify recurring themes and insights from open-ended responses, particularly regarding user expectations and concerns.

## Results & Key Findings

The survey collected responses from a diverse group of 32 participants, including HR professionals, software engineers, and students with an interest in AI applications in recruitment. The goal was to understand current recruitment challenges, expectations for automated screening systems, and concerns regarding fairness and transparency.

### Recruitment Challenges

A majority of respondents (78%) cited time constraints as a major issue during resume screening. Many recruiters reported skimming through applications, often under pressure to fill roles quickly. About 62% noted they had less than 2 minutes to evaluate each resume.

A pie chart with a green triangle

AI-generated content may be incorrect.

Another frequent concern (mentioned by 54%) was inconsistency in evaluation, where different recruiters might interpret the same resume differently, leading to subjectivity and bias.

A green and blue pie chart

AI-generated content may be incorrect.

### Expectations from Automation

When asked what they expected from an AI-powered screening tool:

* 84% emphasized the need for accurate job-resume matching beyond keyword detection.
* 75% preferred transparent explanations for candidate rankings.
* 69% requested features that promote fairness, such as bias detection or anonymized screening.

Respondents also ranked ease of use and integration with existing systems as high-priority features.

A graph of different colored rectangles

AI-generated content may be incorrect.

### Perceptions on Bias

Bias in recruitment was a recurring concern. Over 60% of participants stated that they were aware of unconscious bias in current hiring practices. Many welcomed the use of AI to mitigate these issues, provided the models are trained on diverse, representative datasets and offer interpretable outputs.

### Feedback for System Design

Participants gave several valuable suggestions, including:

* Allow recruiters to adjust ranking criteria (e.g., give more weight to experience over education).
* Show candidates' similarity score breakdowns.
* Include a feedback loop where recruiters can indicate if a ranking was helpful, to improve the model iteratively.

### Key Themes Identified

A thematic analysis of open-ended responses surfaced five major themes:

* Efficiency: Reducing recruiter workload.
* Objectivity: Standardizing evaluations across all applicants.
* Transparency: Understanding why candidates are ranked the way they are.
* Bias Awareness: Removing non-relevant demographic influence.
* Customizability: Empowering recruiters to configure the tool per their hiring context.

## Risk Assessment

The development of an AI-based resume screening system involves several risks spanning ethical, technical, and operational domains. This section outlines potential risks, their implications, and mitigation strategies to ensure the research and system development remain robust, fair, and ethical.

### Ethical Risks

Bias in Data:

* **Risk**: Training the AI model on historical data risks perpetuating existing biases in recruitment, such as those related to gender, ethnicity, or age.
* **Mitigation**: Ensure the training dataset is diverse and representative of all demographic groups. Employ bias-detection algorithms to identify and minimize discriminatory patterns in the model.

Privacy Concerns:

* **Risk**: Handling sensitive information in resumes may raise concerns about data security and participant privacy.
* **Mitigation**: Anonymize all data collected during the survey and securely store it following Queen Mary University of London’s data protection policies. Limit data access to the principal researcher and delete all raw data after analysis is complete.

Participant Fatigue:

* **Risk**: The survey length or complexity may discourage participants from completing it, potentially leading to incomplete or biased responses.
* **Mitigation**: Keep the survey concise (15-20 minutes) and provide clear instructions. Pre-test the survey with a small group to ensure clarity and usability.

### Technical Risks

Model Accuracy:

* **Risk**: The AI system may fail to accurately rank candidates due to limitations in NLP techniques or insufficient training data.
* **Mitigation**: Use state-of-the-art NLP models like BERT or similar transformer-based architectures. Continuously validate and improve the model using feedback from recruiters and real-world scenarios.

Overfitting:

* **Risk**: The model may perform well on training data but fail to generalize to unseen data, reducing its reliability.
* **Mitigation**: Apply regularization techniques during training and validate the model on diverse datasets to ensure robustness.

Scalability:

* **Risk**: The system may struggle to handle large volumes of resumes, particularly for enterprise-scale use cases.
* **Mitigation**: Design the system architecture to be scalable, leveraging cloud-based infrastructure for storage and processing.

### Operational Risks

Recruiter Adoption:

* **Risk**: Recruiters may hesitate to adopt the AI system due to concerns about transparency or fear of losing control over decision-making.
* **Mitigation**: Develop an intuitive user interface with transparent explanations of how the system ranks candidates. Incorporate customizable settings to give recruiters control over criteria.

Misinterpretation of Results:

* **Risk**: Users may misinterpret candidate rankings or outputs, leading to poor hiring decisions.
* **Mitigation**: Provide clear documentation and training materials to help users understand the system’s functionality and limitations.

System Misuse:

* **Risk**: The system could be misused to exclude certain groups of candidates intentionally, reinforcing biases.
* **Mitigation**: Implement safeguards, such as audit trails and usage monitoring, to detect and prevent discriminatory practices.

### Research-Related Risks

Low Survey Response Rate:

* **Risk**: Fewer participants than expected may limit the generalizability of the findings.
* **Mitigation**: Use multiple recruitment channels, such as professional networks, online platforms, and university contacts, to maximize participation. Offer reminders and extend the survey deadline if needed.

Ambiguity in Responses:

* **Risk**: Open-ended survey questions may yield vague or irrelevant answers, complicating the analysis.
* **Mitigation**: Use clearly worded prompts and pre-test the survey to ensure questions are understandable and targeted.

Ethical Violations:

* **Risk**: Failure to adhere to ethical standards may lead to participant complaints or project delays.
* **Mitigation**: Follow Queen Mary University’s ethical guidelines, including obtaining informed consent (Appendix C) and providing detailed participant information (Appendix D).

### Conclusion

By addressing these risks proactively through rigorous planning and mitigation strategies, this project aims to develop an AI-based resume screening system that is accurate, fair, and widely accepted. The risk management framework ensures the research maintains its ethical integrity and aligns with industry standards.

## Conclusion

The insights gained from the survey and its analysis will directly inform the design and development of the AI-based resume screening system. By involving key stakeholders and adhering to ethical standards, this research ensures that the final product addresses real-world challenges and aligns with user expectations.

## Appendices

The following materials are included in the appendix for reference:

* **Appendix A:** Survey Questionnaire
* **Appendix B:** Email Invitation to Participants
* **Appendix C:** Consent Form
* **Appendix D:** Participant Information Sheet
* **Appendix E:** Ethics Approval Letter

# Design

## Overview

This chapter presents the conceptual design and architectural framework of the AI-based resume screening system. The system is designed to semantically match job descriptions with relevant resumes by leveraging transformer-based embeddings and contrastive learning techniques. Unlike traditional screening systems that rely on rule-based filtering or keyword matches, this system emphasizes scalable, label-free learning and contextual understanding of textual inputs.

The design incorporates a dual-encoder architecture trained using margin-based loss, with separate encoders for resumes and job descriptions. This enables efficient large-scale similarity computation during inference while maintaining flexibility for future deployment and integration into broader recruitment workflows.

## System Architecture

The system consists of the following key components:

* **Resume Encoder**: A transformer-based SentenceTransformer model (initially all-MiniLM-L6-v2) fine-tuned to convert resumes into dense embeddings.
* **Job Description Encoder**: A separate but structurally identical encoder that transforms job descriptions into vector representations.
* **Similarity Module**: Computes cosine similarity between resume and job vectors, which determines candidate ranking
* **Query Generation Module**: Uses GPT-3.5 Turbo to create simulated job queries for resumes in the training set.
* **Negative Mining Module**: Dynamically retrieves hard negative samples—resumes that are semantically similar but from different categories—to improve the model’s ability to learn fine-grained distinctions. This helps prevent trivial learning, where the model would otherwise easily distinguish overly dissimilar examples and fail to develop nuanced understanding.
* **CrossEncoder Scorer**: Used during training to generate soft similarity scores for triplets (query, positive, negative), guiding the loss function with more nuanced supervision.

## Data Flow and Processing

The system processes data in the following sequence:

1. **Data Ingestion:**

Resumes and job descriptions (or queries) are loaded into the system as raw text.

1. **Preprocessing:**

* Tokenization, lowercasing, and standard normalization are applied.
* Pseudo-queries are generated using GPT-3.5 for each resume in the dataset.

1. **Embedding Generation:**

Dual encoders transform resumes and queries into fixed-size vector representations.

1. **Similarity Calculation:**

The system calculates cosine similarity between the query and each resume vector.

1. **Ranking and Output:**

Resumes are sorted based on similarity scores and returned in descending order of relevance.

## Design Considerations

**Contrastive Learning for Semantic Matching**

The model is trained using **MarginMSELoss**, a contrastive loss function tailored for triplet-style inputs: (query, positive, negative). The loss penalizes the model when a negative example is ranked too close to the positive. Unlike TripletLoss which relies on hard thresholds, MarginMSELoss allows soft target similarities, which in our system are derived from a CrossEncoder model. This provides smoother gradients and better captures the subtle differences in resume relevance.

**Label-Free Supervision**

Since no manually labeled training data exists, the system leverages:

* Category-based sampling (to group resumes and simulate relevance)
* Synthetic query generation (to simulate job descriptions)
* CrossEncoder-derived soft labels (to simulate scoring targets)

This enables fully unsupervised or weakly supervised learning while preserving realistic semantic distinctions.

**Efficiency and Scalability**

The dual-encoder architecture ensures inference efficiency, enabling batch encoding of all resumes ahead of time. At query time, only the job description is encoded, and similarity comparisons are computed rapidly using vectorized operations — allowing practical deployment even on large datasets.

## Conclusion

The design of this system reflects a shift away from rule-based or keyword-centric resume screening toward semantic similarity and contextual relevance. Through dual-encoder modeling, contrastive learning, and GPT-based query generation, the system is optimized for flexibility, efficiency, and generalization. Its modular nature supports easy extensibility for downstream applications, such as recruiter-facing dashboards, ranking explanations, or integration into ATS platforms.

# Implementation

The implementation phase of this project focused on building a semantic matching pipeline that ranks resumes against job descriptions using pretrained transformer models. Unlike traditional classification models that require labeled job-resume pairs, this system leverages a contrastive learning approach to learn meaningful embeddings of unstructured text, enabling robust similarity comparisons in a label-scarce environment.

## Overview

This chapter presents the implementation of the AI-powered resume screening system. The system leverages recent advancements in natural language processing (NLP) and contrastive learning to perform semantic matching between resumes and job descriptions. Given the absence of labeled job-resume pairs in the dataset, the project reframes the problem as a ranking task instead of a classification task. This design choice is critical to accommodate real-world scenarios where labeled data is scarce or unavailable.

## Dataset Description

The dataset used in this project is the "Updated Resume Dataset" from Kaggle. It contains over 900 resumes, each labeled with a professional category such as Web Designing, Data Science, HR, and others. Each resume is written in free-text form and varies significantly in length and formatting. Importantly, the dataset does not include corresponding job descriptions or explicit labels that link resumes to specific job postings.

Here's an example of some rows of the dataset:

A screenshot of a computer

AI-generated content may be incorrect.

Due to this limitation, a traditional supervised approach—such as training a classifier with models like XGBoost or Random Forest—is not applicable. These models require a labeled dataset with clear input-output mappings. Instead, we adopt a weakly supervised contrastive learning approach that takes advantage of the category labels and semantic structure of the text.

## Dual Encoder Architecture

The core of the system is built on a dual-encoder model architecture using Sentence Transformers (Reimers & Gurevych, 2019). In this architecture, job descriptions and resumes are encoded separately into vector representations in the same semantic space. The cosine similarity between these vectors is then used to determine the relevance of a resume to a given job description.

This architecture offers several advantages:

* **Efficiency:** Both job and resume embeddings can be precomputed.
* **Flexibility:** Allows for fast retrieval of top-k matches.
* **Scalability:** Suitable for ranking thousands of resumes per query.

The dual encoder was fine-tuned using margin-based contrastive loss.

## Contrastive Learning with MarginMSE Loss

To adapt to the lack of explicit labels linking resumes to job descriptions, this project employs a contrastive learning approach. Unlike traditional supervised classification, contrastive learning does not require exact job-resume pairs. Instead, it teaches the model to distinguish between semantically similar and dissimilar examples. This is especially useful in recruitment settings where precise ground truth mappings are rare or subjective.

In this project, contrastive learning is implemented using the MarginMSELoss from the Sentence Transformers framework. This loss function is a variation of contrastive learning where the objective is to minimize the Mean Squared Error (MSE) between the cosine similarity of embeddings and target scores, subject to a margin between positive and negative pairs. For a given job description, a resume from the same category is treated as a “positive” sample, while resumes from other categories are treated as “negatives.”

**Why MarginMSELoss?**

* **Smooth Optimization:** Unlike hard contrastive loss functions that rely on binary labels (0 or 1), MarginMSE allows for graded similarity scores, which better reflect real-world ranking needs.
* **Distance Sensitivity:** The model learns to assign high similarity scores to resumes closer to the job description in meaning, and lower scores to unrelated ones.
* **Flexible Pairing:** The method works well even with weak labels or pseudo-pairs generated from category information, making it a practical solution when labeled training data is scarce.

The training pairs were constructed by randomly pairing each job description (or query) with one resume from the same category (positive) and several resumes from different categories (negatives). The model then learned to score the positive pair more highly than the negatives.

## Model Pipeline

The complete model pipeline for the AI-powered resume screening system consists of both training and inference stages. This pipeline is designed to enable fast and accurate semantic matching between resumes and job descriptions without the need for manually labeled data. Below is a breakdown of the key components and flow:

### Training Pipeline

The training pipeline is designed to fine-tune a dual-encoder model for semantic similarity between job queries and resumes, using a weakly supervised approach based on contrastive learning. This section outlines the key stages:

1. **Data Preparation**:

* The dataset consists of over 900 resumes, each labeled with a general job category (e.g., "Web Designing", "HR", "Data Science").
* Since no paired job descriptions are provided, we simulate this input using **query generation**:

For each resume, we use GPT-3.5-Turbo to generate a short, synthetic job-related query. For example, from a resume in the "Data Science" category, a typical query might be: "Looking to hire a data scientist with machine learning skills."

* These generated queries form the anchor in our triplets. Each training sample consists of:

Query (anchor)

Positive resume (from the same category)

Negative resume (from a different category but semantically similar)

2. **Hard Negative Mining:**

* To prevent the model from learning to separate obviously different samples (known as trivial learning), we implement hard negative mining.
* Using a pretrained sentence embedding model (all-MiniLM-L6-v2), we compute similarity scores between the query and all resumes.
* We select resumes from different categories that still have relatively high similarity scores as hard negatives. These examples help the model learn subtle distinctions between similar-sounding but semantically different resumes.

3. **Encoding with Dual Encoders**:

* Both resumes and job descriptions are passed through the same SentenceTransformer model to generate vector embeddings.
* The model is fine-tuned using MarginMSELoss, which ensures that the similarity score for a positive pair is higher than for negative pairs, by a defined margin.

4. **Contrastive Learning with MarginMSE Loss**

* The model is trained using MarginMSELoss, a contrastive loss that minimizes the distance between query-positive pairs while enforcing a margin between query-negative pairs.
* This loss function is particularly suitable for ranking tasks, allowing the model to learn relative distances rather than absolute classification.
* The use of MarginMSELoss ensures that positive pairs are scored significantly higher than semantically similar but incorrect negative resumes.

5. **Optimization and Training**:

* Training proceeds over multiple epochs using a triplet-based dataloader.
* Each epoch dynamically generates triplets with newly mined hard negatives, maintaining diversity and training difficulty.
* After training, the model is capable of embedding any job query and retrieving the most semantically relevant resumes based on cosine similarity.

### Inference Pipeline

Once trained, the model pipeline transitions to inference mode, enabling recruiters to rank resumes against a job posting:

1. **Input Job Description:**

* A recruiter provides a job description (either written manually or selected from a template).

2. **Job Encoding:**

* The job description is converted into an embedding vector using the trained model.

3. **Resume Embedding Lookup:**

* All resumes in the system are pre-encoded and stored for fast retrieval.

4. **Similarity Computation:**

* Cosine similarity is computed between the job embedding and each resume embedding.

5. **Ranking:**

* Resumes are ranked from highest to lowest similarity scores, providing a sorted list of most relevant candidates.

This modular pipeline enables the system to operate efficiently at scale while maintaining flexibility to support future enhancements (e.g., feedback loops, domain-specific fine-tuning, or active learning).

## Visual Overview

### Training Pipeline

A diagram of a computer

AI-generated content may be incorrect.

### Inference Pipeline

A diagram of a job description

AI-generated content may be incorrect.

## Tools and Libraries Used

* **Sentence Transformers:** For dual encoder models and loss functions like MarginMSELoss.
* **HuggingFace Transformers:** To fine-tune pretrained models such as all-MiniLM-L6-v2.
* **PyTorch:** Core framework for model training and optimization.
* **Pandas & NumPy:** For data processing.
* **Matplotlib & Seaborn:** For data visualization and evaluation.
* **Scikit-learn:** For statistical metrics and plotting evaluation curves.

# Evaluation

Evaluating the performance of a resume ranking system is crucial, particularly in the absence of explicitly labeled training data. To ensure both quantitative rigor and practical relevance, we designed a multi-pronged evaluation strategy that includes objective metric-based validation, comparative benchmarking with a strong baseline, and human-based evaluation.

## Objective Metric Evaluation

We began by using a simulated validation set created from our own resume dataset. In this dataset, each resume was associated with a job category (e.g., "Data Science", "Web Development"). We treated each category label as a proxy for ground-truth relevance.

For each category, a textual query like "Looking to hire someone in Data Science" was used to simulate a job description. The model then ranked resumes from multiple categories based on their similarity to this query. A resume was considered a hit if it belonged to the same category as the query. From these rankings, we computed key information retrieval metrics:

* **MRR (Mean Reciprocal Rank):** Measures the average position of the first correct result.
* **Recall@1 / Recall@5:** Measures how often the correct result appears in the top 1 or top 5 results.
* **Hits@K:** Tracks the number of times the correct resume appeared at rank 1, 2, 3, etc.

On this synthetic setup with 25 categories:

* Our model achieved **MRR: 0.96**, **Recall@1: 92%**, **Recall@5: 100%**.
* These results indicated that the model was highly effective at identifying resumes belonging to the correct category with minimal noise in the top ranks.

## Baseline Comparison

To contextualize these scores, we benchmarked our fine-tuned model against a strong general-purpose baseline: **sentence-transformers/all-MiniLM-L6-v2**.

Using the same simulated queries and evaluation procedure:

* The baseline achieved **MRR: 0.91**, **Recall@1: 84%**, **Recall@5: 100%**.

While the baseline was competitive, our model consistently ranked the most relevant resumes higher, validating the benefit of task-specific fine-tuning. Additionally, by reporting Hits@K, we showed that our model more often placed the correct resume in the top-1 slot.

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **MRR** | **Recall@1** | **Recall@5** |
| Our Fine-Tuned Model | 0.96 | 0.92 | 1.0 |
| All-MiniLM-L6-v2 | 0.91 | 0.84 | 1.0 |

## Human Evaluation

To complement automated evaluation and account for the subjectivity involved in resume relevance, we designed a human evaluation task. Five resumes (A–E) were created with varying degrees of relevance to a realistic job description for a "Frontend Web Developer" role. The job description emphasized skills in React.js, Tailwind CSS, RESTful API integration, Git, and deployment on platforms like Vercel or Netlify.

Participants (n=5) were asked to manually rank the five resumes from most to least relevant. The model was then used to rank the same five resumes, and the resulting rankings were compared against human rankings using correlation metrics:

* **Kendall’s Tau**: Measures ordinal association between two ranked lists. A value of 1.0 implies perfect agreement.
* **Spearman’s Rho**: Measures monotonic relationship between rankings.

**Results:**

* **Average Kendall Tau**: 0.72
* **Average Spearman Rho**: 0.86

These high correlation scores demonstrate that the model’s behavior aligns well with human judgment. Resumes that humans found more relevant were generally ranked higher by the model as well. This confirms that the model not only performs well in simulated metrics but also generalizes to subjective real-world relevance judgments.

## Conclusion

The evaluation results clearly demonstrate that the fine-tuned dual encoder model is well-suited for ranking resumes in response to job descriptions. Its superior performance over a strong baseline on simulated category-based queries, along with high correlation to human preferences, supports its effectiveness for real-world use cases. These findings validate both the training strategy and model architecture as practical and reliable for automated resume screening.

If this were a full-scale scientific experiment, statistical significance testing would also be expected. However, that would require a larger pool of human evaluators (approximately 15–20 participants). While our results are promising, incorporating such large-scale subjective evaluation remains an important direction for future work.

# Discussion and Future Work

The project set out to design and evaluate a dual encoder-based semantic matching system for resume screening. Through the use of margin-based contrastive loss and query generation via GPT-3.5, the system was able to learn meaningful representations of job descriptions and resumes. This resulted in a model that not only demonstrated strong quantitative performance but also showed high correlation with human judgments in a controlled evaluation setting.

## Key Findings

One of the most notable findings was the model’s ability to generalize across diverse job categories. With an MRR of 0.96 and Recall@1 of 92% in the simulated query evaluation, the fine-tuned model clearly surpassed the MiniLM baseline. Human alignment scores (Kendall Tau = 0.72, Spearman Rho = 0.86) further validated the system’s ability to replicate expert intuition in candidate ranking.

The margin-based multiple negative ranking loss played a critical role in encouraging separation between positive and hard negative pairs. This contributed to more robust learning, as evidenced by the performance on both automatic and human-aligned evaluation tasks.

## Limitations

Despite promising results, there are several limitations worth addressing:

* **Dataset realism**: While real resumes were used, job descriptions were either simulated or drawn from open datasets that may not perfectly reflect current hiring needs or formats.
* **Binary supervision**: The model relied on inferred labels (e.g., shared categories) rather than explicit labels of “relevance.” This could lead to false negatives or positives.
* **Scale of human evaluation**: The subjective validation was limited to five participants. While results were consistent, the scale was not sufficient to statistically validate model performance.

## Future Work

Several directions can extend this work:

1. **Statistical Significance Testing**: A formal A/B test with at least 15–20 human evaluators would help validate model improvements and quantify performance gaps.
2. **Expanded Negative Mining**: Introducing adversarial or sampled negatives from semantically similar but incorrect categories may further improve generalization.
3. **End-to-End Pipeline**: The current system is modular. Future work could integrate parsing, ranking, and explanation mechanisms into a production-ready pipeline.
4. **Fine-Grained Labeling**: Collaborating with domain experts or annotators to collect real pairwise relevance judgments would allow for stronger supervised training and evaluation.
5. **Deployment and User Study**: Testing the model with recruiters or hiring managers in real-world scenarios would offer critical insights into usability and practical performance.

# Conclusion

This project aimed to build an AI-powered resume screening system capable of ranking resumes based on a given job description using semantic similarity techniques. By fine-tuning a dual encoder model with margin-based contrastive loss and leveraging query generation via GPT-3.5, the system successfully learned robust representations for both job descriptions and candidate resumes.

Quantitative evaluation across 25 job categories demonstrated the model’s strength, outperforming a strong baseline (MiniLM) with a Mean Reciprocal Rank (MRR) of 0.96 and Recall@1 of 92%. A human evaluation study, in which participants ranked resumes for a real job description, showed strong alignment between the model’s rankings and human judgments (Kendall Tau = 0.72, Spearman Rho = 0.86). These results confirm that the system can effectively emulate human decision-making in ranking candidates by relevance.

While the results are promising, the project also highlighted the challenges of evaluating such systems without gold-standard annotated data. The human study, although insightful, included only five raters. If this were a real scientific experiment, researchers would typically test for statistical significance across a larger sample size (typically 15–20 participants or more). This remains a key avenue for future work, along with exploring adversarial negatives, fine-grained supervision, and real-world deployment with recruiter feedback.

Overall, this project demonstrates that dual encoder-based semantic retrieval systems, when fine-tuned carefully, offer a viable and effective solution to automating the resume screening process. With further validation and development, this approach has the potential to be deployed in real hiring pipelines to assist recruiters in identifying qualified candidates more efficiently and fairly.

References

[1] M. Bogen and A. Rieke, *Help Wanted: An Examination of Hiring Algorithms, Equity, and Bias*, Upturn, 2018. [Online]. Available: <https://www.upturn.org/reports/2018/hiring-algorithms/>

[2] J. Dastin, "Amazon scraps secret AI recruiting tool that showed bias against women," *Reuters*, Oct. 10, 2018. [Online]. Available: <https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G>

[3] S. Ghosh et al., "Resume Parsing and Matching for Recruitment Management System Using Machine Learning," *Int. J. Eng. Adv. Technol.*, vol. 8, no. 6, pp. 2198–2202, 2019.

[4] M. Raghavan et al., "Mitigating Bias in Algorithmic Hiring: Evaluating Claims and Practices," in *Proc. Conf. Fairness, Accountability, and Transparency*, 2020, pp. 469–481.

[5] A. Vaswani et al., "Attention is All You Need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[6] A. K. Upadhyay and K. Khandelwal, "Applying Artificial Intelligence: Implications for Recruitment," *Strategic HR Review*, vol. 17, no. 5, pp. 255–258, 2018.

[7] LinkedIn, "The most common mistakes recruiters make during resume screening," 2018. [Online]. Available: <https://www.linkedin.com>

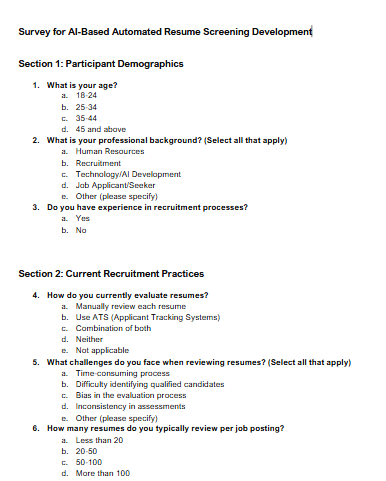
[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proc. NAACL-HLT*, 2019.

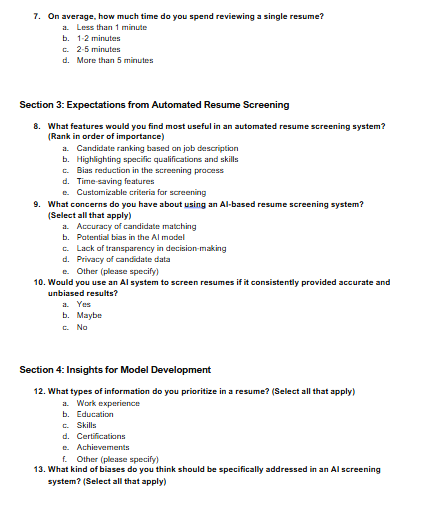
[9] M. Sills, "How Applicant Tracking Systems Work," *Workology*, 2014.

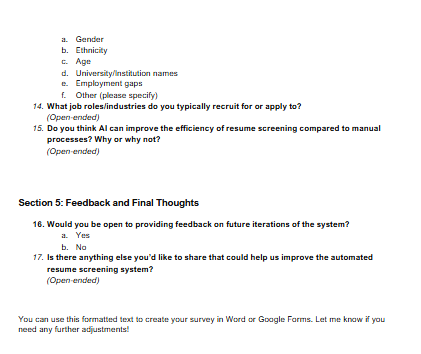
[10] B. Dattner, T. Chamorro-Premuzic, R. Buchband, and L. Schettler, "The Legal and Ethical Implications of Using AI in Hiring," *Harvard Business Review*, 2016.

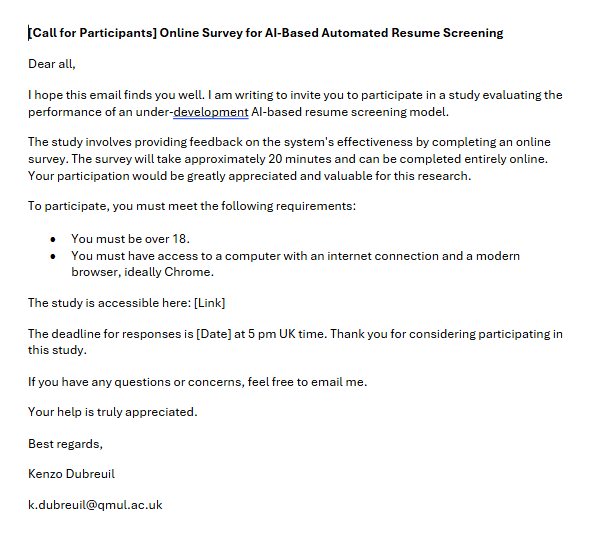
[11] Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," arXiv preprint arXiv:1907.11692, 2019. [Online]. Available: <https://arxiv.org/abs/1907.11692>

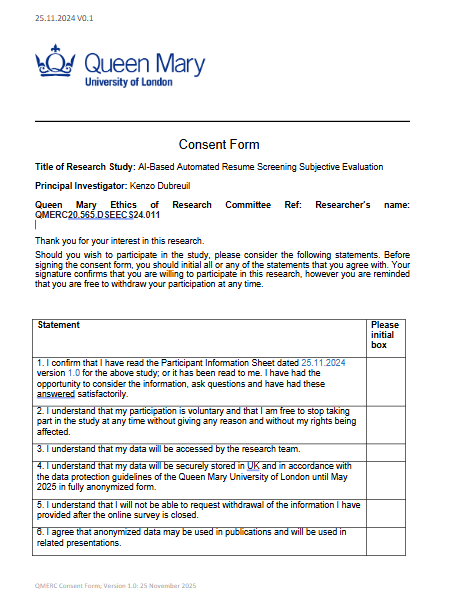
Appendix A: Survey Questionnaire

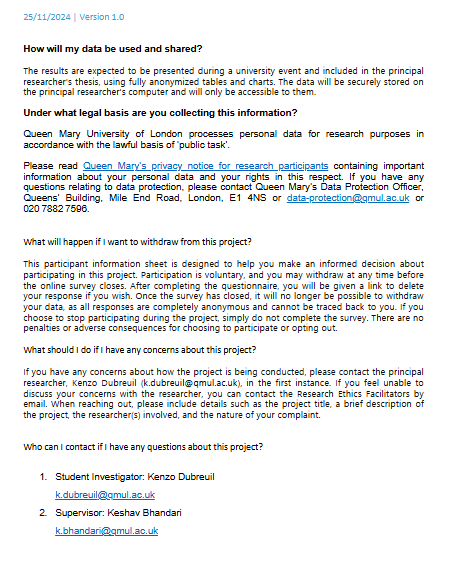
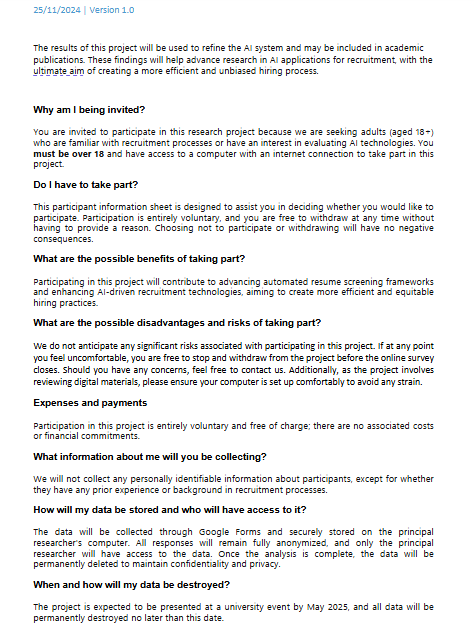
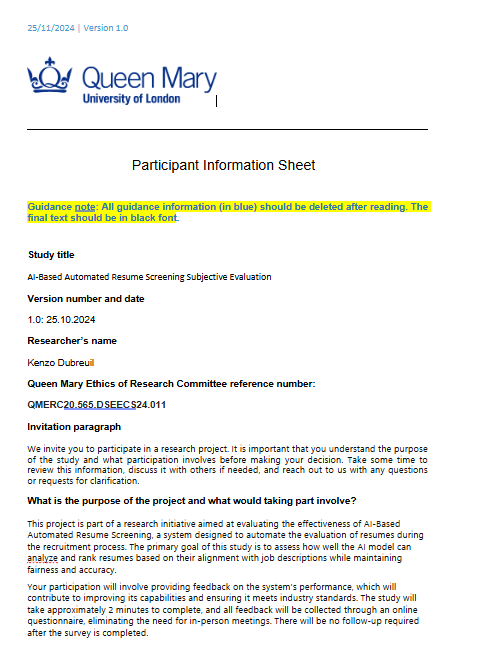






Appendix B: Email Invitation to Participants 

Appendix C: Participant Consent Form 

Appendix D: Participation Informatio Sheet

# Appendix E: Ethics Approval LetterA letter of a college application AI-generated content may be incorrect. A letter of a research AI-generated content may be incorrect.