Resume Parser using hybrid approach to enhance the efficiency of Automated Recruitment Processes

Nirmiti Bhoir[1], Mrunmayee Jakate[1], Snehal Lavangare[1], Aarushi Das[1], and Sujata Kolhe[1]

[1]Datta Meghe College of Engineering

April 17, 2023

Abstract

This study provides a novel resume parsing solution using a hybrid Spacy Transformer BERT and Spacy NLP methodology. The main goal is to create a resume parser that can efficiently extract pertinent data from unstructured resumes that do not adhere to a predetermined resume structure and may contain information presented in a non-standardized manner. We also intend to investigate the usage of video resumes as a fresh source of candidate data and put forth a cutting-edge method for video resume parsing that combines visual and audio processing methods. We employed a hybrid methodology of Spacy Transformer BERT and Spacy NLP to accomplish these goals. A pre-trained deep learning model called Spacy Transformer BERT captures the text's semantic meaning, and Spacy NLP employs natural language processing to glean pertinent information from it. Our method combines the strengths of the two models for high accuracy and efficiency in collecting pertinent information from resumes. Using a dataset of resumes, we ran experiments to gauge how well our suggested system performed. The outcomes demonstrate that our system was highly accurate in retrieving pertinent data, including candidate names, contact information, qualifications, work experience, and other pertinent characteristics.

Resume Parser using hybrid approach to enhance the efficiency of Automated Recruitment Processes

Mrunmayee Jakate[1], Snehal Lavangare[2], Nirmiti Bhoir[3], Aarushi Das[4],

Dr. Sujata R. Kolhe[5]

[1,2,3,4]BE Student, Information Technology, Datta Meghe College of Engineering

Airoli, India

[5]Professor , Information Technology, Datta Meghe College of Engineering

Airoli, India

Mrunmayee Jakate, Snehal Lavangare

Princeton Hiranandani Estate Patlipada, Licodia House Wagle estate Thane(W) India E-mail: mrunmayeej99@gmail.com, snehallavangare19@gmail.com

Nirmiti Bhoir, Aarushi Das Eco Winds Lake Road Mumbai, Runwal Tower Runwal Nagar Kolbad Thane(W) India

E-mail: nirmitibhoir.mumbai@ieee.org, aarushidas.mumbai@gmail.com

Dr. Sujata Kolhe

Airoli Navi Mumbai India

E-mail: sujata.kolhe@dmce.ac.in

Keywords: resume parsing, hybrid methodology, spacy transformer, natural language processing, bert, recruitment

Abstract:

This study provides a novel resume parsing solution using a hybrid Spacy Transformer BERT and Spacy NLP methodology. The main goal is to create a resume parser that can efficiently extract pertinent data from unstructured resumes that do not adhere to a predetermined resume structure and may contain information presented in a non-standardized manner. We also intend to investigate the usage of video resumes as a fresh source of candidate data and put forth a cutting-edge method for video resume parsing that combines visual and audio processing methods. We employed a hybrid methodology of Spacy Transformer BERT and Spacy NLP to accomplish these goals. A pre-trained deep learning model called Spacy Transformer BERT captures the text's semantic meaning, and Spacy NLP employs natural language processing to glean pertinent information from it. Our method combines the strengths of the two models for high accuracy and efficiency in collecting pertinent information from resumes. Using a dataset of resumes, we ran experiments to gauge how well our suggested system performed. The outcomes demonstrate that our system was highly accurate in retrieving pertinent data, including candidate names, contact information, qualifications, work experience, and other pertinent characteristics.

1. Introduction

A resume parser is a piece of automated software that gathers data from resumes or CVs (curriculum vitae) and arranges it in a way that makes it simple for hiring managers, recruiters, and applicant tracking systems to search and evaluate it (ATS). The parser recognises and extracts pertinent information, such as contact details, employment history, education, skills, and other qualifications, using natural language processing (NLP) and machine learning methods. Recruitment firms and HR departments receive a mountain of resumes for each position opening in today's competitive employment market. The manual screening of these resumes takes a lot of time and is prone to mistakes.[2]To overcome this difficulty, automating the resume screening process through resume parsing has shown to be a successful option. In resume parsing, pertinent data is extracted from a resume and organised into a predetermined manner. The candidate's name, contact information, qualifications, job history, and other pertinent information may all be included in the information that was extracted. The time and effort needed for manual screening can be greatly decreased, and the accuracy and consistency of the screening process can be increased, with the help of a well-designed resume parser. For resume parsing, a number of methods have been suggested, including rule-based systems, machine learning models, and methods of natural language processing. Yet, more precise and effective resume parsers have been created as a result of recent developments in deep learning and natural language processing. In this study, we suggest a fresh solution for resume parsing by combining Spacy Transformer BERT with Spacy NLP.

The strengths of both models are used in our suggested technique to extract pertinent information from resumes with great accuracy and efficiency. The Spacy Transformer BERT is a pre-trained transformer model that exhibits outstanding performance in a variety of named entity recognition, sentence classification, and text summarization tasks.[12]

For accurate information extraction, the model must accurately capture the semantic meaning of the resume content. Spacy NLP is a potent tool for dealing with complex syntactic structures and finding the pertinent entities and characteristics in the resume content. [5] To determine which fields on the resume are important, the NLP pipeline performs entity extraction and attribute extraction. The experimental results show that our recommended method outperforms current state-of-the-art resume parsers in terms of accuracy and efficiency. Recruiting agencies can speed up the hiring process by using the automated resume screening method provided by the suggested system.

We looked into the use of video resumes in addition to conventional resumes as a fresh source of applicant

data. While they enable candidates to demonstrate their abilities and personalities in ways that a standard CV cannot, video resumes are growing in popularity. Yet, because video content is unstructured, video resumption provides additional difficulties for information extraction.

2. Literature Review

In 2022, Bhushan Kinge, Shrinivas Mandhare, Pranali Chavan and S. M. Chaware conducted a research on Resume Screening Using Machine Learning and NLP : A Proposed System.[1] The approaches discussed in this paper use a variety of machine learning and neural network models, including SVM, KNN, Word2Vec, cosine similarity, etc. to detect, identify, and categorize different resumes. The models' levels of accuracy range from 78% to 98% depending on the datasets utilized, the difficulty of the learning processes, and the size of the dataset.

In 2022, the authors Tejaswini K , Umadevi V , Shashank M Kadiwal and Sanjay Revanna published a paper on Design and development of machine learning based resume ranking system.[2] The team can have trouble finding the right person at the right time if resume screening is done manually. An automated method for screening and ranking applicants may substantially help with the arduous screening. According to the authors' explanation in their work, the best candidates can be identified using content-based suggestion, which employs cosine similarity to identify the curriculum vitae that are most similar to the provided job description and KNN algorithm to select and rank Curriculum Vitae (CV) based on job descriptions in large quantities. The suggested system performs according to experimental findings, with an average text parsing accuracy of 85% and a ranking accuracy of 92%.

In 2021 Vukadin, Davor, et al. published a research paper on "Information extraction from free-form CV documents in multiple languages." [3] They have used BERT transformer and constructed two multilingual models for obtaining useful information from free-form CVs, and both models were tested on annotated datasets. For two problems they have introduced two models, the first model used to extract "hard" information such as names and previous employment organizations and to classify individual sections of the CV to contextualize the extracted information. In the second model, self-assessed skill competence degrees were detected.

In 2021, Agnieszka Wosiak researched about Automated extraction of information from Polish resume documents in the IT recruitment process. [4] This paper describes about the hybrid solution for the automated extraction of information from Polish resume documents in the IT recruitment process.The anomaly detection in texts and handling outliers in resumes are not done,does not include the outliers in subsequent phases of the analysis

In 2020 Suresh, Yeresime, and A. Manusha Reddy published their research manuscript A contextual model for information extraction in resume analytics using NLP's spacy.[5] 250 resumes dataset of different file formats and structures are used by the authors to perform Named Entity Recognition. The authors have used Spacy: an open source Natural Language Processing library to perform NER. Phrase Matcher feature has been used to annotate the entity. At present looking at the results of this paper it works only on the skills of a candidate. A dictionary of skills required for each job role is created and matched with the document with the help of phrase matchers.

In 2020 Bodhvi Gaur, Gurpreet Singh Saluja, Hamsa Bharathi Sivakumar and Sanjay Singh published their research work titled Semi-supervised deep learning based named entity recognition model to parse education section of resumes.[6]The authors have utilized a resume database of 550 resumes divided as 500 for training and 50 for testing. BILOU encoding scheme is used to train the model. CNN Layer and Bi-LSTM layers are used to train the model. The authors have used this proposed model to extract only the education section of the resumes which includes the degree and institute name.

In 2020, Pradeep Kumar Roya , Sarabjeet Singh Chowdhary and Rocky Bhatia issued a paper on, A Machine Learning approach for automation of Resume Recommendation system.[7] Today, finding the right talent while spending little money and effort online is the biggest challenge facing the industry. In order

to streamline the entire hiring process, there are some significant obstacles that must be overcome. By automating the process, the authors hope to address the aforementioned issues. The system would assist in locating the appropriate CV among the vast quantities of CVs; it would be unconcerned with the format in which the CV was prepared and would provide a list of CVs that best matched the job description provided by the recruiter. The classification process is proposed using several different strategies.

In 2020, Nirali Bhaliya, Jay Gandhi, Dheeraj Kumar Singh published a research on the topic - NLP based Extraction of Relevant Resume using Machine Learning.[8]There are no set rules for how to write a CV / Resume, thus even though the statistics codecs that are used aren't always completely unstructured, it's still difficult to convert them into a structured format. The CV parser supports multiple languages, has easy customization options, development sheets, determination agents, and semantic mapping for limits. The outcomes of parsing with the leasing limit are accurate costs. When mentioning resumes in relation to its types and codecs, its age increases. Its coordination helps users get an API key for collaborative projects. The parser employs two or three rules that instruct the call and address. The CV parser system is used by Scout bundles to identify resumes. Since resumes are organized in such a creative way and contain a variety of real aspects, such as structured and unstructured estimations, meta experiences, etc. The component extraction method from the moved CVs is provided by the proposed CV parser approach.

In 2020 Gunaseelan, B., Supriya Mandal, and V. Rajagopalan published their research work titled Automatic Extraction of Segments from Resumes using Machine Learning.[9] They have utilized a resume dataset of 400 resumes of different file formats like html,pdf,doc,docx,txt. Pydocx and Pdfminer packages have been used to extract text from the files. The authors have used a machine learning approach to classify the headings of a resume. Headings in a resume have been classified into a dictionary consisting of a key : which corresponds to the heading for eg- Skills, Objective, Experience etc and content: which consists of the resume content under that heading. Approximately 26092 lines of text from the resumes are labeled. Two target classes are used: '1' for a heading and '0' for not a heading. Secondly the authors have utilized fuzzy string matching to extract the skillsets of the candidate after heading segmentation. Currently only the skill set category is explored by the authors. XG Boost gave maximum accuracy of approximately 90%.

In 2019, Sujit Amin , Nikita Jayakar , Sonia Sunny , Pheba Babu , M.Kiruthika and Ambarish Gurjar, published a research on Web Application for Screening Resumes.[10] According to the work criteria of a recruiter, the authors suggested a strategy for screening and rating resumes. Job Applicant Side, Server Side, and Recruiter Side are the three modules that make up this system. A pipeline for NLP is used to train the model. To accomplish great accuracy, they used a semi-supervised methodology. The scoring of a resume is determined by a module on the recruiter's end. The candidate who received the highest score will be ranked above the contestant who received the lowest score.

In 2019 Tikhonova, Maria, and Anastasia Gavrishchuk published a research paper on "NLP methods for automatic candidate's cv segmentation."[11] The paper focuses on the automatic CV parsing and segmentation approach. Entities such as work experience, skills, education, additional education are extracted using grid search. The 50 resumes were selected for testing purpose and evaluation was done using Intersection over Union.

In 2019 Bhatia, Vedant, et al. published a research paper on "End-to-end resume parsing and finding candidates for a job description using bert."[12] Information extraction from resumes using two tools to convert the PDF document to text like pdftohtml and Apache Tika. Building a Parser for a Standard Format of Resumes by Converting Resumes to LinkedIn Format, Web Application for Resume Parsing it results in uniform and efficient data collection for the employers. Ranking candidates on the basis of job-description suitability.

In 2018, the authors Abeer Zaroor, Mohammed Maree, and Muath Sabha published a research on A Hybrid Approach to Conceptual Classification and Ranking of Resumes and Their Corresponding Job Posts.[13] The proposed approach consists of a segmentation module where automatic segmentation of various sections of resumes like Education, Experience, Company Name is performed using Natural Language Processing

techniques and regex patterns. A comparative study of DICE v/s ONet is performed which is used to classify the skills of the candidate into a particular job category like software development/frontend/arts etc. Each skill given as an input gives a job role as output along with a weightage in terms of percentage. Therefore the approach is focussed more on automatic matching of resumes to a particular job role.

In 2018 Ayishathahira, C. H., C. Sreejith, and C. Raseek performed research on Combination of neural networks and conditional random fields for efficient resume parsing.[14] Four major steps are employed by the authors which include Extraction of text from files,Text preprocessing and heading segmentation. Used deep learning models like CNN to classify different segments of resume and CRF for labeling entities, their system classified resumes into 3 segments and extracted 23 fields.CNN is used to segment documents into 4 classes - personal, educational, occupational,and others. Total 1200 resumes are used for training the model. The authors achieved accuracy of 91% with CNN model and 43% with the BI-LSTM model.

In 2018 Mohamed, Ashif, et al. performed research on "Smart talents recruiter-resume ranking and recommendation system".[15] They have built three modules namely information extraction, Candidate search, Candidate Ranking Algorithms. Inside candidate ranking they've used Educational Qualifications ranking algorithm, Skills and Work Experience ranking algorithm. Educational qualifications are matched by algorithm I, while skills and professional experience are matched by algorithm II, and the relative score is calculated.

In 2017, Satyaki Sanyal, Souvik Hazra, Soumyashree Adhikary, Neelanjan Ghosh published a paper on Resume Parser with Natural Language Processing.[16] In this paper it tells about how Converting different formats of resumes to text and parsing relevant information using syntactic analysis and semantic analysis is done. Ranking the resume and analyzing information about the candidate from social networking sites like Facebook and Twitter was included.

In 2015, Divyanshu Chandola, Aditya Garg2, Ankit Maurya, Amit Kushwaha researched about Online Resume Parsing System Using Text Analytics. [17] The papers describes about the collection of resumes, searching for keywords stored in knowledge base in the resume text, fetching new keywords from the resumes to build the knowledge base further, ranking and Categorization of candidate based on a rating score. Though the trustworthiness of a resume to shortlist a candidate but since this will not be the final procedure of any company's recruitment process.

In 2015, Swapnil Sonar, Bhagwan Bankar issued a research paper on Resume Parsing with Named Entity Clustering Algorithm.[18] This paper explains about Text Segmentation, Named Entity Recognition, Named Entity Clustering and Text Normalization.Text Segmentation will separate out into segments named as contact information, education information, professional details and personal information segment. In Named Entity Recognition the tokenized text documents are fed to a Named Entity Recognizer. The named entity is in the block of information that needs to be grouped together to do the more process on it. In normalization, expanding some abbreviations using dictionaries is done. Though this method has disadvantages: Complete system is dependent on the web, computer and modern facilities like the internet so this is not useful in rural areas but nowadays with increasing globalization this disadvantage is easily minimized.

3. Proposed Methodology

There are multiple approaches to perform Named Entity Recognition on a piece of text. Natural language processing (NLP) has a subtask called named entity recognition (NER) that deals with locating and extracting particular pieces of data from unstructured text. In particular, it refers to the process of identifying and classifying named entities in text into predefined categories such as person names, organizations, locations, dates, times, monetary values, and others.[18] In our use case we have to identify custom entities inside a resume which include : Name, College Name, Degree, Graduation Year, Years of Experience, Companies worked at, Designation, Skills, Location and Email address.

The extraction of these entities will ease the task of the recruiters and fasten the process of reviewing the resumes. The practical applications of NER are text classification, machine translation etc. It is commonly

used in various fields, such as finance, legal, news, and social media analysis, to extract useful information and insights from large volumes of unstructured text data.

Some of the most commonly used methods include Natural Language Processing, Deep Learning and Machine Learning. Deep Learning and Machine Learning techniques need a predefined dataset for training purposes, where we include resumes of various types and file formats and perform supervised learning. This method automates the prediction process and is not dependent on any predefined list of words. Whereas the NLP process works with a knowledge base. Hence to ensure a good performance our web application proposes a hybrid methodology including the best of both techniques ensuring an optimum output.

3.1 NLP Approach

The process of converting human input languages into software comprehension is known as natural language processing, which is driven by artificial intelligence.[16] The conventional method for solving NLP consists of the following sequential flow of important steps:

3.1.1 Preprocessing of Data

In order to extract the text from various file formats such as pdf and docs, we converted them into a text file. Python libraries pdfplumber and doc2txt were used to convert unstructured resumes to a structured format. Since many people follow their own unique style when it comes to presenting qualifications in resumes, there are numerous ways to represent qualifications, experience and other information to be extracted. It's not easy to convert unstructured resumes into structured format. Tokenization is the first step in preprocessing, which in this case is the partitioning of raw data into small chunks of text, along with removing punctuation marks. The resume text is split into lines, empty lines are removed along with the removal of stopwords and punctuation marks if encountered in any line.

```
#Function to preprocess the resume text
from nltk.tokenize import word_tokenize
punc = '''’·••.!()-[]{};:'"\,<>./?#$%^&*_~'''
def preprocess_text(resume_text):
  res = re.sub(' +', ' ', resume_text)
  lines = res.split("\n")
  non_empty_lines = [line.lower() for line in lines if line.strip() != ""]
  string_without_empty_lines = ""
  for line in non_empty_lines:
        word_tokens = word_tokenize(line)
        filtered_line = [w for w in word_tokens if w not in punc]
        new_line=" ".join(filtered_line)
        string_without_empty_lines += new_line + "\n"
  return string_without_empty_lines
```

Fig 1. Preprocessing Operations

3.1.2 Segmentation of resume

Segmentation is another essential step in NLP which involves partitioning the text into linguistically meaningful units - at the lowest level characters representing the individual graphic symbol. It is challenging to extract segments from the unstructured resumes and screening them in an appropriate way. Segmenting resume topics has the additional benefit of making it easier to explore and analyze information in resumes, as well as compare necessary information between resumes.[9]Searching within a specified topic is important and efficient instead of searching the entire document. For example, in order to extract the technical skills of a candidate, searching within the entire resume might generate incorrect output as it can extract skills which are written anywhere within a resume, like within a project information or work experience. Whereas segmentation of a resume will extract the skills listed within the Technical skills or Skills heading. The information required can therefore be obtained with greater accuracy and efficiency by searching within a specific segment related to a known topic. To perform segmentation of the resume we have utilized the NLP

concepts of fuzzy string matching and n-grams.We have considered 3 major headings of a resume which are usually mentioned in a resume: Skills, Work Experience and Education. Within these 3 headings we extract all the entities. The index of each heading if found in the resume is passed onto the respective function and information is extracted.

Fuzzy string matching

Fuzzy string matching is used to perform approximate matching within strings.[9] It checks for the similarity of the strings being compared. In our application we have used fuzzy string matching to match the resume headings with the predefined list of headings. A list of synonyms and possible headings list which are mentioned in a resume has been created. The n-grams of headings are matched with the most suitable content, for example, the Education qualification in the candidate's CV is mapped according to the heading i.e "Qualification" or "Academic Qualification". N-grams are a set of words which are considered together, for example bigram is the list of all words in the resume in group of 2. Sometimes headings are mentioned in 2 words like Academic Qualification or Work Experience, hence bigrams are extracted and searched for. The set of headings is prepared in order to map the meaningful information. It includes approximate string matching, which is a method of searching for a string that closely matches the given string if there are no exact matches. A similarity matrix is used in order to find the approximate matches. As we want to extract only one string with the highest similarity score, the *extractOne* () function is used which returns an accurate match corresponding to the particular heading. If we find the heading its starting index is noted and only that segment of the resume is extracted and passed to the function.

B	C	D	E	F	G	H	I	J	K
Name	Contact	Education	Languages	Skills	Projects	Experience	Certification	Publications	Volunteering
Full Name	Phone	Studied	Languages Spoken	Technical Skills	Passion Projects	Employment History	certify	publishing	Initiatives
	PERSONAL PARTICULARS	Schooling	Dialects	Skill Highlights	Recent Projects	Work Experience	document	declaration	extracurricular activities
		Tutoring	Language Skills	Specializations		Professional Experience		journal	Volunteer Experience
		Teaching		Additional Qualification		Internships		Report	Additional Activities
		Educational Details		AREAS OF EXPERTISE		Experiences		newsletter	Community Service
		Training		Primary Skills		Work Summary			
		Coaching		Computer Skills		Employer Details			
		Academic Background		IT SKILLS		Industrial Experience and Project Details			
		Educational History		SOFTWARE SKILLS		Professional History			
		Academic Qualification				My Contribution			

```
education_choice=['education ','educations','educational details','educational profile','educational qualifications','academic qualifications']
skills_choice=['skill','skills','technical skill','skill highlights','Computer Skills','it skills']
experience_choice=['experience','professional experience','industry experience','employment history','work experience','work summary','professional history','employer details']
titles=[education_choice,skills_choice,experience_choice]
def find_headings(resume_text):
  flag=-1
  for i in titles:
      for j in i:
        #print(j)
        list_of_bigrams=generate_N_grams(resume_text,2)
        resume_heading=process.extractOne(j,list_of_bigrams)
        if resume_heading[1]==100:
          print(resume_heading)
          flag=titles.index(i)
          resume_title.append(resume_heading[0])
          print(resume_title)
          resume_index.append(resume_text.index(resume_heading[0]))
          print(resume_index)
          if flag==1:
            print(resume_index[-1])
            find_skills(resume_text[resume_index[-1]:])
            break
          elif flag==0:
            find_education(resume_text[resume_index[-1]])
            break
          elif flag==2:
            find_experience(resume_text[resume_index[-1]])
            break
  if flag==-1:
    find_skills(resume_text)
    find_education(resume_text)
```

Fig 2. Segmentation of headings in resume

Knowledge Extraction

Spacy model is used to perform all the natural language processing tasks. The knowledge base of technical skills, organizations, job roles, college names is given as a list of patterns to the Phrase Matcher component

of the Spacy model.[5] The model adds this component in its pipeline. When the text document is given as an input to the model each component acts on the text and passes its output to the next component in the pipeline. It starts with tokenization and ends with named entity recognition element.

Regex Pattern Matching

Regex pattern matching is a powerful technique used in text processing to search and extract particular patterns from a piece of string. We define a search pattern, which is then used to match or find all occurrences of the pattern within a larger text.[13] In a resume some entities have a common pattern, for example: email address has an ending pattern @gmail.com/@hotmail.com. Searching for this pattern of regular expression within the entire resume will extract the required results. Regex pattern matching is one of the techniques that has been integrated in the pipeline to extract the date, years of experience, year of passing, phone number, and email id from the resume.

Phrase matcher- In Spacy, the matcher is a rule-matching engine that works with tokens similarly to regular expressions. The phrase matcher is a rule-based approach. It is also possible to use custom callbacks such as merging entities and applying custom labels. The Spacy allows us to look up the phrases and words we are seeking as well as the tokens contained within the document and their relationships.

We built the pre-trained word embeddings of **skills** specific to the resume which are related to the information technology domain.**Keyword-based** matching identifies words, phrases, and patterns in the resume of a candidate. It was implemented to extract the candidate's name, skills, education qualifications, Designation, company name, and university name.

Testing

Although NLP is a suitable approach for extracting relevant information about desired entities, sometimes it fails to recognize the entities. The drawback of regular expressions and hand-crafted rules is that they cannot generalize well if not properly defined, resulting in limited usage of these methods. For example, instead of returning the candidate's name, the model returns the university name or publication's name.

Deep Learning Information Extraction Module

Deep Learning models have recently proved to be very effective for Named Entity Recognition. Some of the models used for these tasks include Recurrent Neural Networks(RNN), Convolution Neural Networks(CNN), Transformer based models and Conditional Random Fields(CRFs).[14]

We have utilized transformer-based models because they are pre-trained on a large corpus of dataset and can be fine-tuned to achieve state of the art results. Some of the cons of using deep learning models is their intensive use of computational resources, training time and dataset requirement. As we were computationally efficient and had created a good amount of training dataset according to our custom requirements we proposed the hybrid approach - combining the results of both the BERT model and NLP functions. There are multiple transformer-based models like BERT, Ro-berta etc. BERT based models specifically used for ner tasks are available to implement from the Hugging Face transformers. Spacy library also allows you to integrate your custom transformer-based models in its pipeline which is one of the most important advantages for the custom named entity recognition tasks. The entire text gets processed once and predicts the entities. This feature has been added in the latest version of the spacy library.

BERT Model

Bert stands for Bidirectional Encoder Representations Transformer was developed by Google by training it on a massive dataset of unlabelled text followed by fine-tuning on specific NLP tasks. The earlier models developed worked only in one direction of processing texts, whereas the BERT model works bi-directionally

by analysing the contextual meaning of the word from both sides right to left and left to right which captures the meaning of the text more effectively. This is a great advantage for our use-case as in resumes there is no particular fixed structure or format that a candidate follows hence understanding the context of the surrounding words will help in predicting the entity. The following steps were implemented while developing the model.

Collection of Training Dataset and annotation

Deep learning models like BERT require a sufficient amount of training data.[3] We collected 500 resumes of varying file formats and structures and annotated the resume text using Ner text annotator online tool. Here we upload the text inside the resume and select the word which corresponds to a particular entity. Once all entities are marked a json file can be downloaded which consists of all the entities annotated with the starting and ending index of each entity. This online tool made the annotation task easier and faster. The training dataset loaded into the bert model is a json file which consists of 500 resumes text and annotated entities.

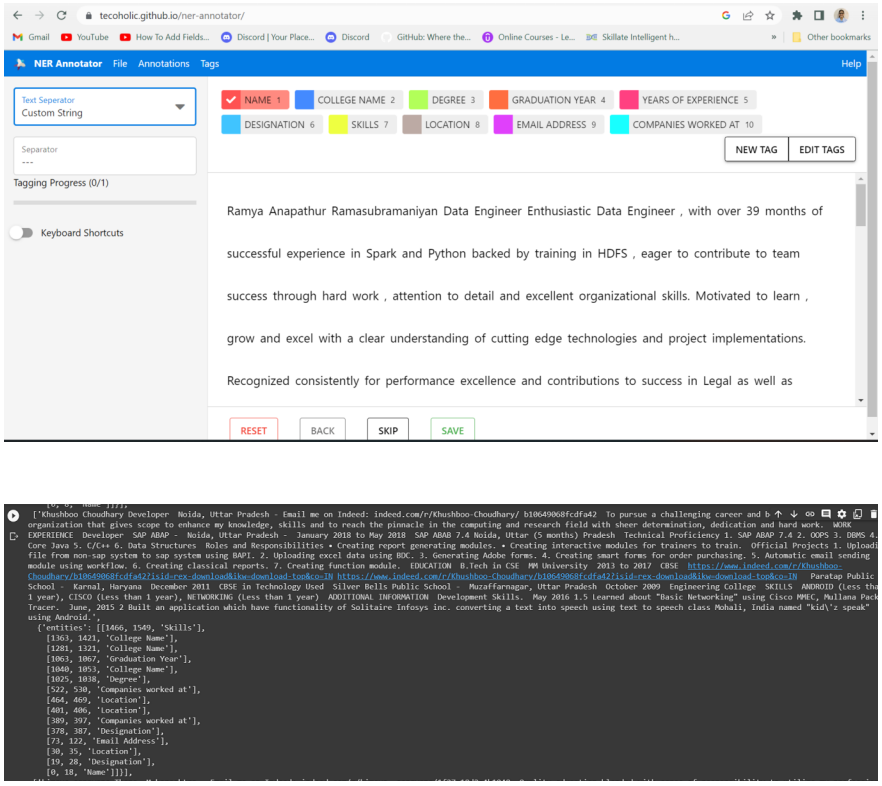

Fig 3. Training Dataset

Features

After creation of the model, the input sentence is first tokenized and then fed into the BERT model. The trained BERT model then classifies the text into a particular entity based on its training.The BERT model itself includes multiple layers of self-attention and feedforward neural networks, which are used to encode the input sentence and extract features that are relevant for NER. The output of the model for each token in the sentence includes a vector representation that captures the semantic meaning of the token in the context of the sentence. Additionally BERT based NER models utilize features like attention masks, segment embeddings, and the positional embeddings (which indicates the position of each entity). The BERT based NER model has been fine-tuned on a labelled dataset which includes the entire resume text along with the

entities. We have considered a total of 10 entities. Apart from the standard entities that Named Entity Recognition process identifies: Name, Location, Organization etc we have included entities that will be helpful to a recruiter while analysing a candidates profile: Name, College Name, Degree, Graduation Year, Years of Experience, Designation, Skills, Location, Email address, Companies worked at.

Spacy- integration

Spacy is an open-source Natural Language Processing library that performs various NLP operations such as tokenization, named entity recognition, dependency parsing etc. In the latest version of Spacy it provides a feature of integrating the state-of-art hugging face transformers by importing the spacy-transformers package. Transformers like BERT,GPT-2,XLNet etc can also be fine-tuned and trained on the custom labelled datasets to perform varied nlp tasks. The spacy model created adds the transformer as a component in its pipeline and instead of spacy performing the named entity recognition the doc is passed to the transformer component and prediction is performed.

Training BERT model

Initially the BERT model has been trained on the colab notebook platform using the GPU as the computation power and training time required to train this model is extensive. Using the GPU makes the process a little faster. This BERT model has been trained for a duration of approximately 1hr 30 minutes and 47 cycles of epoch. We have achieved an accuracy of 90-92%. The loss of the transformer keeps oscillating with the minimum being 481.07.**Fig 4.** BERT Model Training

Testing

We have tested the accuracy of the model by giving different types of resumes as inputs. The bert model is able to accurately predict the maximum number of entities. The screenshot below provides the results of the model.**Fig 5.** Output of BERT Model

Hybrid Methodology

The resume of a particular candidate is very subjective in nature and does not have a fixed structure or format. Hence relying on any one technique of extracting information can be risky. Natural Language Processing techniques are helpful to annotate entities in which the resume information is very standard and structured in nature. Only a set of key terms are used which makes it easier to extract using a predefined knowledge base. But this ideal situation is not possible at all times. To ensure flexibility and more accurate results techniques like machine learning and deep learning had to be explored. We realized that deep learning tools can be integrated easily to perform custom Named entity recognition and they consider the contextual along with the semantic meaning of the annotated entities. Once we train our model to identify those patterns it makes capturing of information from resumes highly accurate.

Our proposed methodology will include the best of both the methods to improve accuracy and efficiency. Initially the resume will be pre-processed and the entire resume text will be provided as input to the deep learning model. The model will annotate the encountered entities. If any entity has been missed or remains unidentified the pre-processed resume text will be passed on to the Natural Language Processing pipeline and remaining entities will be annotated. Our primary focus for this entire process has been to extract meaningful information from a resume. Therefore we have tried to include the best retrieval techniques and achieve an optimum output.

There is no particular metric/criteria to calculate the accuracy of a hybrid model as it optimizes the retrieval process, it tries to combine the best of both the techniques. Therefore to ensure that it actually enhances the process we have performed rigorous testing on multiple formats of resumes. We encountered that the maximum number of entities were annotated with the hybrid method.

Fig 6. Hybrid System Architecture

Video Resume

Nowadays video resume is a prevailing concept and is used to showcase the skills, experience, education and contact details of the candidate in a short and concise manner. Corporates usually prefer a video resume as it is convenient. In our web application there is an option to upload a normal pdf/word resume or a video resume. If a video resume is uploaded, we convert the Speech in the video to text using Automatic Speech Recognition(ASR). This text is then given as input to the BERT model. This model then performs normal Named Entity Recognition and the encountered entities will be identified.

By automatically extracting relevant information from video resumes, these algorithms can help to speed up the hiring process and ensure that the most qualified candidates are identified and considered for the position.

4. Results

In this research work we have proposed a combined approach towards Resume Parsing. Currently our system works for the resumes of the Computer Science and Information Technology domain. The skills, companies and designation knowledge base have been made keeping these domains in mind. We have achieved an accuracy of 90-92% with a deep learning model. This accuracy needs to be further increased by expanding the volume of the dataset of resumes and including resumes from other domains as well. The knowledge base utilized also needs to be fine-tuned to improve its accuracy. The future work includes inclusion of a variety of file formats, more number of entities like publications, extra-curricular activities, hobbies etc to be included to provide a more accurately summarized profile of the candidate. Our main aim is to make the recruiting process faster and easier. This system will be free to use for the candidate so that they get the feel of ATS software. The candidates can include appropriate key terms which get easily identified by the companies. A feedback system for the recruiting company's features can be included to ensure a smooth hiring process.

5. Conclusion

In conclusion, deep learning information extraction models have significantly increased the accuracy and effectiveness of resume parsing when compared to conventional rule-based methods. Our study has shown how well these models work at pulling out pertinent data from resumes and putting it in structured formats for future examination. Deep learning models are advantageous for resume parsing because they can handle different resume formats, languages, and layouts. They may also extract data that may not be expressed clearly in the resume. As a result, recruiting takes less time and costs less money because candidate screening and selection are more effective and efficient. Furthermore, by ensuring that all candidates are judged on their qualifications and skills rather than surface-level characteristics like name, gender, or race, the use of deep learning models in resume parsing can help decrease biases in the recruiting process. This may lead to a workforce that is more inclusive and diverse. Deep learning models, on the other hand, necessitate a lot of data for training, therefore they might not be appropriate for smaller businesses or organizations with tighter budgets. The quality and consistency of the input data, as well as the intricacy of the information to be extracted, may also affect how accurate the models are. Overall, our study has demonstrated that deep learning information extraction models can significantly enhance the hiring process when used for resume parsing. We think that carrying out more research and development in this area can help the hiring process become more effective and efficient while also advancing the field of HR technology.

Received: ((will be filled in by the editorial staff)) Revised: ((will be filled in by the editorial staff)) Published online: ((will be filled in by the editorial staff))

References

1. Kinge, Bhushan, et al. "Resume Screening Using Machine Learning and NLP: A Proposed System." (2022).
2. Tejaswini, K., et al. "Design and development of machine learning based resume ranking system." *Global Transitions Proceedings*3.2 (2022): 371-375.

3. Vukadin, Davor, et al. "Information extraction from free-form CV documents in multiple languages." *IEEE Access* 9 (2021): 84559-84575
4. Wosiak, Agnieszka. "Automated extraction of information from Polish resume documents in the IT recruitment process." *Procedia Computer Science* 192 (2021): 2432-2439.
5. Suresh, Yeresime, and A. Manusha Reddy. "A contextual model for information extraction in resume analytics using NLP's spacy." *Inventive Computation and Information Technologies: Proceedings of ICICIT 2020* . Singapore: Springer Singapore, 2021. 395-404.
6. Gaur, Bodhvi, et al. "Semi-supervised deep learning based named entity recognition model to parse education section of resumes." *Neural Computing and Applications* 33 (2021): 5705-5718.
7. Roy, Pradeep Kumar, Sarabjeet Singh Chowdhary, and Rocky Bhatia. "A machine learning approach for automation of resume recommendation system." *Procedia Computer Science* 167 (2020): 2318-2327.
8. Bhaliya, Nirali, Jay Gandhi, and Dheeraj Kumar Singh. "NLP based extraction of relevant resume using machine learning." (2020).
9. Gunaseelan, B., Supriya Mandal, and V. Rajagopalan. "Automatic Extraction of Segments from Resumes using Machine Learning." *2020 IEEE 17th India Council International Conference (INDICON)* . IEEE, 2020.
10. Amin, Sujit, et al. "Web application for screening resume." *2019 International Conference on Nascent Technologies in Engineering (ICNTE)* . IEEE, 2019.
11. Tikhonova, Maria, and Anastasia Gavrishchuk. "NLP methods for automatic candidate's cv segmentation." *2019 International Conference on Engineering and Telecommunication (EnT)* . IEEE, 2019.
12. Bhatia, Vedant, et al. "End-to-end resume parsing and finding candidates for a job description using bert." *arXiv preprint arXiv:1910.03089* (2019).
13. Zaroor, Abeer, Mohammed Maree, and Muath Sabha. "A hybrid approach to conceptual classification and ranking of resumes and their corresponding job posts." *Intelligent Decision Technologies 2017: Proceedings of the 9th KES International Conference on Intelligent Decision Technologies (KES-IDT 2017)–Part I 9* . Springer International Publishing, 2018.
14. Ayishathahira, C. H., C. Sreejith, and C. Raseek. "Combination of neural networks and conditional random fields for efficient resume parsing." *2018 International CET Conference on Control, Communication, and Computing (IC4)* . IEEE, 2018.
15. Mohamed, Ashif, et al. "Smart talents recruiter-resume ranking and recommendation system." *2018 IEEE International Conference on Information and Automation for Sustainability (ICIAfS)* . IEEE, 2018.
16. Sanyal, Satyaki, et al. "Resume parser with natural language processing." *International Journal of Engineering Science* 4484 (2017).
17. Chandola, Divyanshu, et al. "Online resume parsing system using text analytics." *Journal of Multi-Disciplinary Engineering Technologies* 9 (2015).
18. Sonar, Swapnil, and Bhagwan Bankar. "Resume parsing with named entity clustering algorithm." *Published paper, SVPM College of Engineering Baramati, Maharashtra* (2012).