

AI-powered Resume Screening and Ranking System

A Project Report

submitted in partial fulfillment of the requirements

of

AICTE Internship on AI: Transformative Learning

with

TechSaksham – A joint CSR initiative of Microsoft & SAP

by

Mridul Singh Rajput, mridul.raj.2222@gmail.com

Under the Guidance of

Saomya A. Chaudhary

ACKNOWLEDGEMENT

This project, focused on developing an **AI-powered resume screening and ranking system**, has been a journey of learning and exploration. I would like to express my sincere gratitude to all those who have contributed to its progress.

Firstly, I extend my deepest appreciation to my project supervisor, Soamya A. Chaudhary, for their invaluable guidance, support, and encouragement throughout this endeavor. Their expertise and insights have been instrumental in shaping the project's direction and ensuring its successful development.

I am also grateful to Pavan Sumohana, whose contributions have enriched this project significantly. Their feedback, suggestions, and assistance have been invaluable in overcoming challenges and refining the system's capabilities.

Furthermore, I acknowledge the support of TechSaksham, edunet Foundations. Their resources and collaborative environment have facilitated the research and development process.

Finally, I would like to thank the open-source community and the developers of the various tools and technologies that have been instrumental in building this system. Their contributions have made this project possible.

As this project enters its next phase, with the integration of machine learning and further research, I am excited to continue this journey of innovation and contribute to the advancement of AI-driven recruitment solutions.

Yours Sincerely,

Mridul Singh Rajput

ABSTRACT

This project addresses the challenge of efficiently screening and ranking a large volume of resumes against specific job requirements. Traditional manual resume screening is time-consuming and prone to human bias. To automate this process, an **AI-powered resume screening and ranking system** was developed. The objective was to create a tool that accurately assesses resume relevance based on job descriptions, thereby streamlining the hiring process.

The methodology employed **Natural Language Processing (NLP)** techniques, specifically **TF-IDF vectorization** and **cosine similarity**, to quantify the similarity between resume content and job descriptions. PDF resumes were processed using the **PyPDF2** library to extract textual data. The **scikit-learn** library was used for vectorization and similarity calculations, while pandas facilitated data manipulation and result presentation. A user-friendly interface was built using the **Streamlit framework**, allowing for easy input of job descriptions and uploading of resumes.

Key results demonstrated the system's ability to effectively rank resumes based on their relevance to a given job description. The cosine similarity scores provided a quantifiable measure of candidate suitability, enabling recruiters to quickly identify top candidates. The system significantly reduced the time and effort required for initial resume screening.

In conclusion, this AI resume screening and ranking system offers a practical and efficient solution for automating the initial stages of candidate selection. By leveraging NLP and machine learning, it enhances the speed and accuracy of resume screening, ultimately contributing to a more effective hiring process. Future work may focus on incorporating more advanced NLP techniques, such as named entity recognition and semantic analysis, to further improve the system's accuracy and capabilities.

TABLE OF CONTENT

Abstract	I
Chapter 1. Introduction	6
1.1 Problem Statement	6
1.2 Motivation	6
1.3 Objectives	7
1.4 Scope of the Project	7
Chapter 2. Literature Survey	9
Chapter 3. Proposed Methodology	11
Chapter 4. Implementation and Results	15
Chapter 5. Discussion and Conclusion	19
References	21

LIST OF FIGURES

[illegible]

CHAPTER 1

Introduction

1.1 Problem Statement:

The manual screening and ranking of candidate resumes is a time-consuming and resource-intensive process for recruiters and hiring managers. With a large volume of applications received for each job opening, it becomes challenging to efficiently identify the most qualified candidates. Traditional methods often rely on subjective assessments and key word matching, which can lead to overlooking qualified individuals and introducing human bias. This inefficiency results in delayed hiring processes, increased costs, and potential loss of top talent.

1.2 Motivation:

This project was chosen to address the inefficiencies and limitations of manual resume screening by developing an automated AI-powered system. The motivation stems from the desire to streamline the hiring process, reduce human bias, and improve the accuracy of candidate selection.

Potential applications include:

- **Faster Hiring:** Automating the initial screening process allows recruiters to focus on qualified candidates sooner.
- **Reduced Costs:** Minimizing manual effort reduces hiring costs and improves resource allocation.
- **Improved Candidate Selection:** AI-driven analysis provides a more objective and consistent evaluation of resumes.
- **Enhanced Recruiter Productivity:** Recruiters can focus on higher-value tasks, such as interviews and candidate engagement.
- **Increased Diversity:** Reducing human bias can contribute to a more diverse and equitable hiring process.

The impact of this project lies in its ability to transform the traditional hiring process by leveraging AI to make it more efficient, accurate, and equitable.

1.3 Objective:

The primary objectives of this project are:

- To develop an AI-powered system that can automatically extract text from PDF resumes.
- To create a ranking system that scores resumes based on their relevance to a given job descriptions.
- To develop a user-friendly interface using Streamlit for easy interaction with the system.

1.4 Scope of the Project:

The scope of this project is limited to:

- Developing a system that processes PDF resumes and compares them to a single job description.
- Implementing TF-IDF vectorization and cosine similarity as the primary NLP techniques for similarity assessment.
- Creating a Streamlit web application for user interaction.
- Evaluating the system's performance based on cosine similarity scores.

Limitations include:

- The system's current implementation primarily focuses on textual content and does not incorporate visual or layout analysis of resumes.
- The system's accuracy is dependent on the quality and format of the input resumes and job descriptions.
- The project does not include advanced NLP techniques such as Named Entity Recognition (NER), or transformer models.

- The project does not include integration with Applicant Tracking Systems (ATS) at this time.
- The project does not include any bias mitigation techniques.
- The restriction that only resumes in PDF format are accepted; documents in DOCX format are not supported.

CHAPTER 2

Literature Survey

2.1 Review relevant literature or previous work in this domain.

- The domain of AI-powered resume screening and ranking draws upon a rich body of literature in Natural Language Processing (NLP), Information Retrieval (IR), and Machine Learning (ML).
- Studies have explored various techniques for text similarity, document classification, and information extraction.
- IR research has contributed methodologies for retrieving relevant documents from large datasets, such as vector space models and ranking algorithms.
- ML has provided models for classification and regression, which can be adapted for resume screening tasks.
- Existing work has also investigated the use of rule-based systems and keyword matching for resume analysis.

2.2 Mention any existing models, techniques, or methodologies related to the problem.

- **Keyword Matching:** A basic approach that identifies resumes containing specific keywords from the job description.
- **TF-IDF (Term Frequency-Inverse Document Frequency):** A statistical measure used to evaluate the importance of words in a document relative to a collection of documents.
- **Cosine Similarity:** A metric used to calculate the similarity between two vectors, often used to compare the TF-IDF vectors of resumes and job descriptions.
- **Support Vector Machines (SVMs) and other Classification Models:** Machine learning algorithms that can be trained to classify resumes as relevant or irrelevant.
- **Rule-Based Systems:** Systems that use predefined rules to filter and rank resumes based on specific criteria.

- **Application Tracking Systems (ATS):** While not models or techniques themselves, ATS software often incorporates some level of automated resume screening, typically based on keyword matching.

2.3 Highlight the gaps or limitations in existing solutions and how your project will address them.

Existing solutions often suffer from the following gaps and limitations:

- **Reliance on Keyword Matching:** Basic keyword matching fails to capture the semantic meaning of text, leading to inaccurate results. My project addresses this by using TF-IDF and cosine similarity, which provide a more nuanced measure of document similarity.
- **Limited Contextual Understanding:** Many systems struggle to understand the context of resumes and job descriptions. My project aims to address this by laying a foundation that can be extended to utilize more advanced NLP techniques like transformer models in future work.
- **Lack of User-Friendly Interfaces:** Some existing solutions are complex and require technical expertise to use. My project provides a user-friendly Streamlit web interface, making it accessible to a wider audience.
- **Handling Diverse Resume Formats:** Many systems have difficulty processing resumes in various formats. My project focuses on PDF processing using PyPDF2, but future work will expand to handle more formats.
- **Bias in Training Data:** Many systems are trained on biased data, leading to unfair or discriminatory results. Future iterations of this project will address bias mitigation.
- **Limited Skill Extraction:** Many systems do not extract skills correctly. In future iterations of this project, skill extraction will be improved.
- **Lack of Feedback Loops:** Many systems do not learn from their mistakes. Future iterations of this project will implement feedback loops.

CHAPTER 3

Proposed Methodology

3.1 System Design

System Architecture

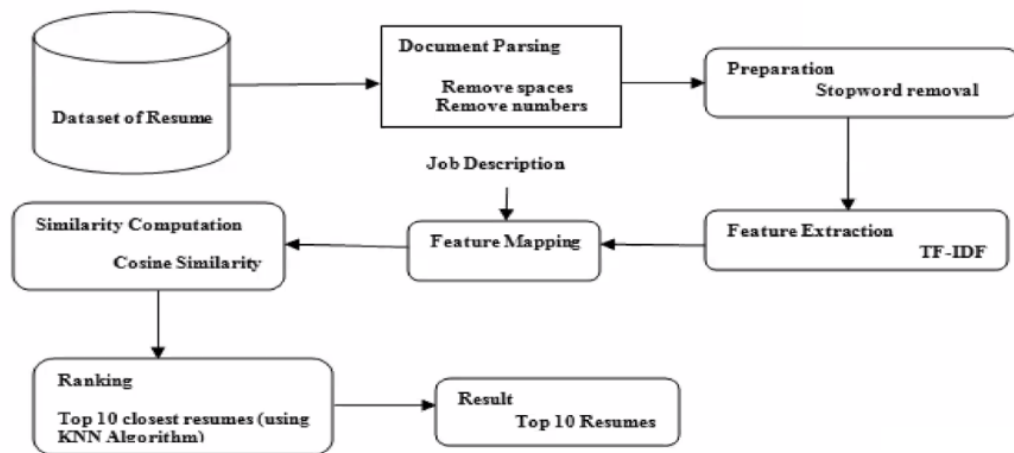


Figure 1

Diagram Overview:

The diagram illustrates the flow of data and processes within our system, starting with the input of resumes and job descriptions and culminating in the output of a ranked list of the top 10 resumes. It highlights the key stages of document parsing, preparation, feature extraction, similarity computation, and ranking.

Detailed Explanation of the Diagram:

Dataset of Resumes:

- This represents the collection of resumes stored in our system, likely in a database or file system.
- It serves as the initial input for the resume screening process.

Document Parsing :

- This stage involves extracting the text content from the resume documents (presumably PDFs, as mentioned in previous discussions).

- The diagram notes two specific parsing operations:
- **Remove Spaces:** Eliminating unnecessary whitespace.
- **Remove Numbers:** Removing numerical data, which might not be relevant for text-based analysis.

Preparation :

- This stage focuses on further cleaning and preparing the text data for analysis.
- **Stopword Removal:** Eliminating common words (e.g., "the," "is," "a") that don't contribute much to the meaning of the text.

Feature Extraction (Rectangle Box):

- This stage involves converting the text data into a numerical representation that can be used for similarity calculations.
- **TF-IDF (Term Frequency-Inverse Document Frequency):** This technique is used to assign weights to words based on their importance in the documents and the overall corpus.

Feature Mapping :

- This stage likely involves mapping the features extracted from the job description to the same feature space as the resumes, ensuring consistency for comparison.

Job Description : This represents the input of the job description text, which is used as the basis for comparing and ranking resumes.

Similarity Computation :

- This stage calculates the similarity between the job description and each resume based on their feature representations.
- **Cosine Similarity:** This specific method is used to measure the cosine of the angle between the vectors, representing the similarity of their directions.

Ranking :

- This stage ranks the resumes based on their similarity scores.
- **KNN Algorithm (K-Nearest Neighbors):** Although mentioned, KNN is typically used for classification or regression, not ranking. In this context, it may be used to identify the "nearest neighbors" (most similar resumes) based on similarity scores. However, a simple sorting algorithm would suffice for ranking.

Result :

- This is the final output of the system, presenting the top 10 closest resumes (most similar to the job description).

3.2 Requirement Specification

Tools and Technologies:

- **Programming Language:** Python
- **Web Framework:** Streamlit
- **PDF Processing:** PyPDF2
- **Data Manipulation:** pandas
- **Machine Learning/NLP:** scikit-learn

3.2.1 Hardware Requirements:

- A standard computer with sufficient processing power and memory to run Python and the required libraries.
- Sufficient storage space for storing resumes and application data.

3.2.2 Software Requirements:

- Python 3.6 or later.
- Streamlit library.
- PyPDF2 library.
- pandas library.

- scikit-learn library.
- pip (Python package installer).
- A web browser compatible with streamlit.
- A code editor or IDE.

CHAPTER 4

Implementation and Result

4.1 Snap Shots of Result:

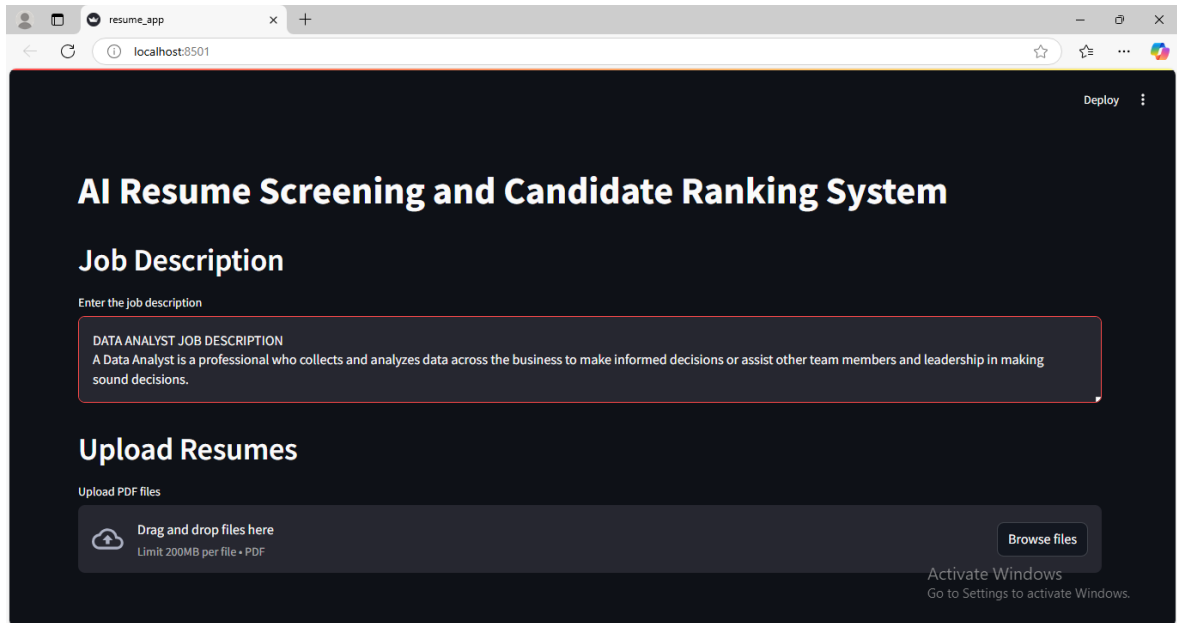


Figure 2

Entering the Job Description :

This snapshot shows the user interface of your AI-powered system, specifically the initial screen where users input a job description and upload resumes. The interface appears to be built using Streamlit, as indicated by the localhost:8501 URL.

Streamlit Application:

- The localhost:8501 URL indicates that this is a Streamlit application, a popular framework for building interactive web applications for machine learning projects.

Job Description Section:

- **"Enter the job description" Label:** Guides the user on what information to provide.
- **Text Area (with "DATA ANALYST JOB DESCRIPTION"):** This is where the user enters or pastes the job description. The example text provided is for a Data Analyst position. The red outline indicates that this element is currently selected or active.

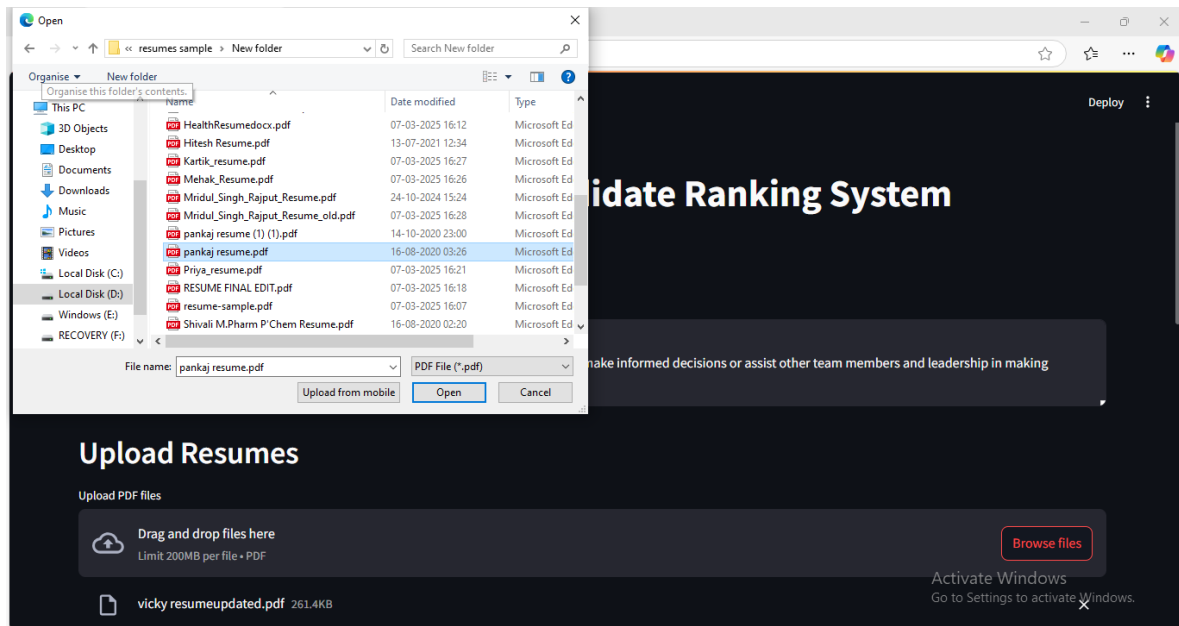


Figure 3

Selecting Resumes for Screening as per Job Description :

This snapshot displays the user interface of your AI Resume Screening and Ranking System, specifically the "Upload Resumes" section. The system is designed to allow users (likely recruiters or hiring managers) to upload PDF resume files for automated analysis and ranking.

Explanation of the Elements:

Drag and Drop Area:

- **Visual Cue:** The dotted outline and the "Drag and drop files here" text encourage users to drag and drop their resume files directly into this area.
- **File Size Limit:** "Limit 200MB per file PDF" informs users of the file size restriction and reiterates the PDF format requirement.

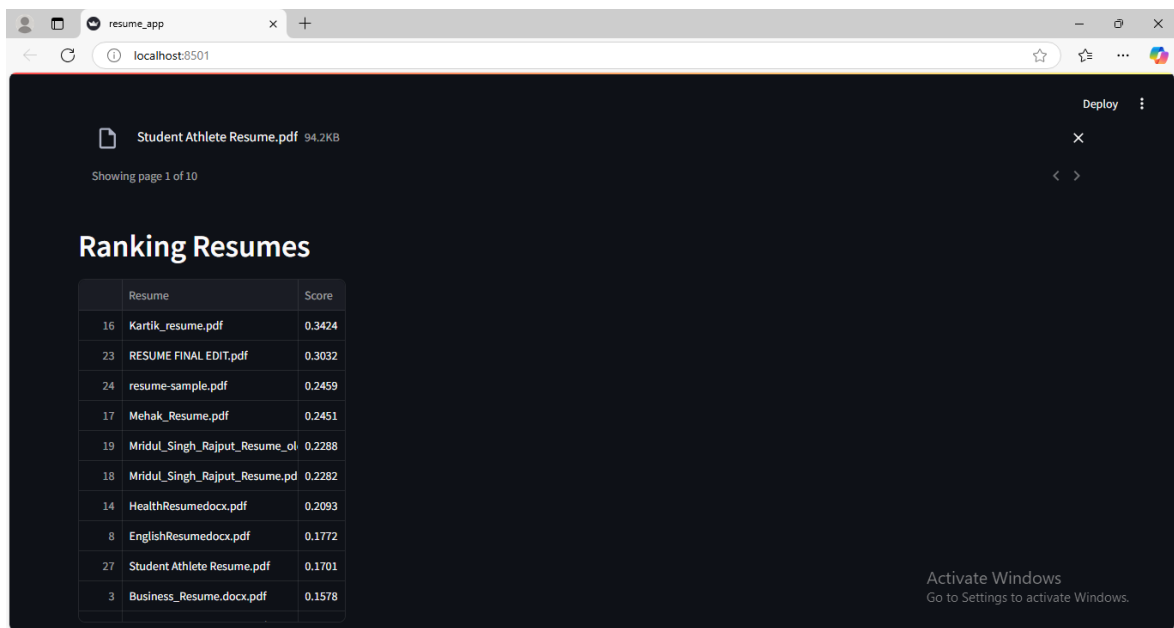


Figure 4

The Resumes Ranked on the basis of Job Description :

This snapshot shows the output of our system after processing the uploaded resumes against the provided job description. It displays a ranked list of resumes along with their corresponding similarity scores. The output of our AI Resume Screening and Candidate Ranking System., it demonstrates the system's core functionality of analyzing and ranking resumes, providing valuable insights to recruiters or hiring managers.

Ranking Resumes : This clearly indicates the purpose of this section of the interface.

Table: Resume and Score Columns:

- **Resume Column:** Lists the names of the uploaded resume files.
- **Score Column:** Displays the similarity score for each resume, indicating how well it matches the job description. The scores are presented as decimal values.

Ranked List of Resumes: The table shows the resumes ranked in descending order of their similarity scores. For Example, the top-ranked resume, "Kartik_resume.pdf" has the highest score of 0.3424.

Output of the AI System:

- It shows the ranked list of resumes, demonstrating the system's ability to analyze resumes and determine their relevance to the job description.

Ranking Algorithm Effectiveness:

- The scores and the order of the resumes reflect the effectiveness of your AI algorithm in identifying the most suitable candidates.

User Interface for Results:

- The table provides a clear and organized way for users to view the results.

4.2 GitHub Link for Code:

<https://github.com/Mridulsr/AI-Resume-Screening-and-Ranking-System-Project>

CHAPTER 5

Discussion and Conclusion

5.1 Future Work:

To further enhance the AI Resume Screening and Ranking System, several improvements can be considered.

Firstly, incorporating more advanced Natural Language Processing (NLP) techniques, such as Named Entity Recognition (NER) and semantic analysis, can improve the system's understanding of resume content and job requirements. This would enable the system to recognize specific skills, qualifications, and experience beyond simple keyword matching.

Secondly, integrating machine learning models for skill extraction and competency assessment can provide a more nuanced evaluation of candidate suitability. Implementing a feedback loop, where recruiters can provide input on the system's rankings, would allow for continuous model refinement and improved accuracy.

Additionally, expanding the system's capabilities to handle diverse resume formats beyond PDFs and integrating with applicant tracking system (ATS) would enhance its practicality. Exploring the use of contextual embeddings and transformer-based models can also improve the system's ability to understand the context of the job descriptions and resumes.

Finally, addressing potential biases in the training data and ensuring fairness in the ranking process should be a priority. Provide suggestions for improving the model or addressing any unresolved issues in future work.

5.2 Conclusion:

The AI Resume Screening and Ranking System project successfully developed an automated tool to streamline the initial stages of candidate selection. By leveraging NLP techniques and machine learning, the system effectively ranks resumes based on their relevance to job descriptions, significantly reducing the time and effort required for manual screening. This project demonstrates the potential of AI to enhance the efficiency and accuracy of the hiring process, enabling recruiters to quickly identify top candidates. The

system's implementation contributions to a more data-driven and objective approach to resume screening, ultimately leading to improved hiring outcomes. By automating the time consuming initial screening process, recruiters can focus on more in depth interviews and evaluations of the top candidates.

REFERENCES

- [1]. Gangoda, Nikethani, et al."Resume Ranking: AI-Based Skill Analysis and Skill Matching System." 2024 Sixth International Conference on Intelligent Computing in Data Sciences (ICDS), IEEE, 2024.
- [2]. Ganga, Varun Sai, et al. "Skill Extractor in Resumes using Machine Learning and Streamlit." 2024 1st International Conference on Trends in Engineering Systems and Technologies (ICTEST). IEEE, 2024.
- [3]. Pradeepa, R., et al. "Intelligent Resume Evaluation Tool Based on Machine Learning for Analysis And Career Advancement." 2024 International Conference on Emerging Research in Computational Science (ICERCS). IEEE, 2024.
- [4]. Mariappan, Prabha, et al. "Smart Resume Screening and Job Represents Recommendation System." Available at SSRN 5141402.
- [5]. Wooi, Zhuang Ru. Job-applicant matchmaking system using natural language processing. Diss. UTAR, 2023.