A Machine Learning Approach for Screening Individual's Job Profile Using Convolutional Neural Network

M.F. Mridha*, Rabeya Basri*, Muhammad Mostafa Monowar†, Md. Abdul Hamid†
*Department of Computer Science & Engineering, Bangladesh University of Business & Technology, Dhaka, Bangladesh
{firoz, rabeya.b}@bubt.edu.bd
†Department of Information Technology, Faculty of Computing & Information Technology, King Abdulaziz University,
Jeddah 21589, Kingdom of Saudi Arabia
{mmonowar, mabdulhamid1}@kau.edu.sa

2021 International Conference on Science & Contemporary Technologies (ICSCT) | 978-1-6654-2132-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICSCT53883.2021.9642652

***Abstract*—Pursuing human things through machines is the current hype in the information technology domain, especially in Artificial Intelligence (AI). As a part of AI, machine learning has grabbed the vogue by spreading its branches to various dimensions and degrees. Deep learning is one of the latest areas, being applied in the massive processing-related fields. In the area of NLP, various tools and technologies have been groomed and faded for their comparative pros and cons. Candidate's job profiles can be screened by analyzing their CV/Resume for selection through the most intellectual and advanced process according to the HR requirement. Each profile is read one by one and marked whether selected for the preliminary stage or not. In this work, we aim to automate this screening process through a depth analysis of NLP to revolutionize the entire recruitment process by making employer tasks easier than ever and we are capable of ranking the CVs of the individuals by matching the total fields in the CVs with the required fields. Moreover, our classification model has provided an accuracy of 74% in terms of precision, recall and f1-score for the BDJOBS site that outperforms other existing methods.**

Keywords—Job Profile Screening, CV/Resume, Artificial Intelligence, Machine Learning, NLP

I. Introduction

The most common traditional job trend that was followed by people, where candidates used to settle for one or two jobs at most in their entire professional life as well as the employers used to take pride that they have employees who worked for them for over two or three decades. However, in this modern era, due to rapid changes in technology, those scenarios are invalid for most employees and employers as well. Switching job is one of the most common practices of our time. This brings a huge headache for the employer to find the right candidate after sorting through a large amount of CV/Resume. 52% of talent acquisition leaders say that ranking the job profile of a candidate as a top is the most critical task in a recruitment process from a vast amount of qualified applicants that will depend on a recruiter's efficiency to automate their workflow [1] in the future. Our research aim is to resolve this difficult issue and simplify the process.

In this modern era, we have a mammoth amount of data to deal with in a short time to keep up with the pace of our daily business. This leads to significant challenges of maintaining accuracy as well as timely processing of data. Artificial Intelligence (AI) can be thought of as an effecting solution to deal these challenges of our daily business life [2]. Machine Learning (ML) is the branch of AI, which helps a machine to be trained for a problem and provides a solution based on the historical data. Initially, the model is dealt with the known parameters to successfully run the algorithm to get the desired output (learning) that will be matched with the known answer [3]. Massive amounts of input data will help the system to learn and process higher computational decisions to that extent. Besides, machine learning has diverse application domains, including manufacturing, retailing, healthcare, travel and hospitality, HR department, finance, power supply optimization, and so on [4].

The main task of the HR department of a company, is to sort out qualified applicants with required degree of experience, education, and skills to fill a particular position. Thus, they need to go through myriad CVs manually to find potential candidates. At the same time, there are some techniques to search candidates' profiles by some keywords that do not apply any intelligence. Moreover, this is immensely time-consuming and often failed to select correct candidates in time. Thus, it raises the cost and the causes the business expansion stumbled. To mitigate this problem, some job sites usually search for CVs with keywords, which also have some limitations since most of the search is based on a normal text-matching algorithm [5]. Here, we like to propose a solution using a Convolutional Neural Network (CNN) to train a machine with a large number of CV's corresponding to a large number of job descriptions which were selected or rejected, with the aim of determining the suitable CV based on the input job description [6]. Organizations collect a huge number of CVs against each requirement and primarily process each CV to classify each of them as selected or not in such a format that would be easier to be fed into ML models. Amongst several traditional ML algorithms like Naive Bayes, Random Forest, Logistic Regression [7], etc. CNN has shown the best result for our research.

The main contribution of our paper is as follows:

- Presenting a CNN-based solution to screen individual's job profile based on their CVs to classify them as selected or rejected candidates.
- Preparing the data in an acceptable manner to feed into Machine Learning (ML) algorithms, as these

algorithms cannot go with textual data [8], but with the cleaning and de-erotization process [9].

- Representing how a realistic model can help the system designers and programmers to design a job profile searching system using AI and ML after analyzing CVs.

The rest of the paper is organized as: Section II provides a brief overview on the previous related tasks; Section III describes our research methodology, where we proposed possible architecture of the system and a working model; Section IV illustrates the experimental results with data preparation; finally, Section V concludes the paper with future scopes.

II. Related Work

In a very layman manner, Machine Learning (ML) can be defined as an advanced, automatic learning process to learn from computers without any labor effort [13]. The learning process starts with training the quality data by machines using different machine learning algorithms. These algorithms can be selected based on the types of data to be trained and on the output that we want [14]. So far keyword search-based job profile screening has been developed and tested. Few works like at goarya.com [15], harver.com [16] and ideal.com [17] has designed and developed their AI-based product but those are also based on keyword and few terminologies. Some efforts [10], [11] and [12] exploit CNN for text classification in general but not specific to job file screening. However, we are highly inspired by those works to propose our method.

Converting human efforts into mechanic intelligence is a novel and new idea in this arena. We pursue to utilize the human efforts that have already been invested and stored in the flat file as we mentioned earlier. Our thoughts give us a view of collecting the job description/advertisement in various media like internet/ job portals social media and newspaper as well as the response of those in terms of CV/Resume, where HR executives manually worked and applied their labor to screening the proper candidates and stamp them as selected or rejected. Taking those CVs, we will process them into mechanic intelligence, which will guide us to screen the job description in the future.

III. Methodology

A. Convolutional Neural Network (CNN)

CNN can be defined as a deep, feed-forward artificial neural network, where neurons are connected in an acyclic manner by using a variation of multilayer perceptron with minimal preprocessing. CNN is basically applicable to computer vision tasks, but recently it has widely been applied to various NLP tasks for its outstanding performance. Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano. It was developed to make the experimentation fast with less effort [18].

B. CNN Application to NLP

Most of the inputs of NLP tasks are sentences or documents that can be represented as a matrix instead of pixels. Each row of the matrix corresponds to a token, typically a character or particularly to a vector that represents a word. These vectors are capable of word embedding (low-dimensional representations) such as word2vec or GloVe. Additionally, these vectors can index the word into a vocabulary. For a 10 word sentence using a 100-dimensional embedding, we would have a 10×100 matrix as our input. In the vision, our filters normally move slide over the local patches of an image, but in case of NLP, these filters slide over the full rows of the matrix of the input words. Thus, the "width" of these filters is usually as same as the width of the input matrix. The height, or region size, may vary, but sliding windows over 2-5 words at a time is typical [19].

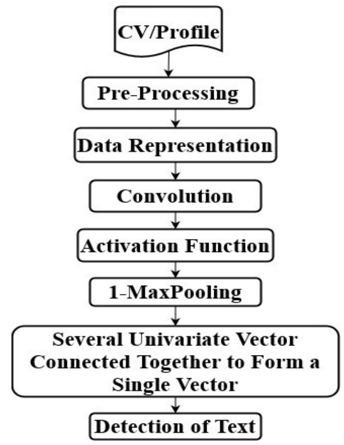

Fig. 1. The flow diagram illustrates the basic CNN applications for NLP processing to screen the correct CVs.

C. Proposed Architecture

Our proposed architecture uses the most recent available technologies in the cloud infrastructure [20]. We use Apache Gobblin [21] for the workflow, data lake for storage, data mining, Apache Hadoop [22] for task distribution. We also want to take advantage of cloud machine learning engines out there which use a convolutional neural network to train the system. Here is the proposed Architecture that we can attempt on.

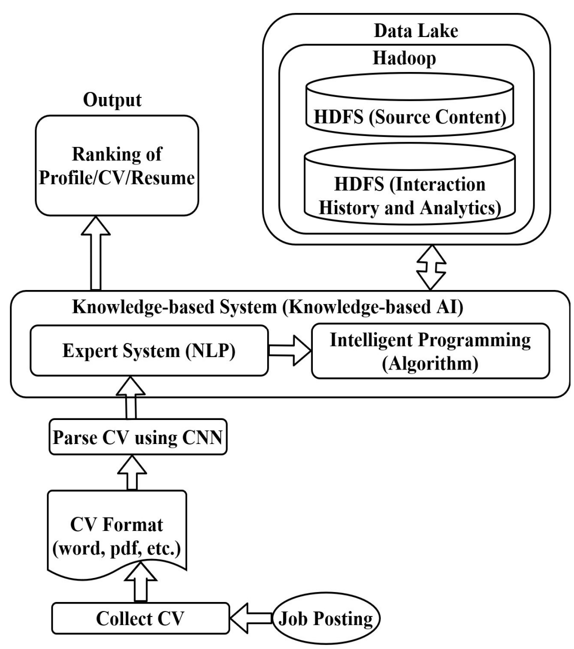

Fig. 2. This workflow diagram represents the basic architecture of our system including the flow of tasks to be performed.

TABLE I. THE TABLE REPRESENTS THE COLEECTED DATA FROM DIFFERENT KNOWLEDGE-BASES FOR IT JOBS.

Java	***Php***	***Python***	***Database***	***Database Administration***	***Data Warehousing***	***Reporting***	***Microsoft Technology***	***Oracle***	***Frontend***	***Code Repository***
object oriented	php	django	mysql	DBA	ETL	Crystal report	ASP.net	Oracle DB	UX	Github
hibernet	laravel	python	rdbms	database design	Powercube	Jasper report	C#.net	Mysql	HTML5	TFS
design patterns	html	flask	mongo db	RDBMS	OLAP		VB.net	PI/Sql	CSS3	Gitlab
spring	mvc	pandas	no sql	SQL	MarkLogic		MS SQL		Jquery	Bitbucket
spring MVC	zend	numpy	sql	recovery	Oracle DWH		Xaml		Angular.js	SVN
java	codeigniter	scikitlearn	hql	failover	Amazon RedShift		ADO.NET		Ajax	Subversion
java ee	Cake Php	sklearn	postgres	DB2	informatica		MS Silverlight		Git	CVS
jsf	LAMP	matplotlib	ms-sql	Oracle	SSIS		MS Azure		Bootstrap	Mercurial
jsp	WAMP	scipy	oracle	MySql	OracleWarehouse Builder OWB		Ms .Net Framework		React.js	Monotone
spring rest api	XAMPP	tensorflow	pi/spl	SqlServer	Oracle Data integrator		C#		Javascript	Bazaar
jasper report	Symphony	keras	backend	database queries	Oracle Golden Gate				Vue.Js	VSTS
spring security	Yii	pytorch	back-end	Oracle RAC	IBM DataStage		c.net		Knockout.js	Perforce Helix Core
	Zend Framework		Administration	Oracle data guard	IBM Ascential		C++		WebRx	IBM Rational ClearCase
	Drupal		OLTP	Oracle cloud Admin	BusinessObjects Data Integrator				Ext.Js	Revision Control System
	Wordpress			Oracle Apps DBA	SAS Data Integrator				D3.JS	VSS
	Zoomla			T-SQL	Dimensional Modeling				Sencha	CA Harvest Software
	Phalcon			PI/SQL	DW architecture				ember.js	PVCS

We also need a knowledge-based pool to train the model, which can be served for multiple purposes to the external or internal audiences [23]. For example, in our case, we determine the keywords for the type of Job Profile.

In the example below, we can see a knowledge base for IT jobs. Below we are showing the high-level dataflow diagram of our system which shows our contribution to this newly proposed system.

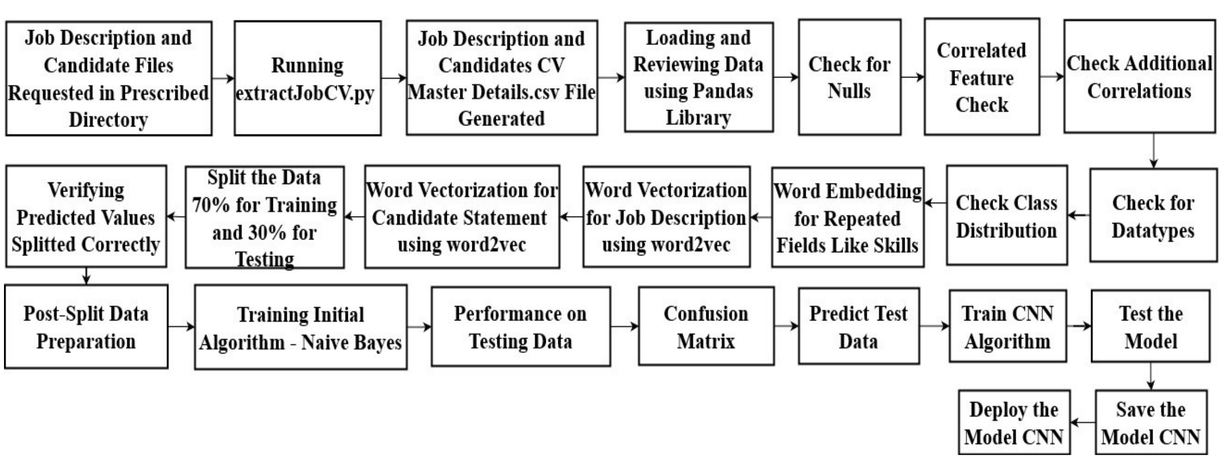

Fig. 3. This flow diagram illustrates the high level data flow used for CV ranking.

We are introducing class distribution to check for data type and word vectorization to split the data. After the data split, we are training the initial data with Naïve Bayes and generation the confusion matrix. After finalizing the data, we are training the CNN and generating the model. After testing the model, we can use it for final data extraction in the production level CV/Resume/Profile.

D. Proposed Work Model

Our first and foremost task is to extract the Job Description. We like to take the most key values from it as described in the previous chapters. To do that we like to propose the following steps:

1) Task 1: Extract Job Description

Step 1: Identify Job Title / Designation
Step 2: Identify Core Business Area [Check with Knowledgebase for relevant words]
Step 3: Identify Experience (in years)
Step 4: Identify Hard Skill Requirements [Check with Knowledgebase for relevant words]
Step 5: Identify Educational requirements [Check with Knowledgebase for relevant words]
Step 6: Identify Soft Skills Requirements [Check with Knowledgebase for relevant words]
Step 7: Identify Geo-Location/Address/ Place of work

Now, the challenging part is to extract the candidate's Profile/CV/Resume. Here the Profile/CV/Resume are segmented in the several sections. Therefore more steps are required the extract the desired fields and their attrie proposing the following:

2) Task 2: Extract Profile/CV/Resume

Step 1: Identify Personal Data (and mask it)
Step 2: Identify Core Experience Area [Check with Knowledgebase for relevant words]
Step 3: Identify Experience (in years)
3. a: Total experience
3. b: Experience in each job
3. c: Employer reputation [Check with Knowledgebase for relevant words]
Step 4: Identify Hard Skill [Check with Knowledgebase for relevant words]
4. a. 1: by core experience + years
4. b. 2: by relevant experience + years
Step 5: Identify Educational requirements [Check with Knowledgebase for relevant words]
5. a: University/ College/ Degree
5. b: Professional/continuous education Certification
Step 6: Identify Soft Skills Requirements [Check with Knowledgebase for relevant words]

IV. Data Processing

A. Preparing the data to feed into ML algorithms

A master's details relationship has been established between these two types of records.

1) Job Description: This entity contains a unique job id for every job advertised for.

2) CV/RESUME/PROFILE: This is a collection of CVs having unique CVID and JOBID as child.

The output file should look like as the following table.

TABLE III. THE TABLE PROVIDES THE JOB DESCRIPTIONS AND APPLICANT INFORMATION USED FOR OUR EXPERIMENTATION.

Job Description								***Applicant Information***					
Position Advertised For	Job Responsibility	Employment Status	Educational Requirement	Required Skills and Experience	Job Location	Salary	Other Benefits	Position Applied For	Personal Details	Skill Details	Experience Declaration	Educational History	Expected Salary
Text	Text	Text	Text	Text	Text	Number	Text	Text	Text	Text	Text	Text	Number

Common information are extracted from different formats of job descriptions and CV/Resume/Profile. Various types of job descriptions come in a different format depending on a lot of parameters like local or international, specified position, or normal position. Some sample screenshots are coming in the preceding section to comprehend the complexity of the job description in various formats. Required information are extracted into a structured file preferably a CSV file.

B. Data Preparation

Fields taken for experimentation were identified and analyzed from a lot of job descriptions and job postings on online job sites and various sources. The most common fields have been taken both for Job Description and Resume. Now, those common fields need to be extracted from the directory structure. The case here, not the one that is a completely unstructured file from, where information needs to be extracted [24]. From the analysis and research, we have done on the various types of job descriptions and Resume/CV formats so that at least we know their common fields, style, and formats they are usually used, though there are some heterogeneous terminologies for the same name.

TABLE II. THE TABLE REPRESENTS A CV MATRIX CONTAINING DIFFERENT FIELDS TO EXPLAIN JOB CATEGORY, SECTOR, APPLICANT'S NAME, ADDRESS, CAREER OBJECTIVES, SKILLS, ETC.

Job Sector	***Category***	***Personal Details***			***Job Profile***			
		Name	***Address***	***Contact Details***	***Career Objective***	***Career Summary***	***Skill Summary***	***Professional History***
IT & Telecom.	Programming	Yes	Yes	Yes	Yes	Yes	Yes	Yes
IT & Telecom.	Management	Yes	Yes	Yes	Yes	Yes	Yes	Yes
IT & Telecom.	Designing	Yes	Yes	Yes	Yes	Yes	Yes	Yes
IT & Telecom.	Quality Assurance	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Garments/Textile	Management	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Garments/Textile	Procurement							
Garments/Textile	Quality Assurance							
Marketing/Sales	Brand Management							
Marketing/Sales	Digital Marketing							
Marketing/Sales	Sales							

TABLE IV. THE TABLE ILLUSTRATES A JOB DESCRIPTION MATRIX AFTER ANALYZING DIFFERENT JOB SITES.

		Job Description					*Requirements*	
Site Name	***URL***	***Position***	***No. of Vacancy***	***Functional Role***	***Responsibilities***	***Employment Status***	***Educational***	***Experience***
BDJOBS	https://www.bdjobs.com/	Yes	Yes	No	Yes	Yes	Yes	Yes
Careerjet	https://www.careerjet.com.bd/	Yes	No	No	Yes	Yes	Yes	Yes
Skilljobs	http://skill.jobs/	Yes	No	No	Yes	Yes	Yes	Yes
Naukri	https://www.naukri.com/	Yes	No	No	Yes	No	No	Yes
Monster	https://www.monster.com/	Yes	No	No	Yes	No	No	No
Timesjob	https://www.timesjobs.com/	Yes	No	Yes	Yes	No	No	No
CareerAge	https://www.careerage.com/	Yes	No	Yes	Yes	No	Yes	Yes
Indeed	https://www.indeed.com/	Yes	No	No	Yes	No	No	No
CareerBuilder	https://www.careerbuilder.com/	Yes	No	No	Yes	No	No	No
Dice	https://www.dice.com/	Yes	No	No	Yes	No	No	No
Idealist	https://www.idealist.org/en/	Yes	No	No	No	No	No	Yes
LinkedIn	https://www.linkedin.com/	Yes	No	No	No	Yes	No	No

C. *Eliminating Heterogeneous Terminology*

As we browsed various types of job description on multiple sites, we observed that similar content or material were placed under different terminologies. We referred it as heterogeneous terminologies. For example, in BDJOBS and Careerjet job site, we observed they used the term "Job Context" to give an overall brief about the job advertised, whereas, in the SkillJobs, Naukri, and Monster job site, they have used "Job Summary" to express similar brief about the job advertised. The candidates used Addresses or Contact Details to describe their contact information. They also used sections like a profile or career summary to give a brief about their professional summary.

To overcome these heterogeneous terminologies, we had to decide and linked similar terminology in advance, so that they can be extracted in a common name and linked to the heterogeneous terminology [25]. Thus, the system can be trained appropriately to determine correct information from a job description and CVs posted by the candidates.

V. EXPERIMENTAL RESULT

To apply the proposed model, we have to annotate the Profile/CV/Resume. Here following fields of a Profile/CV/Resume are annotated: Name, Email, Mobile, [Personal Data- Masked], Address, Sex, Age, Graduation Year, Degree, College Name, Reference, Training And Certification, Professional Membership, Research Publication, LinkedIn Profile, Skills, Total Job Experience, Job1-CompanyName, Job1-Designation, Job1-Experience, Job1-Skills.

A. *Example of CV after being annotated:*

TABLE V. THE TABLE ILLUSTRATES THE SAMPLE ANNOTATION OF THE JOB PROFILE OF AN APPLICANT.

Annotated Field	*Category*	*Annotation by Candidates*
Name	text	Rony Das
Years of Experience	number	6
Technology	text	Java SE, Java EE, SQL and CSS, Java Script, Ajax, jQuery
Education	text	Computer Science
Mobile	number	01810652261
Email	text	ronydas@gmail.com
Gender	text	Male
Location	text	Dhaka
Salary	number	50,000
Reference	text	Mr. Safiur Rahman

B. *Output:*

After the training of the data using CNN, we obtained the following classification report from the system which gives an accuracy measurement of the system:

TABLE VI. THE TABLE PROVIDES THE CLASSIFICATION REPORT TO SHOW THE ACCURACY MEASUREMENT OF THE SYSTEM.

	precision	*recall*	*f1-score*	*support*
0	0.81	0.78	0.79	151
1	0.61	0.65	0.63	80
accuracy			0.74	231
macro avg	0.71	0.72	0.71	231
weighted avg	0.74	0.74	0.74	231

TABLE VII. THE TABLE PROVIDES THE EXPERIMENTAL RESULTS AFTER APPLYING OUR MODEL ON DIFFERENT JOB SITES.

Job Site	*Precision*	*Recall*	*F1-score*
BDJOBS	0.74	0.74	0.74
Careerjet	0.61	0.65	0.61
Skilljobs	0.62	0.67	0.66
Naukri	0.5	0.51	0.5
Monster	0.58	0.67	0.55
Timesjob	0.44	0.45	0.41
CareerAge	0.65	0.7	0.68
Indeed	0.7	0.71	0.71
CareerBuilder	0.68	0.7	0.7
LinkedIn	0.71	0.72	0.71

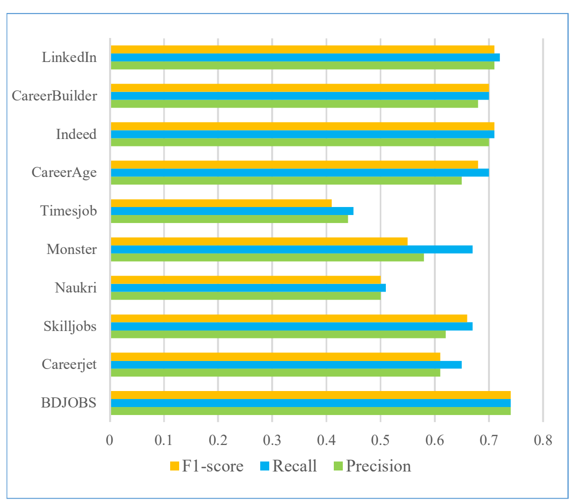

Fig. 4. This precision, recall, and f1-score graph shows the accuracy of the predicted test values of our system on different job sites.

Now, let us examine some real world scenario and three potential CVs with a given job description, presented in Table VIII to match them with the following mandatory fields using our work model.

5+ years' experience with J2EE, Web Logic/Web Sphere and Linux being EEE/CSE graduate having 50,000+ salary and job location in Dhaka.

TABLE VIII. THE TABLE ILLUSTRATES THE PROCESS OF EXAMINING THREE CVS WITH THE MANDATORY KEYWORDS.

Keyword	*Experience*	*Technology*	*Education*	*Salary*	*Location*	*Industry*
CV_1	3	J2EE, Web Logic, Web Sphere, Linux	EEE graduate	50,000	Dhaka	—
CV_2	6	J2EE, Web Logic, Web Sphere, Linux	CSE graduate	60,000	—	—
CV_3	5	J2EE, Oracle, HTML, CSS	EEE graduate	—	Dhaka	IT

Here is the result we get if we ask for ranking of the CVs based the jobs description:

TABLE IX. THE TABLE ILLUSTRATES THE RANKING OF THE JOB PROFILE OF AN APPLICANT AFTER MATCHING THE TOTAL GIVEN FIELDS IN THE CV/RESUME WITH THE REQUIRED FIELDS SET BY THE EMPLOYEER.

	Total Fields in the CV	*No of Matching Required fields*	*No of Matching Non Required fields*	*Ranking*
CV_1	8	3	5	2
CV_2	5	4	1	1
CV_3	9	2	7	3

VI. CONCLUSION

From our model, it can be clearly seen that the system provides the best accuracy of 74% for BDJOBS site compared to other sites and it is capable of ranking the CVs by screening individuals job descriptions based on the matching value of the total fields in the CV with the required fields using some potential keywords. However, the ranking and positive weights are given for a Profile/CV/Resume might be changed based on different companies or employer's preferences. Given the amount and variation in data, it is hard to come up with a suitable solution in a short time. However, given more data and time to train the system may result in desired output. Many enterprises are working on this type of solution and we think our research may help them to get some understanding of which way to move forward. We understand the hurdles especially annotating a huge number of CVs in propose to train the system. In the future, more advanced technology may overcome this issue by automating the process to make the system more robust and less tedious to train the data, which will provide better matching result with the proposed system.

REFERENCES

[1] AI For Recruiting: A Definitive Guide For HR Professionals. Available Online: https://ideal.com/ai-recruiting/.

[2] S. Brindha, K. Prabha, and S. Sukumaran, "A survey on classification techniques for text mining," 3rd International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 1-5, Coimbatore, India, 2016.

[3] H. M. M. Hasan, F. Sanyal, D. Chaki, and M. H. Ali, "An empirical study of important keyword extraction techniques from documents," 1st International Conference on Intelligent Systems and Information Management (ICISIM), pp. 91-94, Aurangabad, 2017.

[4] Machine Learning Applications. Available Online: https://www.stxnext.com/machine-learning-applications-examples-industries.

[5] C. Yin, J. Xiang, H. Zhang, and J. Wang, "A New Classification Method for Short Text Based on SLAS and CART," 1st International Conference on Computational Intelligence Theory, Systems and Applications (CCITSA), pp. 133-135, Ilan, Taiwan, 2015.

[6] Y. Kim, "Convolutional Neural Networks for Sentence Classification," Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751, 2014.

[7] S. Kavitha, S. Varuna, and R. Ramya, "A comparative analysis on linear regression and support vector regression," Online International Conference on Green Engineering and Technologies (IC-GET), pp. 1-5, Coimbatore, 2016.

[8] J. Zhao and Q. Gao, "Annotation and Detection of Emotion in Text-based Dialogue Systems with CNN," arXiv.org, Cornell University, 2017.

[9] M. Pandey and V. Ravi, "Detecting phishing e-mails using text and data mining," International Conference on Computational Intelligence and Computing Research, pp. 1-6, IEEE, Coimbatore, India, 2012.

[10] T. Delteil, "Character-level Convolutional Networks for text Classification," Available Online: https://github.com/ThomasDelteil/TextClassificationCNNs_MXNet.

[11] A. Jacovi, O. S. Shalom, and Y. Goldberg, "Understanding Convolutional Neural Networks for Text Classification," Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, pp. 56-65, 2018.

[12] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent Trends in Deep Learning Based Natural Language Processing," IEEE Computational Intelligence Magazine, vol. 13, pp. 55-75, 2018.

[13] H. Sayoud, "Automatic authorship classification of two ancient books: Quran and Hadith," 11th International Conference on Computer Systems and Applications (AICCSA), pp. 666-671, IEEE/ACS, Doha, Qatar, 2014.

[14] S. Meraji, "How to prepare for FACE 2017? Malaysia's Greatest AI Hackathon," Available Online: https://medium.com/@sinameraji/how-to-prepare-for-face-2017-malaysias-greatest-ai-hackathon-5a43239beb9e.

[15] Recruiting done right the first time. Available Online: https://goarya.com.

[16] Volume hiring starts with Harver. Available Online: https://harver.com.

[17] Accurate, Efficient, and Fair Talent Decisions. Available Online: https://ideal.com.

[18] Keras: The Python Deep Learning library. Available Online: https://keras.io.

[19] Deep Learning for NLP: An Overview of Recent Trends. Available Online: https://medium.com/dair-ai/deep-learning-for-nlp-an-overview-of-recent-trends-d0d8f40a776d.

[20] How Search Engines like Google Retrieve Results: Introduction to Information Extraction using Python and spaCy. Available Online: https://www.analyticsvidhya.com/blog/2019/09/introduction-information-extraction-python-spacy.

[21] Gobblin Documentation. Available Online: https://gobblin.readthedocs.io/en/latest/Gobblin-Architecture.

[22] Apache Hadoop. Available Online: https://hadoop.apache.org.

[23] Ready for ITSM at high velocity? Available Online: https://www.atlassian.com/itsm/knowledge-management/what-is-a-knowledge-base.

[24] W. Yin, K. Kann, M. Yu, and H. Schütze, "Comparative Study of CNN and RNN for Natural Language Processing," 2017.

[25] M. T. Reza and M. S. Zaman, "Analyzing CV/Resume using Natural Language Processing and Machine Learning," Department of Computer Science and Engineering School of Engineering and Computer Science, BRAC University, 2017.