Application of NLP for Information Extraction from Unstructured Documents

Chapter · January 2022

DOI: 10.1007/978-981-16-2126-0_54

Shushanta Pudasaini
Tribhuvan University

Sujan Adhikari
Pokhara University

Application of NLP for Information Extraction from Unstructured Documents

Shushanta Pudasaini[1], Subarna Shakya[2], Sagar lamichhane[3], Sajjan Adhikari[4], Aakash Tamang[5], and Sujan Adhikari[6]

[1] Advanced College of Engineering & Management, Kupondole Road, Lalitpur
shushanta574@gmail.com
[2] Institute of engineering, Tribhuvan University, Pulchowk, Lalitpur
drss@ioe.edu.np
[3] Herald College Kathmandu, Hadigaun Marg, Kathmandu
sagar_lamichhane@yahoo.com
[4] Nagarjuna College of Information Technology, Bangalamukhi, Lalitpur
sajjanadhikari464@gmail.com
[5] Patan College For Professional Studies, Kupondole, Patan
dumjanaakash@gmail.com
[6] NAMI college, Gokarneshwor, Kathmandu
zeradt@gmail.com

Abstract. The world is intrigued by data. In fact, huge capitals are invested to devise means that implements statistics and extract analytics from these sources. However, when we examine the studies performed on applicant tracking systems that retrieve valuable information from candidates' CVs and job-descriptions, they are mostly rule-based and hardly manage to employ contemporary techniques. Even though these documents vary in contents: the structure is almost identical. Accordingly, in this paper, we implement an NLP pipeline for the extraction of such structured information from a wide variety of textual documents. As a reference, textual documents which are used in applicant tracking systems like CV (Curriculum Vitae) and job vacancy information have been considered. The proposed NLP pipeline is built with several NLP techniques like document classification, document segmentation and text extraction. Initially for the classification of textual documents, Support Vector Machines (SVM) and XGBoost algorithms have been implemented. Different segments of the identified document are categorized using NLP techniques such as chunking, regex matching and POS tagging. Relevant information from every segment is further extracted using techniques like Named Entity Recognition (NER), regex matching and pool parsing. Extraction of such structured information from textual documents can help to gain insights and use those insights in document maintenance, document scoring, matching and auto filling forms.

Keywords: Keywords: NLP, information extraction, segmentation, Named Entity Recognition (NER), GaussianNB, SVM, spaCy

1 Introduction

A document consists of multiple information where some of it can be very important and some can be less. Previously datas was collected in unstructured repositories of texts but with the boom of the Internet, most of current data is collected from online platforms [14]. Although we can find data everywhere on the internet, it is humanly impossible to read all of them and extract the required information. In-order to overcome this limitation, the approach of Information extraction has come to great attention among the NLP developers.

Information Extraction is a process of identifying appropriate information from the input document and converting information into representation suitable for storage, process and retrieval via computational methods. The input is a collection of documents like (News Articles, Research Papers, Reports, emails) and extracted information is representation of relevant information from source document according to specified criteria. [13]

Extracting information from a given textual document has been a frequently performed NLP task. Every time, new methods and techniques are implemented to create a model or an approach that would extract only the valuable information from a given corpus. However, documents can never be of a fixed format. It can either be structured or unstructured. Extracting information from unstructured format is more challenging than from the structured one as it can be in any format. Also, it is very important to know the types of information that we are going to extract from the document. The information that is to be extracted can vary from the type of information the document contains.

Since it is a demanding task to identify the important information from all sorts of documents, we decided to target a specific one. The type of document that we are mainly focusing on are CV and information on job vacancy. To make it more precise, for this research purpose, CV and job vacancy details related only to the IT (Information Technology) field have been used. As both these documents are widely used for the recruitment purpose, our final result could be beneficial for easing the recruitment process. Our approach could lead the way to automate the manual process of selecting the right candidate for a job.

To achieve our goal, we have come up with a set of methods that could help us in extracting the required information from the given document. Our defined methods will help us in extracting the major points such as personal information, educational background and previous working experience from a given CV. Similarly, from job vacancy, we will extract the job position, skills, responsibilities, education and work experiences demanded.

The methods that have been applied to achieve our goal has been properly discussed in the sections below.

2 Literature Review

The recent advances in Natural Language Processing have made it possible to understand the difficult patterns in textual corpus. This led to achieving the state-of-the-art result in different tasks of NLP such as Named Entity Recognition (NER), Text Classification, Part of Speech (POS) tagging and Information Extraction. As information extraction deals with extracting informative texts from a given document, it has been used for many commercial purposes too. One of which is CV parsing. Several commercial products related to HR automation like Sovren [1], Daxtra [2], Akken [3] and many others have been performing CV parsing.

There are several methods implemented for CV parsing, namely entity based information extraction methods, rule based information extraction methods, statistical information extraction methods, learning based methods. In previous implementations, CV parsing was performed by building a knowledge base system with a huge amount of data and retrieving keywords from a new CV by looking up on the knowledge base [4]. Big data tools combined with text analytical tools and Named Entity recognition have also been applied for CV parsing [5] which was able to achieve F-measure around 95% on CV parsing.

In a paper which is titled “Application of Machine Learning Algorithms to an online Recruitment System”, similar research has been done with the vision of making online recruitment effective and efficient. The author has tried multiple approaches like Linear Regression, M5 Model Tree, REPTree, SVM with polynomial kernel and SVM with PUK universal kernel. Among all of these approaches, the author states that the SVM with PUK universal kernel gave the best result. [6]

Canary, an application that leverages the NLP features to extract the medical related information from a document has applied user-defined grammars and lexicons for information extraction. The major steps which were used includes normalization of text and mapping acronyms and synonyms, defining vocabulary using user-specified words, creating a grammar rule that combines the words to form target phrases and setting a specific condition for extracting the information. The result extracted from this application can be used for biomedical research, clinical decision support for further processing. [9]

In the paper “FoodIE: A Rule-based Named-entity Recognition Method for Food Information Extraction” [10], the authors have proposed a rule based NER approach which is based on computational linguistics and semantic information for extracting food related information. The steps that they have done includes cleaning of text, using “coreNLP” and “USAS semantic tagger” for POS tagging which is further processed and then tag the tokens related to food using the “USAS semantic tagger” and a custom-defined rule. They stated that as food tokens are mostly either noun or adjective, they used this idea to improve their false positive rate by focusing on these POS tags. They have achieved a result of 97% precision, 94% recall and 96% F1 score while testing on 200 datasets. The paper “Using Stanford NER and Illinois NER to Detect Malay Named Entity Recognition” [11] has provided us the comparative result of the two NER tagging models. The models were tested in Malay language which is spoken in Brunei, Indonesia, Malaysia and Singapore. All together, 4 tags (PERSON, MISC, LOC and ORG) were used. In the end of the test, the author found that the Stanford NER gave the better tagging result with higher Precision and F1 score. Whereas, In the journal "Named Entity Recognition Approaches and Their Comparison for Custom NER Model" [12], the author Shelar, Kaur, Heda and Agrawal had performed a NER comparison between spaCy, Apache OpenNLP and TensorFlow. There the authors concluded that the NER model trained with spaCy provided better results.

3 Methods

3.1 Custom spaCy Pipeline

All the tasks that have been performed for parsing a document have been implemented using the spaCy pipeline component. spaCy is an open-sourced python library which is vastly used for Natural Language Processing (NLP) related tasks. We have created a custom spaCy pipeline component as per our need and added them in the spaCy's NLP pipeline.

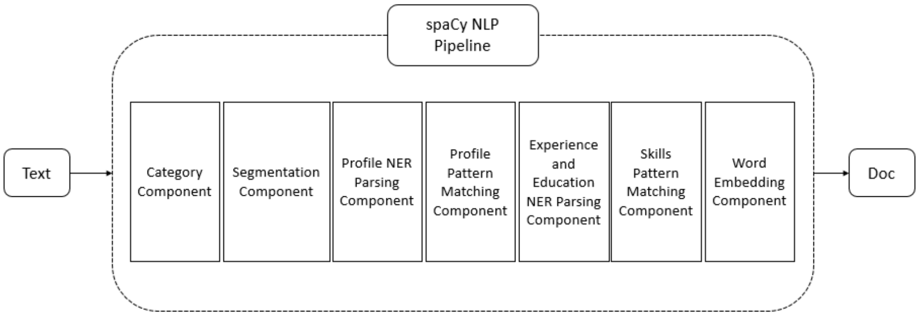

spaCy NLP Pipeline
Text
Category Component
Segmentation Component
Profile NER Parsing Component
Profile Pattern Matching Component
Experience and Education NER Parsing Component
Skills Pattern Matching Component
Word Embedding Component
Doc

Fig. 1. Custom spaCy pipeline components for parsing information

Figure 1 shows the custom pipeline components that we had built and added in the spacy's pipeline. The task of each of the component are: -

- Category Component
  This component helps in identifying the document's type.
- Segmentation Component
  This component is used for segmenting the CV into different sections
- Profile NER parsing Component
  This component is used to extract the name and address from a given text
- Profile Pattern Matching Component
  This component is used to extract the other personal information from a given text

- Experience and Education NER Parsing Component
  This component is used to extract information regarding experience and education
- Skills Pattern Matching Component
  This component is used to extract the different skills from a given text
- Word Embedding component
  This component is used to extract the embedding value of the words

Whenever we pass a text document on our spaCy object, it goes through these pipeline components and returns spaCy's Doc object. With the use of the Doc object, we can access it's attribute which contains our result.

Taking in consideration about the execution time, we have created separate methods for using the separate pipeline components. Which means that, when we use the "Segmentation Component", the other remaining pipeline components will be disabled. This has vastly helped us in improving the overall execution time.

3.2 Document Categorization

A document can be of different types and it is essential for the system to know whether the examined data is appropriate or not. As we are focusing on the CV and job vacancy related documents, we have created a classification model that would help us in identifying a given document. We divided the document into three classes, i.e. CV, job-vacancy-detail and others. The class "others" means any document that does not come under the other two classes.

A total of 10670 documents were used for training, in which 3754 were CVs, 3512 were job-vacancy-detail and 3404 were others like news articles and training certificates. 75% of the data has been used to train the model and remaining were used for testing. While training data were preprocessed by performing tokenization, removing stopwords and unwanted characters (punctuations, emails and bullet points), lemmatizing the words and converting them to lowercase. The data was linearly separable, thus SVM model identified most of the classes accurately with accuracy of 98.7%. We had also tried other ML algorithms like Naive Bayes and Random Forest, but SVM was the one that outperformed others.

After the identification of the document type (i.e either CV or job vacancy detail), the remaining major tasks are segmentation and information extraction.

CV Segmentation

The segmentation of CVs is the first major task. From a human perspective, we can split a CV into different parts on the basis of the textual information they contain. We can separate personal information, experience, objective, education part just from looking at them. And normally, these are the basic information contained in a CV.

To achieve segmentation of CVs similar to humans, we have created a function which segments CVs on the basis of different titles and provides their respective information. In order to identify, if the word belongs to the title or not, we created a Machine Learning model using the GaussianNB classifier algorithm. When a CV is read, it checks possible titles. If the model classifies a word as a title, then the textual information that follows would be allocated under it until a new title is found.

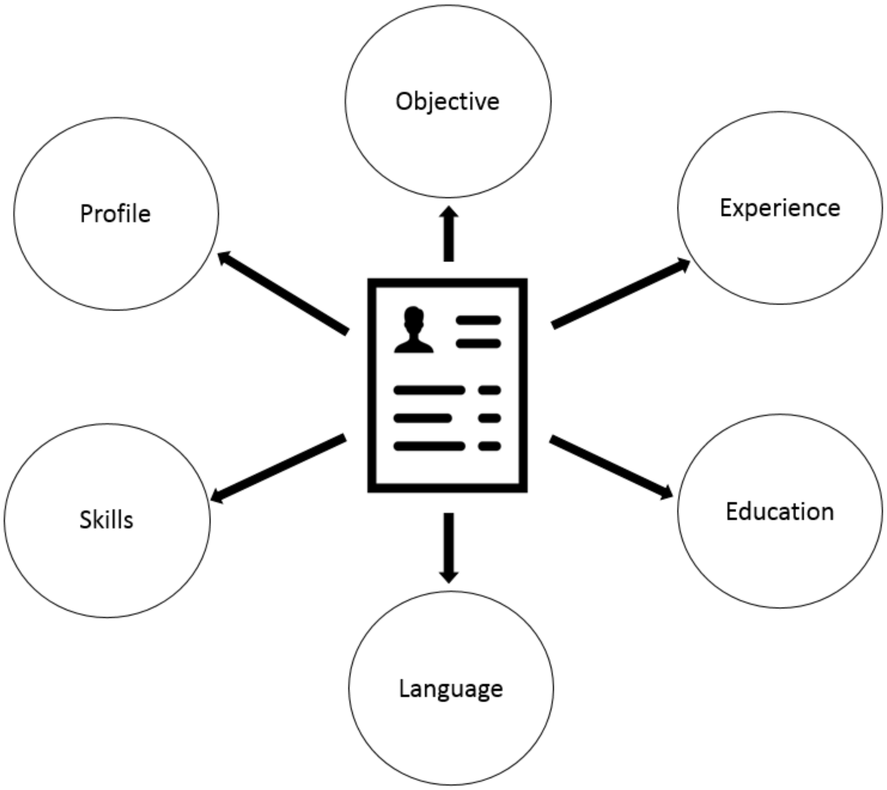


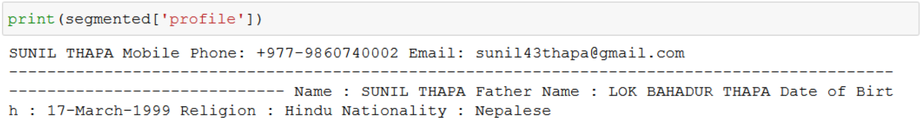
Fig. 2. The different titles in which a CV will be segmented

Segmenting CVs in different parts has vastly helped the system. With the segmented information, we are able to use a certain part of a CV for a respective task. If we want to extract personal information of the user, then for that we can only use the 'profile' segment.

Parsing of Information using NER

After successfully segmenting the CV into different parts, we had to extract the required information from them.

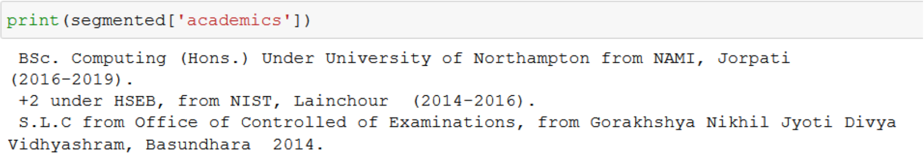
In retrieving the required information, one of the components that has been widely used is the Stanford NER model. We have created a custom Stanford NER model with the help of about 350 CVs. While training the NER model,



```
print(segmented['profile'])
```

```
SUNIL THAPA Mobile Phone: +977-9860740002 Email: sunil43thapa@gmail.com
---------------------------------------------------------------------------------------------
----------------------------- Name : SUNIL THAPA Father Name : LOK BAHADUR THAPA Date of Birt
h : 17-March-1999 Religion : Hindu Nationality : Nepalese
```

Fig. 3. Result of the profile section segmented from a CV



```
print(segmented['academics'])
```

```
 BSc. Computing (Hons.) Under University of Northampton from NAMI, Jorpati
(2016-2019).
 +2 under HSEB, from NIST, Lainchour  (2014-2016).
 S.L.C from Office of Controlled of Examinations, from Gorakhshya Nikhil Jyoti Divya
Vidhyashram, Basundhara  2014.
```

Fig. 4. Result of the education section segmented from a CV

the Conditional Random Field (CRF) algorithm has been used. Conditional Random Field (CRF) is a probabilistic graphical model which is widely used in Natural Language Processing (NLP) areas such as neural sequence labelling, Parts-of-Speech tagging (POS), Named Entity Recognition (NER), etc [7]. The Conditional Random Field (CRF) algorithm has been providing good results in many NER tagging tests. In the research paper "Named Entity Recognition using Conditional Random Fields", [15] the NER tagging performed in Marathi Language using CRF algorithm has provided a very good result.

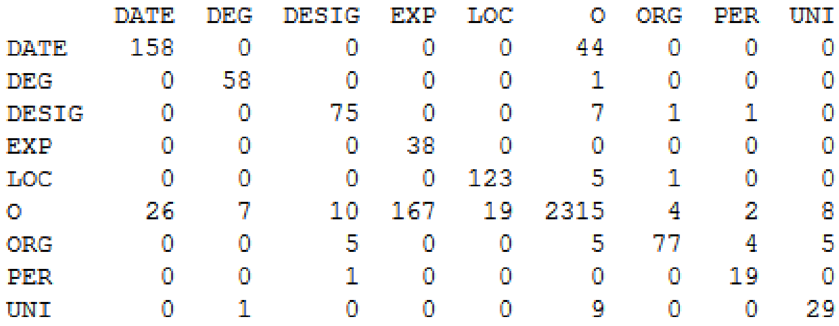
When tagging NER components, different tags that were used are as follows, PER (person), LOC (location), DATE, ORG (organization), DESIG (designation), EXP (experience), DEG (education), UNI (University), O (Unwanted tokens/text)

After the NER model was created, we tested it with 40 new CVs of similar format. The result from the testing was quite good. We have provided the confusion matrix and the classification report below.

Beside the NER model, we have also implemented CSV parsing for extracting information like skills, nationality and languages. We created a set of csv files, and listed down all the information regarding them. For skills, we separated it into two parts – technical skills and soft skills. So, if we want to retrieve the list of skills from a CV, we would read the csv file containing the skills and try matching the token between them. Same process was done for extracting the nationality and language from a CV.

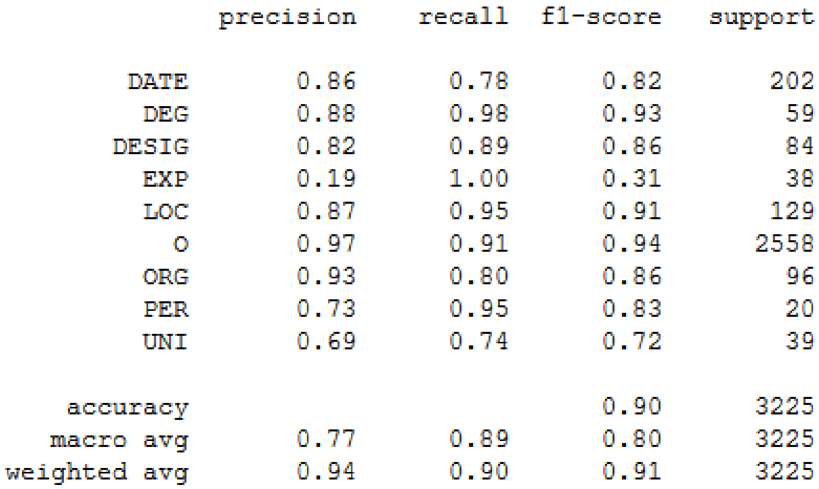
3.3 Job Vacancy Information Parsing

From the job vacancy type document, we worked on extracting the information like organization name, job location, job designation, educational requirement, work experience requirement and the skills (soft skills and technical skills) requirement.



```
       DATE  DEG  DESIG  EXP  LOC     O  ORG  PER  UNI
DATE    158    0      0    0    0    44    0    0    0
DEG       0   58      0    0    0     1    0    0    0
DESIG     0    0     75    0    0     7    1    1    0
EXP       0    0      0   38    0     0    0    0    0
LOC       0    0      0    0  123     5    1    0    0
O        26    7     10  167   19  2315    4    2    8
ORG       0    0      5    0    0     5   77    4    5
PER       0    0      1    0    0     0    0   19    0
UNI       0    1      0    0    0     9    0    0   29
```

Fig. 5. Confusion matrix of the tagged NER result



```
              precision    recall  f1-score   support

        DATE       0.86      0.78      0.82       202
         DEG       0.88      0.98      0.93        59
       DESIG       0.82      0.89      0.86        84
         EXP       0.19      1.00      0.31        38
         LOC       0.87      0.95      0.91       129
           O       0.97      0.91      0.94      2558
         ORG       0.93      0.80      0.86        96
         PER       0.73      0.95      0.83        20
         UNI       0.69      0.74      0.72        39

    accuracy                           0.90      3225
   macro avg       0.77      0.89      0.80      3225
weighted avg       0.94      0.90      0.91      3225
```

Fig. 6. Classification report of the NER model while testing

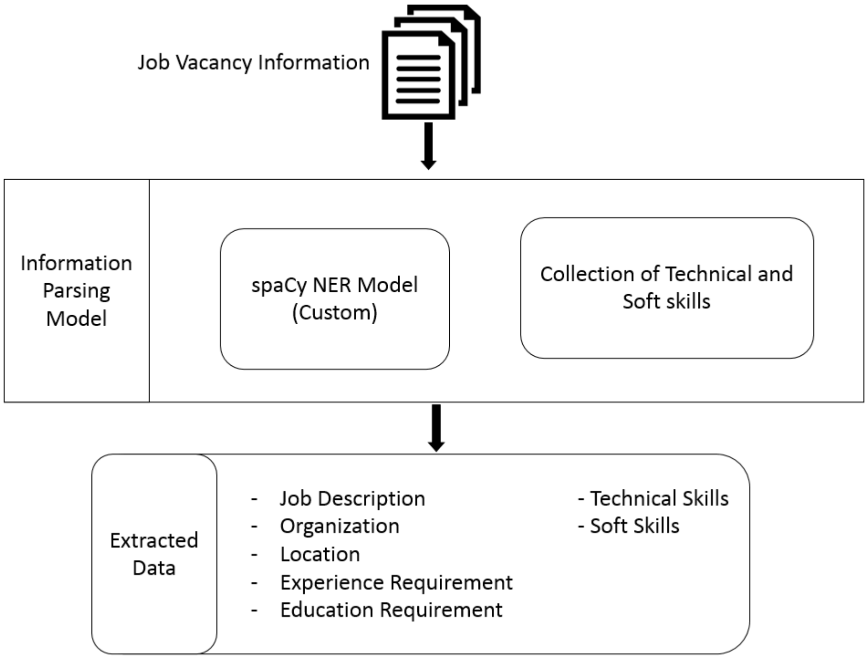


Fig. 7. Block diagram representing Job Vacancy Information Parsing module

Here, to extract the information we have used the custom trained spaCy NER (Named Entity Recognition) model. spaCy uses the Deep learning techniques for NER and can be summarised into 4 steps: Encode, Embed, Attend and Predict. Initially, it performs Feed Forward Neural Network to get the word embedding (Embed). Then, CNN (Convolutional Neural Network) is used to get the context dependent word embeddings for the tokens (Encode). The embeddings are then reduced into a single vector representation using the rule based self attention mechanism (Attend). At the end, single vectors are used for predicting the class using a Neural network (Predict). [8]

While training, 400 job vacancy documents were used having the parameters epoch = 40, optimizer = 'sgd' and dropout = 0.35. From the spaCy NER model we could extract the 'Organization Name', 'Job Location', 'Job Designation', 'Education' and 'Experience'. Whereas for the technical skills and soft skills, we extracted them using the pool parsing.

4 Conclusion and Future Work

Through this paper, we have demonstrated an efficient and accurate structured-information extraction from textual documents. Such accurate information extraction is possible by the use of NLP techniques with the application of Machine Learning (ML) and Deep Learning (DL) models. With the reference of textual documents like job vacancy information and CV, which are frequently used in applicant tracking systems, we have developed this system and tested it with documents of similar type. The evaluation metrics obtained are higher and the execution time for such information extraction is also better. Such highly optimized and accurate information extraction systems can be used in various other fields like research publications, job portals etc. However, the data used for training ML and DL models should be tweaked according to corresponding applications of the system.

Although the approaches that we have found now have solved our issue, we might have to change it in the near future, in case the requirement changes or new types of data are received. There are still some methods which are yet to be tested such as embedding using the BERT model, trying document similarity instead of token similarity and so on. Also, in the next stage, we will be focusing on extracting information from unstructured documents of other domains too.

References

1. Singh, S., 2018. Natural Language Processing for Information Extraction. arXiv, [online] Available at: ¡https://arxiv.org/pdf/1807.02383.pdf¿ [Accessed 20 January 2021].
2. Zeroual, I. and Lakhouaja, A., 2018. Data science in light of natural language processing: An overview. Procedia Computer Science, 127, pp.82-91.
3. Sovren.com. 2020. Home. [online] Available at: ¡https://www.sovren.com/¿ [Accessed 24 September 2020].
4. Technologies, D., 2020. Daxtra - CV Parsing. [online] Cn.daxtra.com. Available at: ¡http://cn.daxtra.com/¿ [Accessed 24 September 2020].
5. AkkenCloud. 2020. Top Staffing And Recruiting Software Solution — Akkencloud. [online] Available at: ¡https://www.akkencloud.com/¿ [Accessed 24 September 2020].
6. Chandola, D., Garg, A., Maurya, A. and Kushwaha, A., 2015. ONLINE RESUME PARSING SYSTEM USING TEXT ANALYTICS. [online] Jmdet.com. Available at: ¡http://www.jmdet.com/wp-content/uploads/2015/08/CR9.pdf¿ [Accessed 24 September 2020].
7. Das, P., Pandey, M. and Rautaray, S., 2018. A CV Parser Model Using Entity Extraction Process And Big Data Tools. [online] Mecs-press.net. Available at: ¡http://www.mecs-press.net/ijitcs/ijitcs-v10-n9/IJITCS-V10-N9-3.pdf¿ [Accessed 24 September 2020].
8. Faliagka, E., Ramantas, K. and Tsakalidis, A., 2012. Application of Machine Learning Algorithms to an online Recruitment System. The Seventh International Conference on Internet and Web Applications and Services
9. Malmasi, S., Sandor, N., Hosomura, N., Goldberg, M., Skentzos, S. and Turchin, A., 2017. Canary: An NLP Platform for Clinicians and Researchers. Applied Clinical Informatics, 08(02), pp.447-453.
10. Popovski, G., Kochev, S., Seljak, B. and Eftimov, T., 2019. FoodIE: A Rule-based Named-entity Recognition Method for Food Information Extraction. Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods,.

11. Sulaiman, S., Wahid, R., Sarkawi, S. and Omar, N., 2017. Using Stanford NER and Illinois NER to Detect Malay Named Entity Recognition. International Journal of Computer Theory and Engineering, 9(2), pp.147-150.
12. Shelar, H., Kaur, G., Heda, N. and Agrawal, P., 2020. Named Entity Recognition Approaches and Their Comparison for Custom NER Model. Science & Technology Libraries,
13. Pathak, K., 2020. A Tour Of Conditional Random Field. [online] Towards AI — The Best of Tech, Science, and Engineering. Available at: ¡https://towardsai.net/p/machine-learning/a-tour-of-conditional-random-field-7d8476ce0201¿ [Accessed 24 September 2020].
14. Patil, N., Patil, A. and Pawar, B., 2020. Named Entity Recognition using Conditional Random Fields. Procedia Computer Science, 167, pp.1181-1188.
15. Honnibal, M., 2016. Embed, Encode, Attend, Predict: The New Deep Learning Formula For State-Of-The-Art NLP Models · Explosion. [online] Explosion. Available at: ¡https://explosion.ai/blog/deep-learning-formula-nlp¿ [Accessed 25 September 2020].