ALY 6080: INTEGRATED EXPERIENTIAL LEARNING

Spring 2023: College of Professional Studies,

Northeastern University

Individual Project Proposal Draft: Resume Classification using Domain adapation

Submitted By:

Siddharth Alashi

NUID: 002728528

alashi.s@northeastern.edu

Submitted To

Prof. Herath Gedara, Chinthaka Pathum Dinesh

05/22/2023

Individual Project Proposal Draft: Resume Classification using Domain Adaptation

Siddharth Alashi || 002728528 || alashi.s@northeastern.edu
Northeastern University, Vancouver, Canada

I. Statement of Purpose

This paper will summarize all the 3 bibliographies written and also propose my intentions of Job Recommendation system using Artificial Intelligence. In the first annoted bibliography inspired by the work in the paper "*Automated tool for Resume Classifier*" which, the paper explores the creation and execution of a resume classifier app that uses an ensemble learning approach with a voting classifier. This app aims to classify a candidate's profile into a relevant field based on their mentioned interests, work experience, and expertise. The 2nd paper "*Domain Adaptation for Resume Classification Using Convolutional Neural Networks*" The paper proposes a noel method for classifying the resume data of Job applicants into 27 different job categories. The 3rd paper "*A Survey of Job recommendation system*" concludes that The objective of recommender system technology is to assist users in discovering items that align with their individual interests. This article will provide an overview of the e-recruiting process and various existing approaches for developing personalized recommender systems specifically tailored for candidates and job matching. The **overall goal** is to assist users in finding job opportunities that align with their individual interests.

II. Scope of the project

Approach: The resume classifier application comprises two key modules: **the Natural Language Processing Pipeline (NLPP) and the Classification module**. HR team inputs new recruit profiles into the NLPP, which eliminates unnecessary information and extracts relevant data as tokens for the classification module. This module analyzes the token list to classify resumes into suitable domains. The application generates a graph showing candidate relevance to different domains based on their interests, work experience, and expertise. If there are no available jobs in the candidate's most suitable domain, the HR team can consider assigning them a project in an alternate domain of interest. This automation streamlines project allocation, eliminating the manual and time-consuming process of manually reviewing resumes. In this extended paper version, a new classifier (K-nearest neighbors) has been added to the ensemble-based voting classifier, improving application efficiency. The updated learning model achieves better accuracy in resume classification and supports additional domains not covered in previous work.

The new recruits projects based on the content present in their respective resumes. The data in the resumes of the candidates are subjected to a NLPP in order to obtain only necessary and relevant details.

Tokenisation, Stop word deletion, Parts of Speech Tagging, Named Entity Recognition In the current stage of the pipeline, certain tokens are removed by identifying candidate names, organization names, and place name tokens. These tokens are deemed trivial and do not significantly impact the project allocation process for the candidate. This represents the final step of the pre-processing pipeline. The remaining tokens resulting from this step are then considered by the classifier for the purpose of classification.

Opportunity to be Solved: Overall, focus is placed on developing a dynamic machine learning automation tool that is not simply dependent on training data when assigning tasks to new recruits. We added to our earlier efforts. A learning model that can categorise resumes with greater accuracy and support more new domains.

III. Scope of a Project

The scope of work for a job recommendation system project may vary depending on the particular demands and objectives. However, the following are some common actions and deliverables that may help the project fulfil its objectives:

1. Data Collection: Create a massive dataset of job listings, complete with job titles, descriptions, qualifications, and other relevant information. This can be performed through scraping employment portals, collaborating with job boards, or using publically accessible data sources.

2. Data Preprocessing: Clean and preprocess the obtained data to remove duplicates, address missing values, standardise formats, and ensure consistency. Text normalisation, tokenization, stemming, and other natural language processing techniques may be employed to prepare the data for further analysis at this step.

3.Feature Engineering: Take meaningful features from job listings and use them to represent various aspects of the jobs. This could include things like job title, required skills, industry, location, salary range, and so on. These characteristics will be critical in developing an effective recommendation system.

4. User Profiling: Create mechanisms for gathering information about user preferences and profiles. This may entail directly collecting information from users (via surveys or questionnaires) or implicitly based on their interaction with the system (via job searches, clicks, and application history). User profiling aids in the personalization of recommendations based on individual preferences.

5. Recommendation Algorithms: Create and test different recommendation algorithms to generate job recommendations. Techniques such as collaborative filtering, content-based filtering, hybrid approaches, and more advanced methods such as matrix factorization or deep learning models are examples of this. The algorithms chosen will be influenced by the specific requirements and data available.

6. Evaluation Metrics: Define appropriate evaluation metrics to assess the performance of the recommendation system. Common metrics include precision, recall, F1-score, mean average precision, and user satisfaction. These metrics will aid in determining the effectiveness and efficiency of the system.

7. User Interface (UI): Design a user-friendly interface for users to interact with the recommendation system. Users should be able to enter their preferences, view recommended jobs, provide feedback, and track the status of their application through the user interface. The UI should also include filtering and sorting options to improve the user experience.

8. Testing and Validation: Thoroughly test and validate the recommendation system to ensure its accuracy, robustness, and scalability. This includes both individual component unit testing and system-wide testing. Validate the results with a holdout dataset or user feedback, and iterate the system as necessary.ance.

Key Deliverables:

1. Dataset of job listings (cleaned and preprocessed)

2. Feature representation of job listings

3. User profiling mechanism

4. Implemented recommendation algorithms

5. Evaluation metrics and evaluation results

6. User interface design and development

7. Tested and validated recommendation system

8. Comprehensive system documentation

9. Deployed job recommendation system

10. Maintenance and monitoring plan

These deliverables collectively support the development and deployment of a job recommendation system, enabling personalized job recommendations to users based on their preferences and job-related information.

In-detail: 1. Stop – word deletation : The resume of the candidate is filled using “is”, “are”, “etc”, The removal of such words will help in better classification of the candidates profile. The elimination is critical because including stop words in the training set would result in false learning by the classifier, limiting classification efficiency.

2. **Parts of Speech Tagging**: In NLPP (Natural Language Processing Pipeline), the subsequent step after eliminating stop words is POS tagging. This involves assigning Part

of Speech tags to each token. English has 8 distinct parts of speech, including Verb, Noun, Adjective, Adverb, Pronoun, Preposition, Conjunction, and Interjection. In the context of resume analysis, the tools, technologies, and projects mentioned are categorized as either nouns or pronouns. Since the domain allocation in the resume depends on these features, only the tokens labeled as nouns and pronouns (NNP) are considered for further processing in NLPP. This reduction in the number of words being classified significantly reduces computation. Figure 6 illustrates the POS tagging of the tokens, with only the nouns progressing to the subsequent stage in the pipeline.

3. **Named Entity Recognition**: is applied to the tokens labeled as nouns and pronouns. This process identifies candidate names, educational institutions, and place names. In the current stage of the pipeline, these tokens are eliminated because they hold little importance in determining project allocation to the candidate. This represents the final step of the preprocessing pipeline. The tokens that remain after this step are considered by the classifier for classification. Tokens recognized as names, places, or organizations are removed, as depicted. Only the contents displayed in figure 8 proceed to the classification module.

IV. Background Research

Advancements in AI and ML have resulted in significant advancements in semantic analysis and text categorization . The authors in have conducted a comparative analysis of various ML techniques for text classification and categorization. However, their approach has a limitation: the categories for document classification must be predefined during the training phase. Consequently, the learning model can only classify resumes within the initially trained domains. V Ram and Prasanna have advocated for neural networks in text categorization , emphasizing that sufficient data input to these networks leads to accurate category predictions. Yieng Huang and Jingdeng Chen have proposed the use of deep neural networks in text classification , which outperform traditional single-layer neural networks due to their multiple layers of neurons. Nevertheless, all these approaches heavily rely on training data for classification. In contrast, our approach aims to overcome this dependency on the training set, enabling effective resume classification into relevant domains and subsequent project allocation for candidates.

V. Data Collection

Project Proposal: Automated Tool for Resume Classification Using Semantic Analysis

Design and Data Collection Methods:

To accomplish the goal of automated resume classification using semantic analysis, the following data and data collection methods can be utilized:

1. Resume Data: Collect a diverse dataset of resumes from various sources, such as job portals, career websites, recruitment agencies, or publicly available resume databases. The resumes should cover a wide range of industries, job positions, and skill sets.

2. Job Description Data: Gather a comprehensive set of job descriptions from different industries and domains. These job descriptions will serve as the reference for classifying resumes into relevant categories.

3. Training Data: Manually label a subset of the collected resumes with their corresponding categories based on the job descriptions. This labeled data will be used as the training set for building the classification model.

4. Exploratory Data Analysis: Perform initial data exploration to understand the characteristics of the resume dataset. Analyze factors such as resume length, sections (education, experience, skills, etc.), formatting styles, and other relevant features that may impact the classification process.

5. Natural Language Processing (NLP) Techniques: Apply NLP techniques like tokenization, stemming, and part-of-speech tagging to preprocess the resume data. These techniques will help extract meaningful information and features from the resumes.

6. Feature Extraction: Extract relevant features from the preprocessed resumes, such as job titles, skills, education details, work experience, certifications, and other relevant information. These features will serve as inputs for the classification model.

7. Model Training and Evaluation: Utilize machine learning or deep learning algorithms to train a classification model on the labeled training data. Evaluate the model's performance using appropriate evaluation metrics such as accuracy, precision, recall, and F1-score. Adjust and fine-tune the model parameters as necessary.

8. Cross-Validation and Testing: Perform cross-validation to assess the generalizability of the model and mitigate overfitting. Apply the trained model to classify unseen resumes from the dataset and evaluate its performance on the test set.

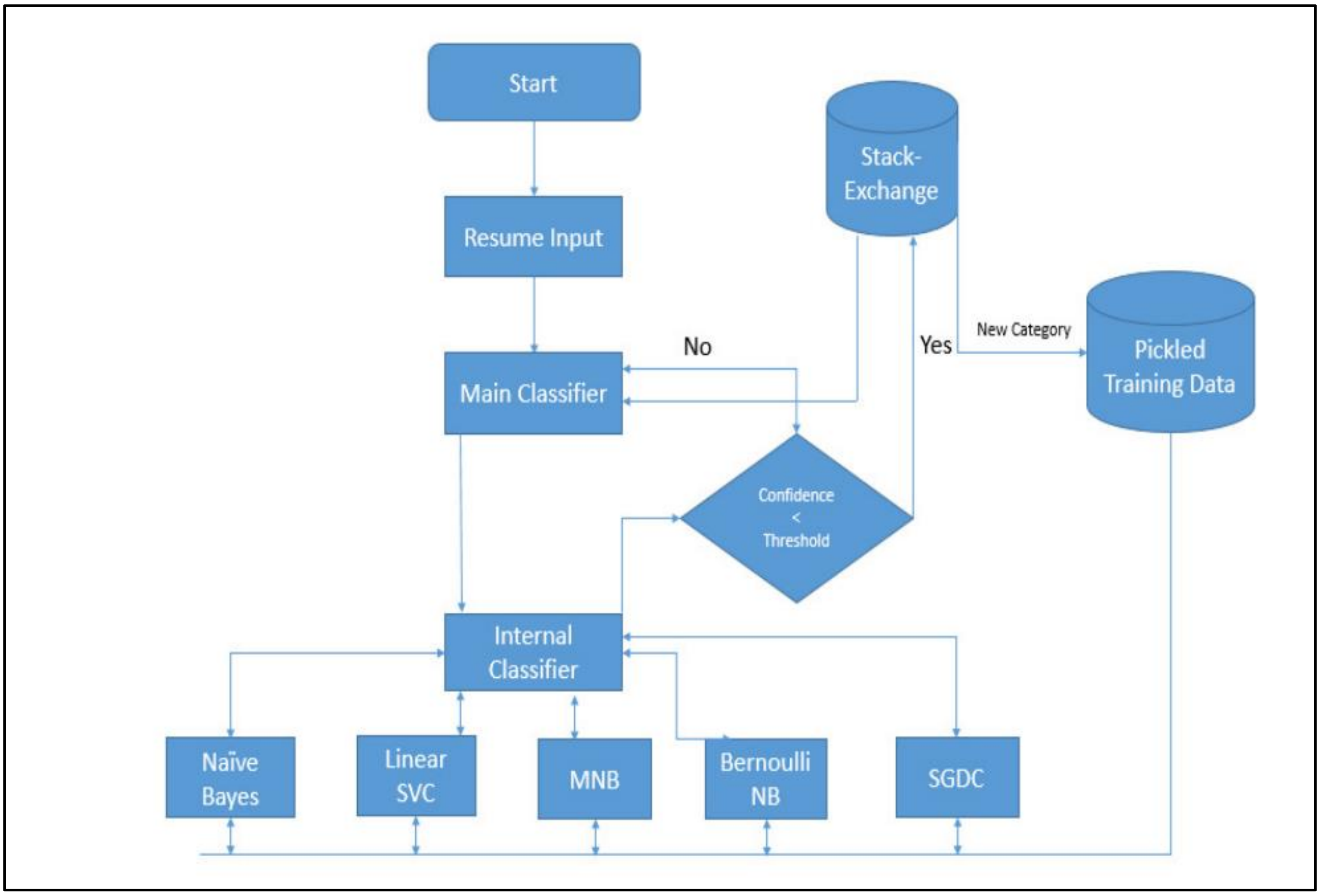
9. Iterative Refinement: Iterate on the model and feature engineering process based on the insights gained from the analysis and evaluation results. Continuously refine the model to improve its accuracy and robustness.

10. Documentation: Document the entire process, including data collection methods, preprocessing steps, model architecture, evaluation metrics, and results. This documentation will provide a reference for future updates, maintenance, and further enhancements of the automated resume classification tool.

By following these design and data collection methods, the project aims to develop an automated tool for resume classification using semantic analysis, providing a scalable and efficient solution for sorting and categorizing resumes based on job requirements and domains.

VI. Implementation and Methodology

After the NLPP module generates tokens from the text content in the resume, these tokens are passed to the Classification module for domain classification and project allocation. Figure 9 presents the flow chart of the Classification module, which employs an ensemble learning-based voting classifier consisting of five individual classifiers: i) Naive Bayes Multinomial Naive Bayes, iii) Linear SVC, iv) Bernoulli Naive Bayes, and v) Logistic Regression.



The classifiers will help us know the least efficiency on the training set. This ensures that the classifier with greater efficiency in categorising the tokens to a domain has a greater influence than the classifier with less efficiency in mapping the profile to a domain.

Bar Graphs: The output of the resume classifier application is represented by a bar graph that displays the candidate's profile suitability for different domains. The x-axis shows the list of domains, while the y-axis represents the model's confidence in classifying the candidate's profile into a specific domain. A higher value on the y-axis indicates a greater suitability of the candidate for the corresponding domain.

The resume classifier application assigns a confidence score and categorises the resume into the AI domain. This categorization is based on the recognition of deep learning tools like "tensorflow," "keras," and "theano," as well as machine learning buzzwords like "spam filter," "analysis and classification," and "fraud detection" that the application's resume classifier tool has identified.

The project proposal aims at recommendation system technology is to aid users in finding items that match their personal interests. This article aims to give a summary of the e-recruiting process and different approaches currently available for creating personalized recommender systems that are specifically designed for candidates and job matching. The ultimate objective is to support users in discovering job opportunities that are well-suited to their individual interests. FastText and the CON method yielded the highest F1 score in the semantic relatedness task. However, when the same model was trained on the 2ch corpus, it did not perform as well in the task of semantic similarity. This suggests that the effectiveness of a word embedding model is not solely determined by its internal algorithm, but also by the vocabulary used in the chosen corpus. Ideally, a model that better captures the human perception of semantic relatedness within the specific context of a given vocabulary would yield the best results on that corpus. In the future, we plan to expand our comparison to web forums in other languages and propose a typological analysis of semantic shifts in web slang across different cultures, supported by word representations from various languages.

VII. Several Test and Analysis

With the data generated by an AI model that prepares cover letters and resumes, several types of analysis can be conducted. Here are some potential approaches:

1. Statistical Tests: You can perform various statistical tests to gain insights from the data. For example, you could use hypothesis testing to assess the effectiveness of the AI model in generating high-quality cover letters and resumes. This could involve comparing the model's output with human-generated cover letters and resumes using appropriate statistical tests.

2. Pattern Searching: The data can be analyzed to identify patterns or trends. This could involve exploring common phrases, formatting styles, or language preferences within the cover letters and resumes. By identifying such patterns, you can gain insights into what works well and what doesn't, and use that information to improve the AI model's performance.

3. Regression Models: Regression analysis can be applied to understand the relationship between different variables in the cover letters and resumes. For example, you could investigate how certain characteristics (such as the use of specific keywords or the inclusion of particular experiences) correlate with the likelihood of an applicant getting an interview or a job offer. This analysis can help fine-tune the AI model's recommendations.

4. Forecasting or Time Series Models: If you have a substantial amount of historical data, you can utilize forecasting or time series models to predict future trends. This could involve predicting changes in the demand for specific skills or industry preferences over time. By

forecasting such trends, you can improve the AI model's ability to tailor cover letters and resumes to meet evolving market demands.

Additionally, you can combine different types of analysis to gain a comprehensive understanding of the data. For example, you can use statistical tests to evaluate the significance of observed patterns, and then incorporate regression or forecasting models to make predictions or derive insights for future improvements. The choice of analysis methods depends on the specific research objectives and the available data.

Preference

1. Sayfullina, L., Malmi, E., Liao, Y., & Jung, A. (2017). Domain Adaptation for Resume Classification Using Convolutional Neural Networks. In Lecture Notes in Computer Science (pp. 82–93). Springer Science+Business Media. https://doi.org/10.1007/978-3-319-73013-4_
2. Gopalakrishna, S. T. (2019, January 1). Automated Tool for Resume Classification Using Sementic Analysis. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3349094

Al-Otaibi, S.T., Ykhlef, M.: A survey of job recommender systems. Int. J. Phys. Sci. 7(29), 5127–5142 (2012