REVA UNIVERSITY
Bengaluru, India

A Project Report on

Resume Shortlisting and Ranking with Transformers

Submitted in Partial Fulfilment for Award of Degree of

Master of Technology

In Artificial Intelligence

Submitted By

Vinaya James

R20MTA11

Under the Guidance of

Akshay Kulkarni

Mentor,RACE | Manager, Publicis Sapient

REVA Academy for Corporate Excellence - RACE

REVA University

Rukmini Knowledge Park, Kattigenahalli, Yelahanka, Bengaluru - 560 064

race.reva.edu.in

August, 2022

REVA UNIVERSITY
Bengaluru, India

Candidate's Declaration

I, **Vinaya James** hereby declare that I have completed the project work towards the **Master of Technology in Artificial Intelligence** at, REVA University on the topic entitled **Resume Shortlisting and Ranking with Transformers** under the supervision of **Akshay Kulkarni**, Mentor, Artificial Intelligence, RACE, REVA University. This report embodies the original work done by me in partial fulfillment of the requirements for the award of a degree for the academic year **2022**.

Place: Bengaluru Vinaya James

Date: 27-08-2022

REVA UNIVERSITY
Bengaluru, India

Certificate

This is to Certify that the project work entitled **Resume Shortlisting and Ranking with Transformers** carried out by **Vinaya James** with **SRN R20MTA11**, is a Bona fide student of REVA University, is submitting the project report in fulfilment for the award of Master of Technology in Artificial Intelligence during the academic year 2022. The Project report has been tested for plagiarism and has passed the plagiarism test with a similarity score less than 15%. The project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the said degree.

Akshay Kulkarni

Mr. Akshay Kulkarni

Dr. Shinu Abhi

External Viva Panelists

Names of the Examiners

1. Dr. Santosh Nair, Founder, Analytic Edge
2. Rajkumar Dan, Data Scientist Consultant, Dell

Place: Bengaluru

Date: 27-08-2022

REVA UNIVERSITY
Bengaluru, India

Acknowledgment

I am highly indebted to **Dr. Shinu Abhi**, Director,RACE Corporate Training, REVA university for the guidance and support provides throughout the course and my project.

I would like to thank **Mr. Akshay Kulkarni** for the valuable guidance provided as my project guide to understand the concept and in executing this project.

It is my gratitude towards our Chief Mentor, **Dr. Jay Bharateesh Simha**, and all other mentors for the valuable guidance and suggestions in learning various data science aspects and for the support. I am grateful to them for their valuable guidance on several topics related to the project. I am thankful to my classmates for their support, suggestions, and friendly advice during the project work.

I would like to acknowledge the support provided by the founder and Hon'ble Chancellor, **Dr. P Shayma Raju**, Vice-Chancellor, **Dr. M. Dhanamjaya**, and Registrar, **Dr. N Ramesh**. Sincere gratitude is extended to all personnel of the RACE program office who were consistently helpful in meeting all program requirements. It is my sincere gratitude towards my parents and my family for their kind co-operation. Their encouragement also helped me in the completion of this project

Place: Bengaluru
Date: 27-08-2022

REVA UNIVERSITY
Bengaluru, India

Similarity Index Report

This is to certify that this project report titled **Resume shortlisting and ranking with Transformers** was scanned for similarity detection. Process and outcomes are given below.

Software Used: https://www.turnitin.com/
Date of Report Generation: 17-Oct-2022
Similarity Index in %: 13%
Total word count: 5963
Name of the Guide: Mr. Akshay Kulkarni

Place: Bengaluru
Date: 27-08-2022

Vinaya James

Verified by: M N Dincy Dechamma

Dr. Shinu Abhi,
Director, Corporate Training

Place: Bengaluru
Date: 27-08-2022

List of Abbreviations

Sl. No	Abbreviation	Long Form
1	NLP	Natural Language Processing
2	BERT	Bidirectional Encoder Representations from Transformers
3	SBERT	Sentence BERT
4	PDF	Portable Document Format
5	MS	Microsoft
6	AI	Artificial Intelligence
7	NER	Named-Entity Recognition
8	HR	Human Resource
9	RoBERTa	Robustly Optimized BERT Pretraining Approach
10	ALBERT	A Lite BERT
11	OOV	Out-Of-Vocabulary
12	NLI	Natural Language Inference
13	SNLI	Stanford NLI
14	DAN	Deep Averaging Network
15	EDA	Exploratory Data Analysis
16	LDA	Latent Dirichlet Allocation
17	MLM	Masked Language Model
18	STS	Semantic Textual Similarity
19	biLM	Bidirectional Language Model
20	AE	AutoEncoding
21	AR	AutoRegressive
22	MMR	Maximal Marginal Relevance
23	CLS	Classical Least Squares
24	NLTK	Natural Language Tool Kit
25	JD	Job Description

List of Figures

List of Tables

Abstract

The project will help the human resource domain to eliminate the time-consuming task of the recruitment process. Screening resume is the most critical and challenging task for human resource personnel. NLP techniques are the computer's ability to understand the spoken/written language. Now a day's online recruitment platform is more with consultancies. So, a single job most times will get hundreds of applications. HR personnel put more labour into the candidate selection area to find the best fit for the job.

Most time shortlisting the best fit for the job is time-consuming and finding an apt person for the job is hectic. The project will help to shortlist the candidates with a better match for the job based on the skills provided in the resume. As it's an automated process, the personalized favour and soft skills of the candidate will not be affected by the hiring process.

Sentence-BERT (SBERT), a modification of the BERT network using Siamese and triplet networks that can derive semantically meaningful sentence embeddings.

An end-to-end tool for the HR domain, which takes hundreds of resumes along with required skills for the job as input and provides the better-ranked candidate fit for the job as output. The SBERT is compared with BERT and proved that it was superior to BERT.

Keywords: Natural Language Processing, Sentence-BERT, Automatic Recruitment Process, Sentence Embedding.

Table of Contents

Chapter 1: Introduction

In a business or organization, it is indeed critical to make the proper hiring decisions for particular positions for Human Resources Manager or Head-hunter [1]. Recruitment tools like "LinkedIn" and "Monster" search for skills and identify candidates who are qualified for open positions. The list of resume search results may be lengthy. All resumes should be manually reviewed to identify possible applicants. Especially, large companies like "Google" frequently receive hundreds of thousands of resumes each year for job applications. As a result, automation is introduced to make the work easy with time-saving.

The introduction of NLP simplifies the process of automatic sentence or test recognition from papers or resumes [2]. In general vector, representations have become an ever-existing entity in the NLP context. Word embedding properties have seen major developments in the recent past, such as the superposition of word contexts [3], pointwise linkages to shared data values between related words, and linear substructures [4]. However, locating sentences that are comparable based on context, meaning, subject, etc. is a problem in naturally occurring language processing, which leads to the issue of statement integration. The text can be organized, produced, processed, etc., and based on this, it converts the vectors of the words into a vector representation of the sentence. In contrast, an embedding definition is a numerical representation of a word, phrase, sentence, or longer natural language utterance in a particular space. Additionally, word embeddings distributed semantic vector representations of words have drawn a lot of interest in recent years and have undergone a lot of changes [5]. The word embeddings were developed by analyzing the word distribution among arbitrary text data, and because they are familiar with word semantics, they are a crucial part of many semantic similarity algorithms.

Compared to word embeddings, sentences encompass a higher level of information: the complete semantics and syntactic declarations of the sentence. The word2vec model, which is one of the most popular models, employs neural

networks to create word embeddings [6]. Recent transformer-based models' pre-trained word embeddings provided cutting-edge outcomes in a variety of NLP tasks, including semantic similarity. Based on this, modern language models like RoBERTa, BERT [7][8], and ALBERT use transformers to traverse the underlying corpus in both directions and create vector representations of text data. In 2009, the BERT transformer model outperformed the best results available for different NLP tasks, including semantic similarity. The BERT model is typically tuned to serve a specific NLP task using labeled training data. Multilingual BERT is enhanced by the alignment method suggested by Cao et al. [9]. Thus, the BERT models produce bilingual illustrations of words that consider the word's context in both directions.

Performing sentence pair regression, like clustering and semantic search [10], can be time-consuming with BERT due to the large corpus size. Converting a sentence into a vector that encodes the semantic meaning of the sentence is one efficient technique to address this issue. Transformers are the guide to a state-of-art model, and the next section discusses more works accomplished on the NLP concept.

Chapter 2: Literature Review

As sentence embeddings are more realistic representations of text, recent papers on sentence embedding transformers have focused on the mechanics of recognition. Fast Text is a model that incorporates a character-based n-gram model in addition to the established word embedding techniques Word2Vec and GloVe [11]. This makes it possible to calculate word embeddings from the vocabulary.

Vaswani et al. [12] suggested the Transformer, a self-attention network, as a remedy for the neural sequence-to-sequence issue. When a self-attention network is used to visualize a phrase, each word is represented by a scaled sum of all the other words in the phrase. Liu et al. [13] used a sentence's inner attention to show that self-attention pooling existed before self-attention networks. By generalizing scalar attention to vectors, Cho et al. [14] devised a fine-grained attention approach for neural machine translation. Natural Language Inference (NLI) and other supervised transfer tasks can help complex phrase encoders that are often pre-trained like language models. The Universal Sentence Encoder improves unsupervised learning by training a transformer network on the Stanford NLI (SNLI) dataset. According to Giorgiet al. [15], the task that sentence embeddings are trained on greatly influences their quality. The SNLI datasets, according to Conneau et al. [16], are appropriate for training sentence embeddings.

Therefore, to successfully develop the Chinese Depth Approximation Networking (DAN) and transformers, Parameswa et al. [17] offer an approach that uses Internet responses. Naseem et al. [18] fed the static word embeddings GloVe through the deep neural networks and carried out a controlled neural interpretation operation to obtain the perspective encoding. The Remiers et al. [19] ELMo model is a method for a deeper summarized description. The internal states of words are used to teach a deep bidirectional Language Model (biLM) that has already been trained on a large text corpus. Pre-trained language models of the fine-tuning variety have unfrozen pretrained parameters that can be

changed for a new assignment. The text presentation is no longer extracted using this approach. Two significant tasks were introduced by Devlin et al. [20] in their autoencoding pre-trained language model BERT, a deep bidirectional Transformers model: Mask Language Model (MLM) and Next Sentence Prediction (NSP). During the tuning stage, the only activities that deviate from one another are those at the input and output layers. Dai, Zihang, et al. [21] proposed the Transformer-XL generalized autoregressive pre-trained language model, which can learn bidirectional contexts by maximizing expected likelihood across all possible factorization order permutations. Additionally, two groups of the pre-trained language model can be identified based on the pre-training procedure: Auto Encoding (AE) and Auto-Regressive (AR). AR language models like ELMo and XLNet aim to estimate the probability distribution of a text corpus by employing an autoregressive model. However, AE language models like BERT and its variations like RoBERTa by Liu, Yinhan, et al. [22] seek to recreate the original data from corrupted input without resorting to explicit density estimates.

One major problem with BERT-type models is the introduction of fictitious symbols like MASK during pretraining, despite their complete absence from the final output text. The pre-trained model effectively illustrates how words or phrases link to one another and access various data. Thus, embedding-based key phrase extraction has recently demonstrated strong performance. Lee et al. [23] suggested a Deep Belief Network (DBN) to model the hierarchical relationship between key phrase embeddings. It is simple to distinguish the target document from others using this strategy. Reference Vector Algorithm (RVA) by Papagiannopoulou et al. [24] is a key phrase extraction technique that uses local word vectors as a guiding principle, employing an average of the embeddings trained on distinct files using GloVe as the reference vector for all candidate phrases. The score used to rank the candidate key phrases is then determined by calculating how closely the embeddings of each candidate key phrase match those of the reference vector.

Bennani-Somerset et al. proposed EmbedRank [25] based on the cosine similarity between the candidate key phrase's embeddings and the document's sentence embeddings. EmbedRank creates a document representation using the phrase embedding models Doc2Vec and Sent2Vec. They employ Maximal Marginal Relevance (MMR) to further broaden the keyword's applicability. Pre-trained language models, in particular, BERT, have recently gained popularity and significantly enhanced performance for several NLP applications. Pre-trained BERT and its variations have mostly proved successful in the English language. A language-specific model might be retrained using the BERT architecture for other languages, or one could employ pre-trained multilingual BERT-based models.

Chapter 3: Problem Statement

Identifying a qualified candidate for the job is a complex undertaking. Typically, manual processes are used in the traditional hiring process. The HR department's qualified recruiters and other significant resources are needed for the manual recruitment process. Businesses sometimes receive a substantial volume of resumes for each job opening, some of which may not even be suitable for the position. Additionally, these hiring procedures take a lot of time and effort to discover qualified applicants for open positions.

Therefore, manually selecting the most pertinent applicants from a lengthy list of potential candidates is difficult. Numerous recent research has focused on the drawbacks of the manual hiring process. Dealing with resumes in the advertising of job specifications and hiring procedures. Selecting people who fit a given job profile is a vital task for most firms. As online hiring becomes more common, traditional hiring methods become less successful.

Recruiting the most pertinent multilingual candidates through the manual hiring process is one of the most critical issues in multilingual job offers and resumes.

The proposed model investigates developing a resume shortlisting and ranking with Transformers to address the difficulty of selecting the right candidate out of two hundred resumes. An automated recruiting system is essential to make it easier for job seekers to access recruitment opportunities and to minimize the amount of human effort involved in the hiring process.

Chapter 4: Objectives of the Study

The main goal is to improve the current resume ranking scheme and make it more adaptable for both parties. (1) *Those who were hired as candidates:* Candidates who have recently graduated and are looking for a job. A significant portion of those applicants is so desperate that they are willing to work in any position unrelated to their skill set and abilities. (2) *The client organization that recruits the applicants:* A job recruiter's objective is to select the top candidate from all qualified applicants based on the Job Description (JD). The process takes time for both the individual and the business. With an automatic resume sorting system, the business may produce the finest candidate list possible based on the limits and requirements they gave for that particular role. Since the appropriate person will be hired for the position, this hiring method will benefit both the candidate and the organization. Therefore, neither the client firm nor the hired candidate would have any regrets.

The three objectives of this study are:

The three objectives of this study are;

1. *Collect the resumes as per the defined JD:* The resumes are collected by HR from online job platforms, referrals from existing employees, and third-party consultancies. But getting the exact JD-related resumes are challenging. For this model two hundred resumes have been collected.

2. *Build a custom algorithm to shortlist the resume as per the JD given:* The skills are extracted to a *pandas* data frame with the help of *ResumeParser*. The resume matches will be shortlisted and added to a list based on the JD that HR has provided.

3. Create a ranking *algorithm to get the best out of shortlisted resumes:* JDs and candidates' skill sets are compared, and the most suitable individual with the necessary skills is shortlisted. The model is trained using SBERT and

BERT model encodings, and the top N resumes are sorted according to their cosine similarity to sentences and words, respectively.

Chapter 5: Project Methodology

Finding the "right" applicant for a position has never been simple. In addition to having the required training and work experience, a prospective employee typically needs to fit in with the current team and support the company's vision. A new dilemma has emerged with the rise of internet job boards and globalization. Today's recruiting professionals frequently need to analyze hundreds of online profiles and resumes only to pick whom to approach due to how simple it has become to build an online profile and apply to a position with a few clicks.

It is not surprising that various technology solutions have been offered to aid recruiters in addressing the problem of candidate screening because automating the shortlisting of candidates can lead to lower costs and higher recruiter productivity. To rank the resume successfully, an efficient context test-based embedding is needed. To achieve textual similarity, transfer-based models like BERT and XLNet are used. In addition, an efficient pre-trained model is required due to its poor accuracy. SBERT, a version of the BERT network that can produce semantically significant sentence embeddings by combining Siamese and triplet networks, is employed in this way [26]. The suggested architecture is depicted in Figure No. 5

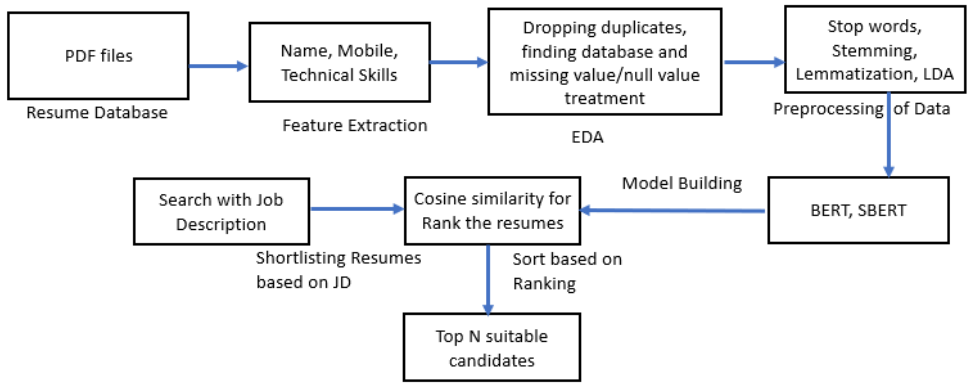


Figure No.5: Proposed Model

The number of resumes is the input used in this proposed model methodology to shortlist the most suitable individuals. *ResumeParser* is used to extract the

required details of jobseekers, like Name, Mobile Number, Email Address, and Skills. Exploratory Data Analysis (EDA) is used to remove duplicates, locate databases, and predict and remove missing or null values from text or sentences in resumes. The next step is data pre-processing. At this step, improper words are eliminated, the text is normalized, and the words are prepared for further processing with stop words, Stemming, Lemmatization, and Latent Dirichlet Allocation (LDA). Making vector representations of all words and documents and collectively embedding them in a common vector space is the first step in the extraction of relevant skills and expertise. Then SBERT and BERT are used to create a model.

The produced embedding vectors can be subjected to the Semantic Textual Similarity (STS) task employing metrics such as the Manhattan distance, Cosine similarity, and Euclidean distance. BERT, Siamese networks, and Pooling layer are the three key principles that the SBERT architecture employs. Using a pre-training method called MLM, which typically masks a portion of the tokens in a sentence and predicts them based on their context. BERT is a deep bidirectional transformer because MLM predicts that tokens will approach it from both directions. This is different from traditional left-to-right models, which only work one way. The JD is then matched with terms described in skills to get a cosine similarity score that may be used to rank and shortlist candidates appropriately in the HR field.

Chapter 6: Resource Requirement Specification

6.1 NLTK

Natural Language Tool Kit [NLTK] 3.5 [39], sentences were broken up into paragraphs, and words were lemmatized. Programming language is the most well-known framework for building Python programs that make use of human language data. It offers user-friendly interfaces to more than 50 corpora and lexical resources, including WordNet, as well as several text processing libraries for several languages. The text analysis process examines the text of the query by chunking it into words and identifying word categories (POS tags) such as nouns, verbs, adverbs, adjectives, etc. It is descended from the root adverb "near" which is where the word "nearest" comes from and determines after using the geodatabase, the general query is called.

6.2 Data Pre-Processing

NLP techniques are typically needed for systems that enable resume word tabulations and sentence analyses. Of course, there are variations. However, sometimes unnecessary or "stop" words like "of" and "a," which frequently contribute little to meaning, are first eliminated. Sometimes, meaningless custom words are manually added to stop word lists. Then, words are "stemmed," perhaps replacing synonyms and changing "qualities" and "quality" to "quality." Finally, numbers are used in place of the stemmed words for grouping or other analysis tasks.

The term "dictionary" refers to a list of distinct words for each category of documents after the stop words have been eliminated and the words have been stemmed. These stemmed nonstop words can easily address several fields in a database by adding field names to them. Both words with several grammatical forms and words with similar meanings derived from them are handled by an algorithm. Taking all these stages into account, the techniques employed in the example are:

Step 1: Divide the text into words.
Step 2: Eliminate all punctuation and symbols and, if desired, lowercase all words.
Step 3: Eliminate the stop words.
Step 4: Use the Snowball Stemming Algorithm to stem the words.
Step 5: Add parenthesis to each word before adding the field names (if appropriate).

6.3 Data extraction from resumes for skills and experience

Data extraction from resumes are always tiersome. Here we used ResumeParser for extraction of required data like Name, Mobile Number, Email Address, Skills.

```
! pip install resume-parser
```

6.4 Transformers

Sentence Transformers is a Python framework for cutting-edge embeddings of text, sentences, and images. To identify sentences with shared meaning, these embeddings can be compared, using cosine-similarity. For semantic textual similarity searches, semantic search, or paraphrase mining, this is helpful. The framework, which is built on PyTorch and Transformers, provides a sizable library of pre-trained models that have been adjusted for a variety of purposes.. The models pass stringent testing and operate across a range of jobs. The code is further optimized to offer the fastest speed.

Additionally, they use BERT 4's official TensorFlow code as the foundation of the project. Notably, limit the longest sequence to 64 characters in order to save GPU RAM. It use the settings (epochs = 1, learning rate = 2e 5, and batch size = 16) to fine-tune the siamese BERT on the dataset. The results might not match what they've released. The writers cited in Issue #50 of Sentence Transformers at https://github.com/UKPLab that this is a typical occurrence and could be

connected to the random seed. Keep in mind that their implementation is dependent on Huggingface5's Transformers repository.

6.5 Cosine Similarity

Using the cosine similarity approach, BERT measures how similar a dataset and a text are. For this kind of problem, the SBERT is preferable than a regular BERT model. Similar sentences may not always be represented in the vector space in the same ways according to the usual model. The embeddings created by SBERT, however, can be contrasted using a cosine similarity. The generation and comparison of sentence embeddings using SBERT was demonstrated to be quick, accurate, and successful. These embeddings could subsequently be used to compare resumes and job description.

Chapter 7: Software Design

The suggested model will be required the below inputs:

1. Resumes
2. Job description

After the NLTK data pre-processing, used SBERT cosine-similarity algorithm for similarity detection between the resumes and job description. The similar resumes which have highest rank will be listed based on the requirement provided by HR.

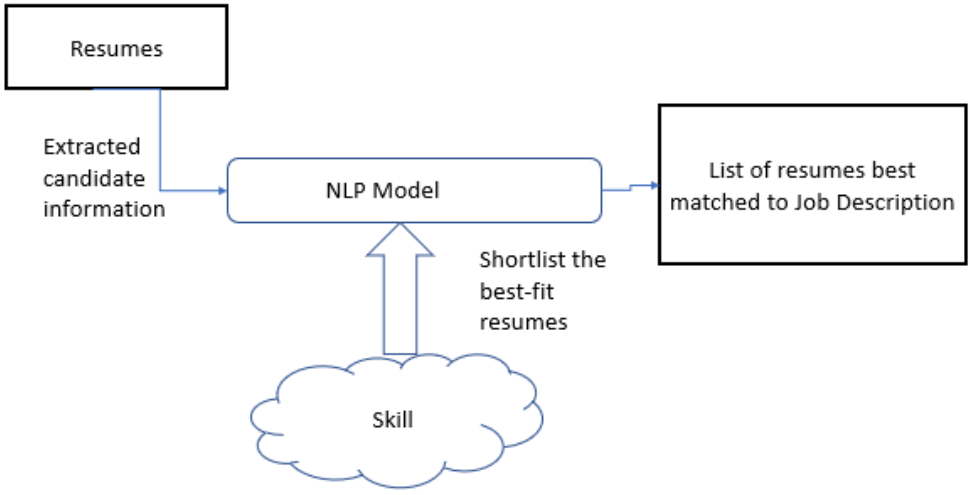


Figure No.7: Software Design

Chapter 8: Implementation

Two hundred resumes are gathered from various sources as part of the data collection process. All the necessary features, including Name, Email address, Mobile Number, and abilities, are retrieved with the aid of *ResumeParser*. A model for BERT and SBERT is constructed based on the JD following EDA and data preprocessing. The *N* number of resumes with the highest ranking is listed using the cosine similarity between the skill set and required JD.

SBERT for the STS task, permits two steps in the prediction of similarity:

Step: (1) First, using a sentence encoder, obtain sentence embeddings for each sentence.

Step: (2) Next, as the model-predicted similarity SBERT and BERT, compute the cosine similarity between the two embeddings of the input sentence. SBERT used *the bert-base-nli-mean-tokens* model and is compared to the BERT *bert-base-uncased* model.

Chapter 9: Testing and validation

Sentence embedding that outperforms the Classical Least Squares (CLS) vector is obtained by the average of the SBERT context embedding's one or two layers. The average of the last two layers of SBERT, denoted by *last2avg*, consistently yields better results than the average of the last layer of BERT. Since unsupervised learning methods like topic modelling do not guarantee that their results can be understood, correlation metrics have gained popularity among text-mining experts.

The degree of semantic similarity among top-ranking terms in each topic is measured by correlativity. It determines the co-occurrence scores of words in the modelled documents. As coherence also works with syntactic information with the aid of a sliding window that traverses across the corpus and checks occurrences. The notion behind coherence calculation is strongly related to embedding representations of text. Different methods can be used to calculate the correlation. SBERT gives a better solution than BERT when a comparison of top ten ranked resumes based on JD.

Chapter 10: Analysis and Results

The consistency and alignment of different sentence embedding models (BERT and SBERT) and their averaged STS results, are displayed in Table No.10. Among BERT and SBERT, models with superior alignment and homogeneity outperform in comparison with the models which do not have alignment and homogeneity. Scores for each pair of sentences are calculated by applying the cosine similarity scoring function to the sentence vectors. The SBERT method is used to arrange the sentences to maximize the sum of their similarities. The findings of this method show that the SBERT method always performs better than the BERT method.

Data Set	Model	Correlation value for Similarity
STS1	SBERT	0.42649
	BERT	0.194206
STS2	SBERT	0.378602
	BERT	0.119996
STS3	SBERT	0.377433
	BERT	0.047986
STS4	SBERT	0.374302
	BERT	0.156387
STS5	SBERT	0.373682
	BERT	0.182748
STS6	SBERT	0.373111
	BERT	0.048559

Table No. 10: Coherence value comparison for BERT and SBERT

Based on Table No.10, the graphical representation is provided as Figure No. 10.

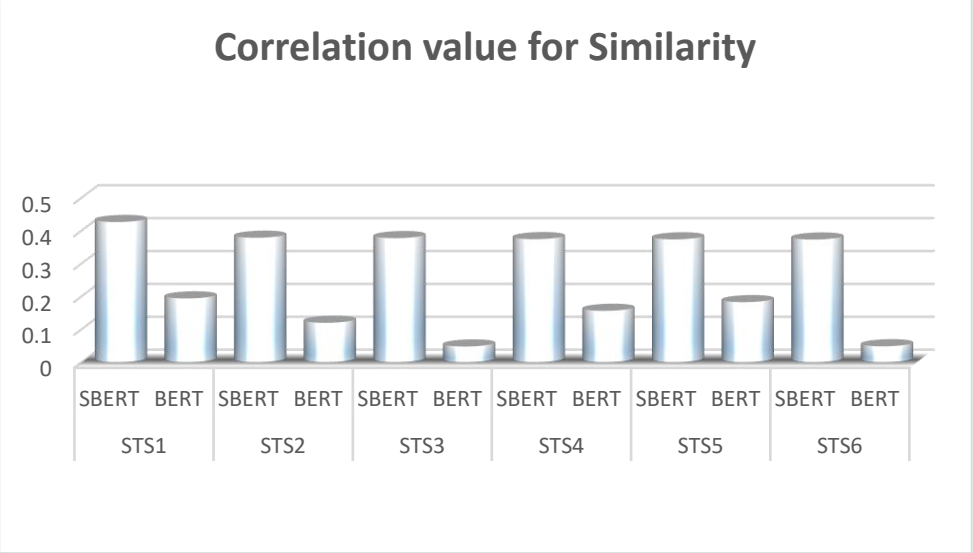


Figure No. 10: Performance analysis of SBERT with Coherence

Table No. 10 analysis reveals that the SBERT performs better than the BERT in terms of correlation. Compared to SBERT, the similarity variation is lower, as depicted in Figure No. 3. Resumes in the STS3 dataset exhibit a relatively low similarity variance. As a result, the SBERT executes the ideal sentence embedding to rank the candidate's information with the JD of the job provider in the minimum similarity variance.

Chapter 11: Conclusions and Future Scope

The proposed SBERT transformer helps recruiters screen resumes more quickly and effectively, cutting the cost of hiring. Thus, the company will then have access to a potential applicant who will be successfully placed in a business that appreciates the candidate’s skills and competencies. These days, many applicants submit applications for interviews. Every interview should include a review of resumes. Going through each resume one by one is not a good idea. It becomes quite difficult for the HR team to narrow down candidates for the following stage of the hiring process. The SBERT streamlines the process by summarizing resumes and classifying them by how closely they match the organization's necessary skills and requirements.

The proposed method evaluates candidates' skills and ranks them by the JD and skill requirements of the employing organization. A summary of their resume is supplied to provide a fast overview of each candidate's qualifications. One of the main issues is when a candidate lists skills for which they have no experience because the model focuses on the skill set listed on the resume submitted by the candidate. Artificial Intelligence techniques or any other effective sentence embedding transformers can be used for further improvement.

Bibliography

[1] Siddique, C. M. "Job analysis: a strategic human resource management practice." The International Journal of Human Resource Management 15.1 (2004): 219-244.

[2] Sanabria, Ramon, et al. "How2: a large-scale dataset for multimodal language understanding." arXiv preprint arXiv:1811.00347 (2018).

[3] Arora, Sanjeev, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. "A latent variable model approach to pmibased word embeddings." Transactions of the Association for Computational Linguistics 4 (2016): 385-399.

[4] Rieck, Bastian, and Heike Leitte."Persistent homology for the evaluation of dimensionality reduction schemes."Computer Graphics Forum.Vol. 34.No. 3. 2015.

[5] Stein, R. A., Jaques, P. A., & Valiati, J. F. (2019). An analysis of hierarchical text classification using word embeddings. Information Sciences, 471, 216-232.

[6] Mishra, Mridul K., and JaydeepViradiya."Survey of Sentence Embedding Methods." International Journal of Applied Science and Computations 6.3 (2019): 592-592.

[7] Chernyavskiy, Anton, Dmitry Ilvovsky, and PreslavNakov. "Transformers:“The End of History” for Natural Language Processing?." Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Cham, 2021.

[8] P. S. Suryadjaja and R. Mandala, "Improving the Performance of the Extractive Text Summarization by a Novel Topic Modeling and Sentence Embedding Technique using SBERT," 2021 8th International Conference on Advanced Informatics: Concepts, Theory and Applications (ICAICTA), 2021, pp. 1-6, doi: 10.1109/ICAICTA53211.2021.9640295

[9] Cao, Steven, Nikita Kitaev, and Dan Klein. "Multilingual alignment of contextual word representations." arXiv preprint arXiv:2002.03518 (2020).

[10] H. Choi, J. Kim, S. Joe and Y. Gwon, "Evaluation of BERT and ALBERT Sentence Embedding Performance on Downstream NLP Tasks," 2020 25th International Conference on Pattern Recognition (ICPR), 2021, pp. 5482-5487, doi: 10.1109/ICPR48806.2021.9412102.

[11] Dharma, EDDY MUNTINA, et al. "The accuracy comparison among word2vec, glove, and fasttext towards convolution neural network (CNN) text classification." J TheorApplInfTechnol 100.2 (2022): 31.

[12] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).

[13] González, José Ángel, Lluís-F.Hurtado, and FerranPla. "Transformer-based contextualization of pre-trained word embeddings for irony detection in Twitter." Information Processing & Management 57.4 (2020): 102262.

[14] Heeyoul Choi, Kyunghyun Cho, Yoshua Bengio,Fine-grained attention mechanism for neural machine translation,Neurocomputing,Volume 284,2018,Pages 171-176,ISSN 0925-2312.

[15] Giorgi, John, et al. "Declutr: Deep contrastive learning for unsupervised textual representations." arXiv preprint arXiv:2006.03659 (2020).

[16] Conneau, Alexis, et al. "Supervised learning of universal sentence representations from natural language inference data." arXiv preprint arXiv:1705.02364 (2017).

[17] Parameswaran, P., Trotman, A., Liesaputra, V., & Eyers, D. (2021). Detecting the target of sarcasm is hard: Really??. Information Processing & Management, 58(4), 102599.

[18] U. Naseem and K. Musial, "DICE: Deep Intelligent Contextual Embedding for Twitter Sentiment Analysis," 2019 International Conference on Document Analysis and Recognition (ICDAR), 2019, pp. 953-958, doi: 10.1109/ICDAR.2019.00157.

[19] Reimers, Nils, and IrynaGurevych. "Alternative weighting schemes for elmoembeddings." arXiv preprint arXiv:1904.02954 (2019).

[20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North

American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

[21] Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., & Salakhutdinov, R. (2019). Transformer-xl: Attentive language models beyond a fixed-length context. arXiv preprint arXiv:1901.02860..

[22] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

[23] Jo, T., & Lee, J. H. (2015). Latent keyphrase extraction using deep belief networks. International Journal of Fuzzy Logic and Intelligent Systems, 15(3), 153-158.

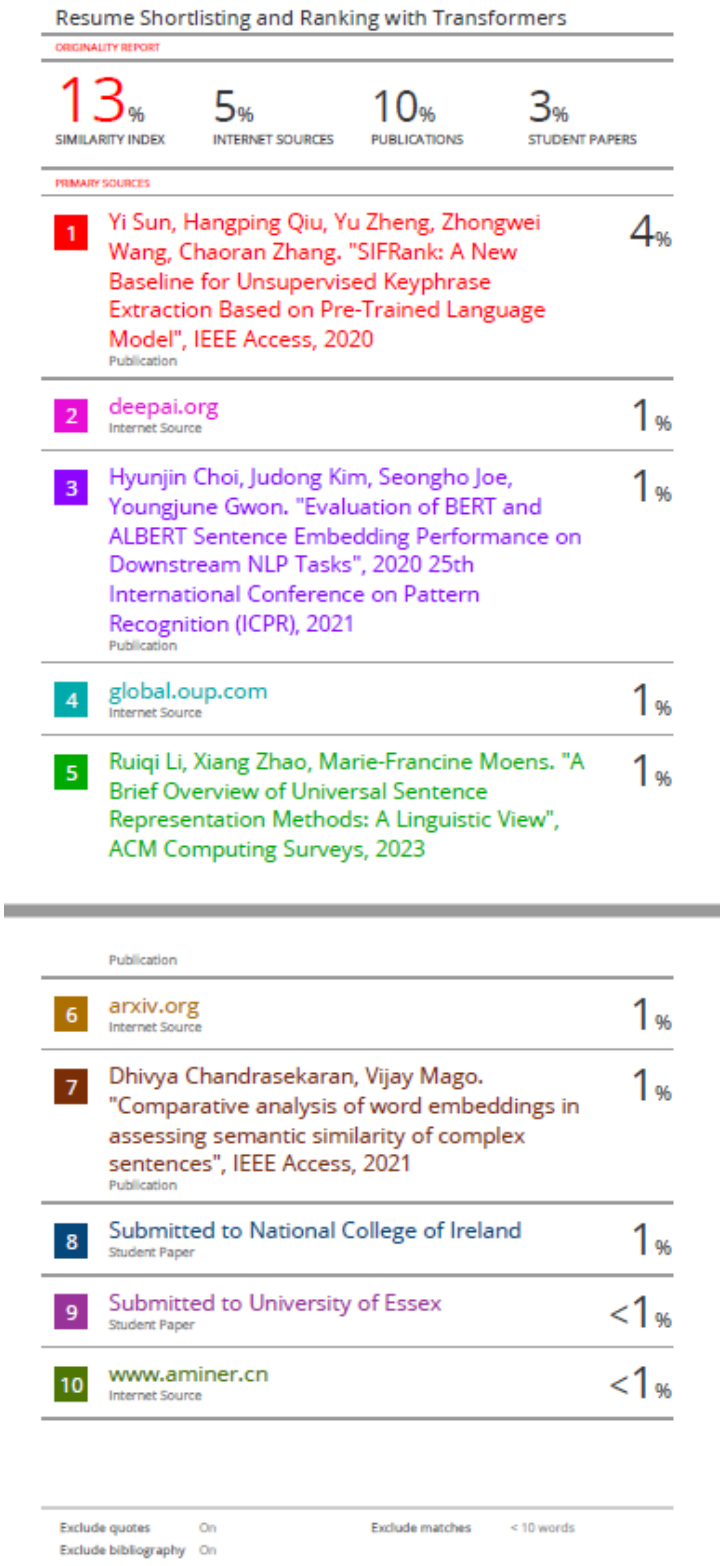
[24] Papagiannopoulou, Eirini, and GrigoriosTsoumakas. "Local word vectors guiding keyphrase extraction." Information Processing & Management 54.6 (2018): 888-902.

[25] Kamil Bennani-Smires, Claudiu Musat, Andreea Hossmann, Michael Baeriswyl, and Martin Jaggi. 2018. Simple Unsupervised Keyphrase Extraction using Sentence Embeddings. In Proceedings of the 22nd Conference on Computational Natural Language Learning, pages 221–229, Brussels, Belgium. Association for Computational Linguistics.

[26] Reimers, N., & Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084.

Appendix

Plagiarism Report



Resume Shortlisting and Ranking with Transformers

ORIGINALITY REPORT

13%	5%	10%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Yi Sun, Hangping Qiu, Yu Zheng, Zhongwei Wang, Chaoran Zhang. "SIFRank: A New Baseline for Unsupervised Keyphrase Extraction Based on Pre-Trained Language Model", IEEE Access, 2020 Publication	4%
2	deepai.org Internet Source	1%
3	Hyunjin Choi, Judong Kim, Seongho Joe, Youngjune Gwon. "Evaluation of BERT and ALBERT Sentence Embedding Performance on Downstream NLP Tasks", 2020 25th International Conference on Pattern Recognition (ICPR), 2021 Publication	1%
4	global.oup.com Internet Source	1%
5	Ruiqi Li, Xiang Zhao, Marie-Francine Moens. "A Brief Overview of Universal Sentence Representation Methods: A Linguistic View", ACM Computing Surveys, 2023 Publication	1%
6	arxiv.org Internet Source	1%
7	Dhivya Chandrasekaran, Vijay Mago. "Comparative analysis of word embeddings in assessing semantic similarity of complex sentences", IEEE Access, 2021 Publication	1%
8	Submitted to National College of Ireland Student Paper	1%
9	Submitted to University of Essex Student Paper	<1%
10	www.aminer.cn Internet Source	<1%

Exclude quotes On
Exclude bibliography On
Exclude matches < 10 words

GitHub Repository

https://github.com/vinayasandhiav/SBERT

Publications in a Journal/Conference Presented/White Paper[1]

Paper Submitted: *Vinaya James, Akshay Kulkarni, Rashmi Agarwal*, " **Resume Shortlisting and Ranking with Transformers**" i4c2022 - 4th International Conference on Circuits, Control, Communication and Computing (I4C) – 2022
Date of submission: 13/10/20

[1] URL of the white paper/Paper published in a Journal/Paper presented in a Conference/Certificates to be provided.

Resume Shortlisting and Ranking with Transformers

Vinaya James
REVA Academy for Corporate Excellence (RACE), REVA University, Bangalore, India
vinayajames.AI01@race.reva.edu.in

Akshay Kulkarni
REVA Academy for Corporate Excellence (RACE), REVA University, Bangalore, India
akshaykulkarni@race.reva.edu.in

Rashmi Agarwal
REVA Academy for Corporate Excellence (RACE), REVA University, Bangalore, India
rashmi.agarwal@reva.edu.in

Abstract*— The study shown in this paper helps the human resource domain eliminate the time-consuming recruitment process task. Screening resume is the most critical and challenging task for human resource personnel. Natural Language Processing (NLP) techniques are the computer's ability to understand spoken/written language. Now a day's, online recruitment platform is more vigorous along with consultancies. A single job opening will get hundreds of applications. To discover the finest candidate for the position, Human Resource (HR) employees devote extra time to the candidate selection process. Most of the time, shortlisting the best fit for the job is time-consuming and finding an apt person is hectic. The proposed study helps to shortlist the candidates with a better match for the job based on the skills provided in the resume. As it is an automated process, the candidate's personalized favor and soft skills are not affected by the hiring process. The Sentence-BERT (SBERT) network is a Siamese and triplet network-based variant of the Bidirectional Encoder Representations from Transformers (BERT) architecture, which may generate semantically significant sentence embeddings. An end-to-end tool for the HR domain, which takes hundreds of resumes along with required skills for the job as input and provides the better-ranked candidate fit for the job as output. The SBERT is compared with BERT and proved that it is superior to BERT.

***Keywords*— Natural Language Processing, Sentence-BERT, Automatic Recruitment Process, Sentence Embedding.**

I. INTRODUCTION

In a business or organization, it is indeed critical to make the proper hiring decisions for particular positions for Human Resources Manager or Head-hunter [1]. Recruitment tools like "LinkedIn" and "Monster" search for skills and identify candidates who are qualified for open positions. The list of resume search results may be lengthy. All resumes should be manually reviewed to identify possible applicants. Especially, large companies like "Google" frequently receive hundreds of thousands of resumes each year for job applications. As a result, automation is introduced to make the work easy with time-saving.

The introduction of NLP simplifies the process of an automatic sentence or test recognition from papers or resumes [2]. In general vector, representations have become an ever-existing entity in the NLP context. Word embedding properties have seen major developments in the recent past, such as the superposition of word contexts [3], pointwise linkages to shared data values between related words, and linear substructures [4]. However, locating sentences that are comparable based on context, meaning, subject, etc. is a problem in naturally occurring language processing, which leads to the issue of statement integration. The text can be organized, produced, processed, etc., and based on this, it converts the vectors of the words into a vector representation of the sentence. In contrast, an embedding definition is a numerical representation of a word, phrase, sentence, or longer natural language utterance in a particular space. Additionally, word embeddings distributed semantic vector

representations of words have drawn a lot of interest in recent years and have undergone a lot of changes [5]. The word embeddings were developed by analyzing the word distribution among arbitrary text data, and because they are familiar with word semantics, they are a crucial part of many semantic similarity algorithms.

Compared to word embeddings, sentences encompass a higher level of information: the complete semantics and syntactic declarations of the sentence. The word2vec model, which is one of the most popular models, employs neural networks to create word embeddings [6]. Recent transformer-based models' pre-trained word embeddings provided cutting-edge outcomes in a variety of NLP tasks, including semantic similarity. Based on this, modern language models like RoBERTa, BERT [7][8], and ALBERT use transformers to traverse the underlying corpus in both directions and create vector representations of text data. In 2009, the BERT transformer model outperformed the best results available for different NLP tasks, including semantic similarity. The BERT model is typically tuned to serve a specific NLP task using labeled training data. Multilingual BERT is enhanced by the alignment method suggested by Cao et al. [9]. Thus, the BERT models produce bilingual illustrations of words that consider the word's context in both directions.

Performing sentence pair regression, like clustering and semantic search [10], can be time-consuming with BERT due to the large corpus size. Converting a sentence into a vector that encodes the semantic meaning of the sentence is one efficient technique to address this issue. Transformers are the guide to a state-of-art model, and the next section discusses more works accomplished on the NLP concept.

II. State Of Art

As sentence embeddings are more realistic representations of text, recent papers on sentence embedding transformers have focused on the mechanics of recognition. Fast Text is a model that incorporates a character-based n-gram model in addition to the established word embedding techniques Word2Vec and GloVe [11]. This makes it possible to calculate word embeddings from the vocabulary.

Vaswani et al. [12] suggested the Transformer, a self-attention network, as a remedy for the neural sequence-to-sequence issue. When a self-attention network is used to visualize a phrase, each word is represented by a scaled sum of all the other words in the phrase. Liu et al. [13] used a sentence's inner attention to show that self-attention pooling existed before self-attention networks. By generalizing scalar attention to vectors, Cho et al. [14] devised a fine-grained attention approach for neural machine translation. Natural Language Inference (NLI) and other supervised transfer tasks can help complex phrase encoders that are often pre-trained like language models. The Universal Sentence Encoder improves unsupervised learning by training a transformer network on the Stanford NLI (SNLI) dataset. According to Giorgiet al. [15], the task that sentence embeddings are trained on greatly influences their quality. The SNLI datasets, according to Conneau et al. [16], are appropriate for training sentence embeddings.

Therefore, to successfully develop the Chinese Depth Approximation Networking (DAN) and transformers, Parameswa et al. [17] offer an approach that uses Internet responses. Naseem et al. [18] fed the static word embeddings GloVe through the deep neural networks and carried out a controlled neural interpretation operation to obtain the perspective encoding. The Remiers et al. [19] ELMo model is a method for a deeper summarized description. The internal states of words are used to teach a deep bidirectional Language Model (biLM) that has already been trained on a large text corpus. Pre-trained language models of the fine-tuning variety have unfrozen pretrained parameters that can be changed for a new assignment. The text presentation is no longer extracted using this approach. Two significant tasks were introduced by Devlin et al. [20] in their autoencoding pre-trained language model BERT, a deep bidirectional Transformers model: Mask Language Model (MLM) and Next Sentence Prediction (NSP). During the tuning stage, the only activities that deviate from one another are those at the input and output layers. Dai, Zihang, et al. [21] proposed the Transformer-XL generalized autoregressive pre-trained language model, which can learn bidirectional contexts by maximizing expected likelihood across all possible factorization order permutations. Additionally, two groups of the pre-trained language model can be identified based on the pre-training procedure: Auto Encoding (AE) and Auto-Regressive (AR). AR

language models like ELMo and XLNet aim to estimate the probability distribution of a text corpus by employing an autoregressive model. However, AE language models like BERT and its variations like RoBERTa by Liu, Yinhan, et al. [22] seek to recreate the original data from corrupted input without resorting to explicit density estimates.

One major problem with BERT-type models is the introduction of fictitious symbols like MASK during pretraining, despite their complete absence from the final output text. The pre-trained model effectively illustrates how words or phrases link to one another and access various data. Thus, embedding-based key phrase extraction has recently demonstrated strong performance. Lee et al. [23] suggested a Deep Belief Network (DBN) to model the hierarchical relationship between key phrase embeddings. It is simple to distinguish the target document from others using this strategy. Reference Vector Algorithm (RVA) by Papagiannopoulou et al. [24] is a key phrase extraction technique that uses local word vectors as a guiding principle, employing an average of the embeddings trained on distinct files using GloVe as the reference vector for all candidate phrases. The score used to rank the candidate key phrases is then determined by calculating how closely the embeddings of each candidate key phrase match those of the reference vector.

Bennani-Somerset et al. proposed EmbedRank [25] based on the cosine similarity between the candidate key phrase's embeddings and the document's sentence embeddings. EmbedRank creates a document representation using the phrase embedding models Doc2Vec and Sent2Vec. They employ Maximal Marginal Relevance (MMR) to further broaden the keyword's applicability. Pre-trained language models, in particular, BERT, have recently gained popularity and significantly enhanced performance for several NLP applications. Pre-trained BERT and its variations have mostly proved successful in the English language. A language-specific model might be retrained using the BERT architecture for other languages, or one could employ pre-trained multilingual BERT-based models.

III. Problem Definition

Identifying a qualified candidate for the job is a complex undertaking. Typically, manual processes are used in the traditional hiring process. The HR department's qualified recruiters and other significant resources are needed for the manual recruitment process. Businesses sometimes receive a substantial volume of resumes for each job opening, some of which may not even be suitable for the position. Additionally, these hiring procedures take a lot of time and effort to discover qualified applicants for open positions.

Therefore, manually selecting the most pertinent applicants from a lengthy list of potential candidates is difficult. Numerous recent research has focused on the drawbacks of the manual hiring process. Dealing with resumes in the advertising of job specifications and hiring procedures. Selecting people who fit a given job profile is a vital task for most firms. As online hiring becomes more common, traditional hiring methods become less successful.

Recruiting the most pertinent multilingual candidates through the manual hiring process is one of the most critical issues in multilingual job offers and resumes.

The proposed model investigates developing a resume shortlisting and ranking with Transformers to address the difficulty of selecting the right candidate out of two hundred resumes. An automated recruiting system is essential to make it easier for job seekers to access recruitment opportunities and to minimize the amount of human effort involved in the hiring process

.

The main goal is to improve the current resume ranking scheme and make it more adaptable for both parties.

1. *Those who were hired as candidates:* Candidates who have recently graduated and are looking for a job. A significant portion of those applicants is so desperate that they are willing to work in any position unrelated to their skill set and abilities.
2. *The client organization that recruits the applicants:* A job recruiter's objective is to select the top candidate from all qualified applicants based on the Job Description (JD). The process takes time for both the individual and the business. With an automatic resume sorting system, the business may produce the finest candidate list possible based on the limits and

requirements they gave for that particular role. Since the appropriate person will be hired for the position, this hiring method will benefit both the candidate and the organization. Therefore, neither the client firm nor the hired candidate would have any regrets.

The three objectives of this study are:

1. *Collect the resumes as per the defined JD*: The resumes are collected by HR from online job platforms, referrals from existing employees, and third-party consultancies. But getting the exact JD-related resumes are challenging. For this model two hundred resumes have been collected.
2. *Build a custom algorithm to shortlist the resume as per the JD given:* The skills are extracted to a *pandas* data frame with the help of *ResumeParser*. The resume matches will be shortlisted and added to a list based on the JD that HR has provided.
3. *Create a ranking algorithm to get the best out of shortlisted resumes*: JDs and candidates' skill sets are compared, and the most suitable individual with the necessary skills is shortlisted. The model is trained using SBERT and BERT model encodings, and the top N resumes are sorted according to their cosine similarity to sentences and words, respectively.

IV. Proposed Method

Finding the "right" applicant for a position has never been simple. In addition to having the required training and work experience, a prospective employee typically needs to fit in with the current team and support the company's vision. A new dilemma has emerged with the rise of internet job boards and globalization. Today's recruiting professionals frequently need to analyze hundreds of online profiles and resumes only to pick whom to approach due to how simple it has become to build an online profile and apply to a position with a few clicks.

It is not surprising that various technology solutions have been offered to aid recruiters in addressing the problem of candidate screening because automating the shortlisting of candidates can lead to lower costs and higher recruiter productivity. To rank the resume successfully, an efficient context test-based embedding is needed. To achieve textual similarity, transfer-based models like BERT and XLNet are used. In addition, an efficient pre-trained model is required due to its poor accuracy. SBERT, a version of the BERT network that can produce semantically significant sentence embeddings by combining Siamese and triplet networks, is employed in this way [26]. The suggested architecture is depicted in Fig. 1.

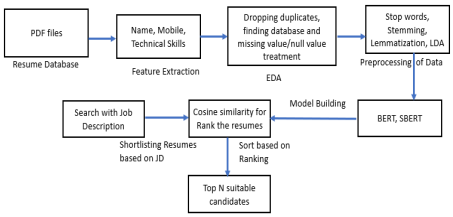


Fig. 1. Proposed Model

The number of resumes is the input used in this proposed model methodology to shortlist the most suitable individuals. *ResumeParser* is used to extract the required details of jobseekers, like Name, Mobile Number, Email Address, and Skills. Exploratory Data Analysis (EDA) is used to remove duplicates, locate databases, and predict and remove missing or null values from text or sentences in resumes. The next step is data pre-processing. At this step, improper words are eliminated, the text is normalized, and the words are prepared for further processing with stop words, Stemming, Lemmatization, and Latent Dirichlet Allocation (LDA). Making vector representations of all words and documents and collectively embedding them in a common vector space is the first step in the extraction of relevant skills and expertise. Then SBERT and BERT are used to create a model.

The produced embedding vectors can be subjected to the Semantic Textual Similarity (STS) task employing metrics such as the Manhattan distance, Cosine similarity, and Euclidean distance. BERT, Siamese networks, and Pooling layer are the three key principles that the SBERT architecture employs. Using a pre-training method called MLM, which typically masks a portion of the tokens in a sentence and predicts them based on their context. BERT is a deep bidirectional transformer because MLM predicts that tokens will approach it from both directions. This is different from traditional left-to-right

models, which only work one way. The JD is then matched with terms described in skills to get a cosine similarity score that may be used to rank and shortlist candidates appropriately in the HR field.

V. MODELING

The term "dictionary" refers to a list of distinct words for each category of documents after the stop words have been eliminated and the words have been stemmed. These stemmed nonstop words can easily address several fields in a database by adding field names to them. Both words with several grammatical forms and words with similar meanings derived from them are handled by an algorithm. Taking all these stages into account, the techniques employed in the example are:

- *Step 1:* Divide the text into words.
- *Step 2:* Eliminate all punctuation and symbols and, if desired, lowercase all words.
- *Step 3*: Eliminate the stop words.
- *Step 4:* Use the Snowball Stemming Algorithm to stem the words.
- *Step 5:* Add parenthesis to each word before adding the field names (if appropriate).

Two hundred resumes are gathered from various sources as part of the data collection process. All the necessary features, including Name, Email address, Mobile Number, and abilities, are retrieved with the aid of *ResumeParser*. A model for BERT and SBERT is constructed based on the JD following EDA and data preprocessing. The *N* number of resumes with the highest ranking is listed using the cosine similarity between the skill set and required JD. The proposed software design is described in Fig. 2.

SBERT for the STS task permits two steps in the prediction of similarity:

- *Step 1:* First, using a sentence encoder, obtain sentence embeddings for each sentence.
- *Step 2:* Next, as the model-predicted similarity SBERT and BERT, compute the cosine similarity between the two embeddings of the input sentence. SBERT used the bert-base-nli-mean-tokens model and is compared to the BERT bert-base-uncased model.

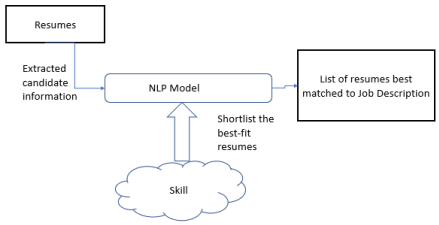


Fig. 2. Software Design

VI. ANALYSIS AND RESULTS

Sentence embedding that outperforms the Classical Least Squares (CLS) vector is obtained by the average of the SBERT context embedding's one or two layers. The average of the last two layers of SBERT, denoted by *last2avg*, consistently yields better results than the average of the last layer of BERT. Since unsupervised learning methods like topic modeling do not guarantee that their results can be understood, correlation metrics have gained popularity among text-mining experts.

The degree of semantic similarity among top-ranking terms in each topic is measured by correlativity. It determines the co-occurrence scores of words in the modeled documents. As coherence also works with syntactic information with the aid of a sliding window that traverses across the corpus and checks occurrences. The notion behind coherence calculation is strongly related to embedding representations of text. Different methods can be used to calculate the correlation. SBERT gives a better solution than BERT when a comparison of top ten ranked resumes based on JD.

Table 1 shows the consistency and alignment of different sentence embedding models (BERT and SBERT) and their averaged STS results. Among BERT and SBERT, models with superior alignment and homogeneity outperform in comparison with the models which do not have alignment and homogeneity. Scores for each pair of sentences are calculated by applying the cosine similarity scoring function to the sentence vectors. The SBERT method is used to arrange the sentences to maximize the sum of their similarities. The findings of this method show that the SBERT method always performs better than the BERT method.

TABLE 1. COHERENCE VALUE COMPARISON FOR BERT AND SBERT

Data Set	*Model*	*Correlation value for Similarity*
STS1	SBERT	0.42649
	BERT	0.194206
STS2	SBERT	0.378602
	BERT	0.119996
STS3	SBERT	0.377433
	BERT	0.047986
STS4	SBERT	0.374302
	BERT	0.156387
STS5	SBERT	0.373682
	BERT	0.182748
STS6	SBERT	0.373111
	BERT	0.048559

Based on Table 1, the graphical representation is provided in Fig. 3.

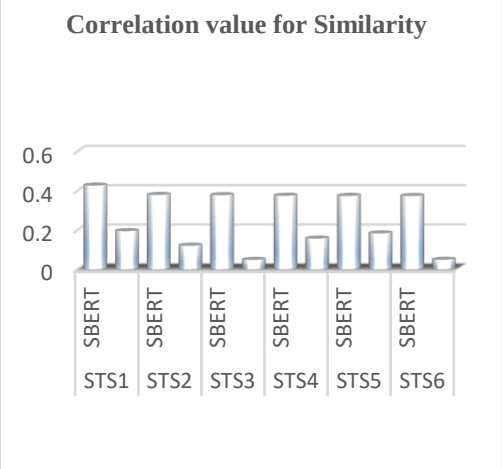


Fig. 3. Performance analysis of SBERT with Coherence

Table 1 analysis reveals that the SBERT performs better than the BERT in terms of correlation. Compared to SBERT, the similarity variation is lower, as depicted in Fig. 3. Resumes in the STS3 dataset exhibit a relatively low similarity variance. As a result, the SBERT executes the ideal sentence embedding to rank the candidate's information with the JD of the job provider in the minimum similarity variance.

VII. CONCLUSION

The proposed SBERT transformer helps recruiters screen resumes more quickly and effectively, cutting the cost of hiring. Thus, the company will then have access to a potential applicant who will be successfully placed in a business that appreciates the candidate's skills and competencies. These days, many applicants submit applications for interviews. Every interview should include a review of resumes. Going through each resume one by one is not a good idea. It becomes quite difficult for the HR team to narrow down candidates for the following stage of the hiring process. The SBERT streamlines the process by summarizing resumes and classifying them by how closely they match the organization's necessary skills and requirements.

The proposed method evaluates candidates' skills and ranks them by the JD and skill requirements of the employing organization. A summary of their resume is supplied to provide a fast overview of each candidate's qualifications. One of the main issues is when a candidate lists skills for which they have no experience because the model focuses on the skill set listed on the resume submitted by the candidate. Artificial Intelligence techniques or any other effective sentence embedding transformers can be used for further improvement.

REFERENCES

[1] Siddique, C. M. "Job analysis: a strategic human resource management practice." The International Journal of Human Resource Management 15.1 (2004): 219-244.

[2] Sanabria, Ramon, et al. "How2: a large-scale dataset for multimodal language understanding." arXiv preprint arXiv:1811.00347 (2018).

[3] Arora, Sanjeev, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. "A latent variable model approach to pmibased word embeddings." Transactions of the Association for Computational Linguistics 4 (2016): 385-399.

[4] Rieck, Bastian, and Heike Leitte."Persistent homology for the evaluation of dimensionality reduction schemes."Computer Graphics Forum.Vol. 34.No. 3. 2015.

[5] Stein, R. A., Jaques, P. A., & Valiati, J. F. (2019). An analysis of hierarchical text classification using word embeddings. Information Sciences, 471, 216-232.

[6] Mishra, Mridul K., and JaydeepViradiya."Survey of Sentence Embedding Methods." International Journal of Applied Science and Computations 6.3 (2019): 592-592.

[7] Chernyavskiy, Anton, Dmitry Ilvovsky, and PreslavNakov. "Transformers:"The End of History" for Natural Language Processing?." Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Cham, 2021.

[8] P. S. Suryadjaja and R. Mandala, "Improving the Performance of the Extractive Text Summarization

by a Novel Topic Modeling and Sentence Embedding Technique using SBERT," 2021 8th International Conference on Advanced Informatics: Concepts, Theory and Applications (ICAICTA), 2021, pp. 1-6, doi: 10.1109/ICAICTA53211.2021.9640295

[9] Cao, Steven, Nikita Kitaev, and Dan Klein. "Multilingual alignment of contextual word representations." arXiv preprint arXiv:2002.03518 (2020).

[10] H. Choi, J. Kim, S. Joe and Y. Gwon, "Evaluation of BERT and ALBERT Sentence Embedding Performance on Downstream NLP Tasks," 2020 25th International Conference on Pattern Recognition (ICPR), 2021, pp. 5482-5487, doi: 10.1109/ICPR48806.2021.9412102.

[11] Dharma, EDDY MUNTINA, et al. "The accuracy comparison among word2vec, glove, and fasttext towards convolution neural network (CNN) text classification." J TheorApplInfTechnol 100.2 (2022): 31.

[12] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).

[13] González, José Ángel, Lluís-F.Hurtado, and FerranPla. "Transformer-based contextualization of pre-trained word embeddings for irony detection in Twitter." Information Processing & Management 57.4 (2020): 102262.

[14] Heeyoul Choi, Kyunghyun Cho, Yoshua Bengio,Fine-grained attention mechanism for neural machine translation,Neurocomputing,Volume 284,2018,Pages 171-176,ISSN 0925-2312.

[15] Giorgi, John, et al. "Declutr: Deep contrastive learning for unsupervised textual representations." arXiv preprint arXiv:2006.03659 (2020).

[16] Conneau, Alexis, et al. "Supervised learning of universal sentence representations from natural language inference data." arXiv preprint arXiv:1705.02364 (2017).

[17] Parameswaran, P., Trotman, A., Liesaputra, V., & Eyers, D. (2021). Detecting the target of sarcasm is hard: Really??. Information Processing & Management, 58(4), 102599.

[18] U. Naseem and K. Musial, "DICE: Deep Intelligent Contextual Embedding for Twitter Sentiment Analysis," 2019 International Conference on Document Analysis and Recognition (ICDAR), 2019, pp. 953-958, doi: 10.1109/ICDAR.2019.00157.

[19] Reimers, Nils, and IrynaGurevych. "Alternative weighting schemes for elmoembeddings." arXiv preprint arXiv:1904.02954 (2019).

[20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

[21] Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., & Salakhutdinov, R. (2019). Transformer-xl: Attentive language models beyond a fixed-length context. arXiv preprint arXiv:1901.02860..

[22] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

[23] Jo, T., & Lee, J. H. (2015). Latent keyphrase extraction using deep belief networks. International Journal of Fuzzy Logic and Intelligent Systems, 15(3), 153-158.

[24] Papagiannopoulou, Eirini, and GrigoriosTsoumakas. "Local word vectors guiding keyphrase extraction." Information Processing & Management 54.6 (2018): 888-902.

[25] Kamil Bennani-Smires, Claudiu Musat, Andreea Hossmann, Michael Baeriswyl, and Martin Jaggi. 2018. Simple Unsupervised Keyphrase Extraction using Sentence Embeddings. In Proceedings of the 22nd Conference on Computational Natural Language Learning, pages 221–229, Brussels, Belgium. Association for Computational Linguistics.

[26] Reimers, N., & Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084.