International Journal of
Computational Linguistics Research

DLINE JOURNALS

Print ISSN: 0976-416X
Online ISSN: 0976-4178

IJCLR 2025: 16 (3)
https://doi.org/10.6025/ijclr/2025/16/3/99-108

A Proposal for Finding Combinations of Key Values From Texts

Rahul Patil
Computer Engineering
Pimpri Chinchwad College of Engineering, Pune. India
rahul3068@gmail.com

Prashant Ahire
Pune Pimpri Chinchwad College of Engineering
Pune Computer Science and Engineering Department patil, Pune. India
prashant.ahire@pccoepune.org

Amol Dhumane
Symbiosis Institute of Technology, Pune. India
amol.dhumane@sitpune.edu.in

Saomya Badoniya
Computer Engineering
Pimpri Chinchwad College of Engineering, Pune. India
saomyabadoniya@gmail.com

Resham Desai
Computer Engineering
Pimpri Chinchwad College of Engineering, Pune
reshamdesai109@gmail.com

Gautam Bhandari
Computer Engineering
Pimpri Chinchwad College of Engineering, Pune. India
gautambhandarino@gmail.com

Bikramjeet Singh Dhami
Computer Engineering
Pimpri Chinchwad College of Engineering, Pune. India
bsdhami2003@gmail.com

ABSTRACT

To extract key value pairs from documents like resumes, we have to use various processing techniques to get relevant information. The extracted information encompasses a diverse range of factors, including education, experience, skills, interests, and employment history. By gathering such information and presenting it in a structured format, hiring managers can thoroughly understand and evaluate the backgrounds of applying candidates, thereby simplifying the entire recruitment process. This versatility allows organisations to leverage various techniques and technologies to extract valuable information from resumes, enabling them to streamline their hiring processes and make informed decisions about their workforce.

Keywords: Key Value Pairs, Resumes, Recruiters, Candidates, Natural Language Processing, Query Parser, XML, JSON, MongoDB

Received: 8 May 2025, Revised 22 June 2025, Accepted 28 June 2025

1. Introduction

The process of extracting key-value pairs from documents, such as resumes, is a method for researching and extracting useful information about candidates in the form of key-value pairs. These informational pairs consist of a key and a corresponding value. This technique is typically used for data processing, information retrieval, and natural language processing. Various documents, such as emails, web pages, text files, and other materials, contain key-value pairs. Different types of structured documents include XML files, spreadsheets, databases, and others. By using key-value pairs, helpful information can be extracted from documents such as admission forms and contact details.

The first step in the extraction process is identifying the appropriate keys and their corresponding value components within the document. Furthermore, the key and the value are paired together to create a single entity. This pair is referred to as a key-value pair. The most effective techniques for this are regular expressions, natural language processing, mathematical models, and machine learning algorithms. Typically, extracting key-value pairs is a crucial step in extracting key values from any document. It helps in automating data processing and analysis. The extracted key-value pairs are used in data analysis, database fullness, and information retrieval.

It applies to both machine learning and natural language processing experience. Despite all this, it has several excellent benefits, including generating useful information from vast volumes of written material.

Example: Key Value K1 AAA, BBB, CCC K2 AAA,BBB K3 AAA, DDD K4 AAA, 2, 01, 01/01/2023 K5 3, ZZZ, 5623

2. Related Work

2.1 Resume Parser Using Natural Language Processing Techniques [4]

Information retrieval systems are used to ascertain relevant parts of a text, combine multiple such parts, and

Key	Value
K1	AAA,BBB,CCC
K2	AAA,BBB
K3	AAA,DDD
K4	AAA,2,01,01/01/ 2023
K5	3,ZZZ,5623

Table 1. Example of Key value pair

generate a structured representation of the information. Various scenarios exist in which HTML-formatted emails can be generated by filling a template with user- and transaction-specific values from databases. Gupta [1] described a generalised process to extract key-value pairs, which can be used for many applications. We analyse pairs for several applications, including identifying semantically similar keywords and clustering these keywords for building information extraction wrappers. Using Ontology, a method for extracting key concepts from text has been developed, called *CFinder*, and it outperforms existing methods. [2](Yong-Bin Kang and Pari {Delir Haghighi} and Frada Burstein]

A method for extracting thematically grouped key terms from text, utilising Wikipedia-based semantic relatedness and community detection, has been proposed by Grineva[3]. Index Finder is an algorithm that extracts key concepts from clinical texts by permuting words and filtering relevant UMLS concepts for indexing. [4] (Zou) Ahonen-Myka et al. (2007) outlined a two-phase process for finding co-occurring text phrases by combining frequent sequence and frequent set discovery techniques. These methods emphasize the significance of semantic relationships, domain knowledge, and co-occurrence patterns to effectively extract key values and concepts from their respective texts, ranging in application from information extraction to the creation of an ontology and text mining. [5] (Helena 2007)

Bhor et al [6] introduced a method utilizing machine learning for the extraction of key-value pairs from resumes. This approach consists of several stages, including data preprocessing, feature extraction, and model training and evaluation. Many studies utilise various machine learning algorithms, including Naive Bayes, Support Vector Machines, and Random Forests, to train the model, subsequently assessing its performance on a dataset comprising several hundred resumes. The aim is to automate the analysis of resumes using natural language processing techniques, thereby saving time and enhancing recruitment efficiency. The research conducted confirmed the practical application of natural language processing for automating resume parsing, including its comprehensive methodology, performance evaluation, comparison with alternative methods, and open-source implementation. This key value research illustrates machine learning techniques for extracting key-value pairs from resumes, presenting a more precise and effective method for this task. The studies employed a variety of machine learning algorithms to provide a thorough understanding of the effectiveness of the discussed methods and evaluate their performance accordingly. The clarity of the explanation regarding the methodology simplifies the replication of the studies, as it encompasses feature extraction, data preprocessing, model training, and

assessment. Although the papers do not explicitly identify any weaknesses, there may be potential limitations such as a narrow evaluation, insufficient real-world testing, restricted scope, limited generalizability, and an absence of discussion surrounding ethical considerations. Despite some limitations, including a small dataset and the lack of comparisons with leading methods, the studies offered a promising strategy for machine learning-driven resume information extraction.

2.2 Automatic Extraction of Segments from Resumes using Machine Learning

Some research, such as that by Gunaseelan [7], proposed a machine learning-based approach for automatically extracting important segments from resumes. The authors use NLP techniques and various machine learning algorithms to analyse resumes and extract relevant sections, and evaluate the system's performance using a resume dataset.

- **Steps in the machine learning-based resume segmentation method:**

a. Data collection: First, collect PDF or Word resumes.
The collection should include a variety of résumé formats, styles, and layouts.

b. Pre-processing removes photos, logos, and titles from the dataset's resumes. Removing stop words, punctuation, and special characters cleans the text.

c. Pre-processed resumes separate personal, education, work, skills, and project information. This is done with SVMs, DTs, and RFs.

d. Bag-of-words, TF-IDF, or word embeddings recover features of resumption segments. Machine learning algorithms use this to represent text numerically.

e. Logistic regression, neural networks, and Naive Bayes are trained using resume segment features. Features predict section type in models.

f. Evaluation: A test set of resumes assesses the trained models' ability to segment resumes. Precision, recall, and F1-score evaluating models.

g. Application: Trained models automatically segregate fresh resumes by assessing each text's part type. Segmented data helps rank keywords.

h. The machine learning-based resume section extraction method has potential for automating resume analysis. Machine learning and natural language processing accurately segment resumes, reducing manual screening time

The above research concludes that the mentioned machine learning-based approach is efficient in automatically screening resumes and can improve accuracy by reducing manual effort. The segmentation method is found to be more promising for automating the analysis of resumes. The partition resumes from text features, allowing us to use Decision trees, support vectors, random forests, etc. The process of feature extraction uses natural language processing. This process is scalable, efficient and accurate. The proposed methods can be limited by factors such as language support, context, and prejudice. This methodology can help streamline the hiring process by automatically passing the resume analysis.

2.3 Intelligent Recruitment System

Chamila Maddumage, Dulanjaya Senevirathne, and Isuru Gayashan[8] automated the hiring process. The sum of the process in the methodology includes data gathering, feature extraction, and other relevant steps. Using random forest and support vector machine, we can easily train the model and improve its effectiveness. The effectiveness is assessed using a dataset consisting of 50 job applications from authors. These studies help save time and effort in the requirements process through a smart requirements system that can automate the recruitment process. By offering a detailed examination of the advantages of each technique, the studies mention multiple machine learning algorithms and assess their efficiency. As the method is simple, the analysis can be repeated. It encompasses various processes, including data collection, feature extraction, data gathering, model training, algorithm development, application, and evaluation.

2.4 Key-Value Pair Searching System via Tesseract OCR and Post Processing

Kaló et al. provided details about the automatic system for extracting key-value pairs from scanned documents. Optical Character Recognition and post-processing techniques are used for the same. This methodology includes steps such as image pre-processing, post-processing, and Tesseract OCR. It utilises regular expressions and data cleaning techniques to extract key-value pairs. This system reduces the manual time and effort required for entering data.

- By using Tesseract by OCR, which is a widely used engine, we can evaluate the performance on a dataset of scanned documents. The analysis mentions various methods, including the use of optical character recognition (OCR). The drawbacks or limitations of this kind of study are that no detailed analysis is provided regarding the methods used in the paper. No comparisons regarding the performance of existing systems for key-value pair extraction are found. The pre-processing techniques that are used in the paper may not necessarily be helpful for all types of documents.

Hence, the research conducted explained an automatic system for extracting key value pairs from scanned documents using OCR techniques, along with pre-processing and post-processing. The lack of comparison represents a promising approach for automated data entry of scanned documents.

3. Research Gap Analysis

We identified several significant gaps in the existing literature.

1. Handling Unstructured Information

Various formats for resumes that contain unstructured data, such as tables, bullet points, and free-form text, are available.

Further research missions could explore multiple methods for efficiently collecting and managing scattered data, ensuring that all information is accurately recorded. Hence, handling unstructured information is essential.

2. Contextual Analysis and Semantic Understanding

Although existing methods are effective at identifying essential data points, such as education and work experience, they may do a better job of interpreting the semantic context of the gathered data.

To increase accuracy and relevance, research may focus on developing algorithms that analyse the context

surrounding key value combinations. Hence, contextual analysis and semantic understanding are essential.

3. Cross-linguistic and Cultural Variability

Along with the idea of writing in multiple languages, resumes can also contain cultural dissimilarities that can affect the importance attributed to the value pairings. More and more investigations into resume content can result in more accurate and inclusive methods. Hence, cross-linguistic and cultural variability is essential.

4. Our Work

Our proposed work can help parse through resumes in Microsoft Word to extract key-value pairs in JSON format. This makes it easier to retrieve specific information efficiently, such as education and hobbies. Our proposed system automatically initiates the extraction process, helping us save labour. This also simplifies the hiring process for any business by using text processing and data extraction algorithms. This improves the efficiency and performance of resume analysis. Through this contribution, we aim to expand the fields of automatic resume parsing and key-value extraction, which will save time and money by enabling users to make informed and effective labour decisions. Advances in machine learning and natural language processing techniques have led to a significant expansion of key value pair extraction from documents in recent years. Primarily, our technology makes the extraction process easier, reduces annual labour, and also allows us to sort authors by skill. Models like NLP and advanced machine learning have enabled this project to expand into key value formats. To present the data extracted from resumes, we'll use the key-value pair format. To complete this project, we'll gather a dataset of resumes in Word or PDF format. From these resumes, we'll extract key-value pairs representing the candidates' qualifications, employment history, educational background, and other relevant information. After converting the raw input file to XML or text format, the key-values will be represented in JSON format.

Our work may be helpful to future organisations in the recruitment process, increasing their efficiency and reducing the time taken and errors made every year.

4.1 Problem Statement and Objectives

To extract key-value pairs from resumes to find specific information within a document, where the key will be a label or identifier for the information, and the value will be the actual information itself.

- To parse through resumes submitted in Microsoft Word format and extract the key-value pairs in *Json* format.

4.2 Methodologies of Problem Solving

1. Identify the Issue: Make a diagnosis of the situation to maintain focus on the problem rather than just its symptoms. When analysing and identifying the underlying reasons of an issue, utilise cause-and-effect diagrams and flowcharts to illustrate the expected steps of a solution. The following sections outline the key steps in problem-solving. These initiatives promote the involvement of interested parties, the use of facts, and the contrast between expectations and reality, focusing on the root causes of an issue. Reviewing and documenting how the present processes are operating (i.e., who does what, with what information, using what technologies, talking with which organizations and persons, in what time frame, using what format) should be your first step. Evaluating how new resources and revised legislation could affect the development of your "what should be"

2. Provide Alternate Solutions: Wait to select a solution until several viable options have been presented. Examining multiple options may significantly boost the value of your perfect answer. This goal standard can

be used to create a roadmap for investigating alternatives once you've decided on the "what should be" model. Brainstorming and group problem-solving strategies are also helpful tools at this stage of the problem-solving process. There should be a wide range of possible solutions to consider before a final decision is made. When addressing problems, it's typical to choose the first workable solution—even if it's not the best fit—because options are considered as they are presented. If we focus only on one thing, we lose out on the chance to learn something new.

3. Assess and Choose an Option: Competent problem solvers use a variety of criteria to determine which option is the best. They consider the amount that a particular solution will address the problem without creating unexpected issues. All concerned parties will accept the alternative. Most likely, the alternative will be implemented. The substitute meets the organisation's needs.

4. Put the solution into practice and monitor its success: Expert problem solvers select the optimal solution based on a variety of criteria. They consider the amount that a particular solution will address the problem without creating unexpected issues. All concerned parties will accept the alternative. Most likely, the alternative will be implemented. The substitute meets the organisation's needs.

4.3 Our Modules

a. The "*myword.py*" and "*main.py*" modules serve as the foundation of our project, which is organised around a Microsoft Word resume document as input. Many essential methods, such as "*word.callwordprocess*()", "word.readwordfile()", "word.generatewordtextfile()", "word.docx_to_xml()",and "word.word_to_json_logic()", are called in "main.py".

b. To make document processing easier, we import necessary libraries and packages into "*myword.py*" like "docx", "lxml", "docx_utils.flatten", "opc_to_flat_opc", and "docx2txt". The input file path is initialized by the "callwordprocess()" method and saved in a global variable named "fname". "readwordfile()" uses "doc.paragraphs" to loop through the document's paragraphs, printing the text contained in the file.

c. Using the "docx" package, the "*generatewordtextfile*()" method takes text from the input resume and saves it in a file called "f1.txt". The "opc_to_flat_opc()" method in "docx_to_xml()" transforms the open XML document to flat XML format with the help of the "*docx_utils*" module. Converting resumes and extracting crucial data for further processing and analysis is made feasible with the help of these modules and features working together.

d. The data, which is produced post-processing the resume text using the "docx2txt" library, is saved in a variable which is called or named "text", and this variable is also a part of the "*word_to_json_logic*()" method present in the code.

e. The information inside the "text" variable is then converted into a list using a newline condition, which is often represented by "*n*". This causes segmentation. Hence, because of segmentation, each line of text in the list is treated as a unique element.

f. Lastly, the key variables will be found with the help of manual looking within the appropriate fields of the resumes. The extracted keys and their values are then prepared and displayed as key-value pairs. This provides structured data for further processing or analysis.

4.4 System Architecture

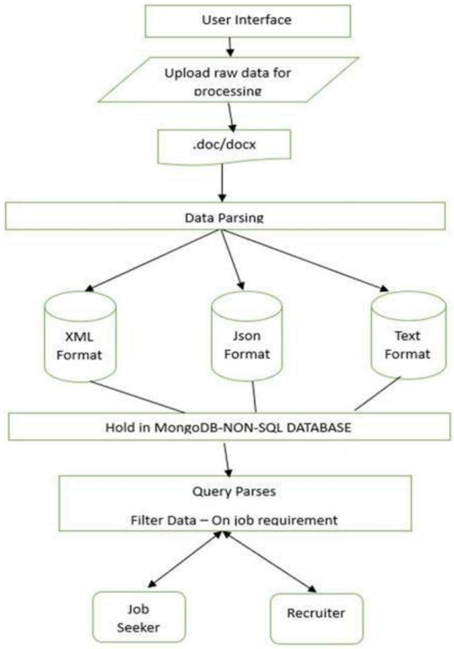

Figure 1. System Architecture

Figure 1 illustrates the system architecture for a Key-Value Pair Extraction framework. The architecture comprises several interconnected components that process input documents to identify and extract relevant key-value information. The system begins with a document input module, which accepts various formats, including scanned images and PDFs. These inputs undergo preprocessing steps, including image enhancement, binarization, and noise reduction, to ensure clarity and improve the accuracy of subsequent analysis.

The layout analysis module segments the document into logical blocks, distinguishing between text, tables, and graphical elements. This segmentation facilitates targeted processing in the next stage: Optical Character Recognition (OCR). The OCR module converts visual text elements into machine-readable text, forming the raw dataset for extraction.

Following OCR, the key-value detection module applies rule-based and machine learning algorithms to identify and pair relevant entities. The model effectively leverages linguistic cues, spatial relationships, and semantic similarity to establish key-value relationships.

Finally, an output module formats the extracted pairs for downstream applications or storage, supporting formats such as JSON or XML for easy integration. The architecture emphasises modularity, scalability, and adaptability to varied document types, ensuring robust performance in diverse real-world scenarios.

5. Limitations

This research limits its coverage and discussion in the proposal stage. The system implementation is not detailed, and we plan to bring it in the forthcoming analysis.

6. Future Directions

Although the initial focus of the analysis may be restricted, the concept of swift probing enables researchers to investigate how the analysis integrates into an actual workflow concerning the target text data, thus offering preliminary feedback for refining the modelling process. If the swift probing approach can indeed be incorporated into a hermeneutic framework in a way that satisfies open-minded researchers, a quicker examination of alternative analytical routes would become feasible. This could create significant additional momentum for transdisciplinary collaboration. It is still premature to identify genuinely text-focused instances of the suggested rapid probing technique. However, to make the conceptual idea more tangible, the work presents scenarios to demonstrate how rapid probing could assist in addressing the scheduling dilemma and the issue of subjectivity, respectively [10]. The first scenario showcases the transfer of intricate analysis pipelines across different corpora, while the second focuses on swift annotation experiments aimed at character mentions within literary texts.

7. Conclusion

This paper concludes that extracting key value pairs from a resume is an essential and time-saving process for an organisation. It helps them to save time and manpower if used to check resumes one by one. It can take time, and accuracy may be lower in such cases. Extracting key value pairs from a resume is much more efficient and can identify candidates according to the organisation. By utilising Python and various SDLC models, we can enhance accuracy, which will benefit the organisation to some extent. The recruitment process can be faster. The opportunities are equal among all the employees, and the best one gets selected. It can also help the organisation collect and classify data for future recruitment purposes, allowing many firms to rely on technology. It is helpful for them in the end. Hence, extracting key values from a resume can help develop organisations and make the recruitment process easier.

References

[1] Gupta, R. (2020). Generic Key Value Extractions from Emails. In: Bellatreche, L., Goyal, V., Fujita, H., Mondal, A., Reddy, P.K. (eds) Big Da*ta Analytics. BDA 2020*. Lecture Notes in Computer Science, P. 12581. Springer,)

[2] Kang, Yong-Bin., Haghighi, Pari Delir., Burstein, Frada. (2014). C Finder: An intelligent key concept finder from text for ontology development, Expert Systems with Applications. V. 41 (9) p. 4494-4504.

[3] Grineva, P., Maria., Grinev, N., Maxim., Lizorkin, D. (2009). Effective Extraction of Thematically Grouped Key Terms From Text, *Published in AAAI Spring Symposium.*

[4] Zou, Q., Chu, W., Morioka, C., Leazer, G., Kangarloo, H. (2003). IndexFinder: A Method of Extracting Key Concepts from Clinical Texts for Indexing. American Medical Informatics Association Annual Symposium.

[5] Ahonen-Myka, Helena., Heinonen, Oskari., Klemettinen, Mika. (2007). Finding Co-occurring Text Phrases by Combining Sequence and Frequent Set Discovery. *In:* Proceedings of the ESF Exploratory Workshop on Pattern Detection and Discovery P. 180 - 189.

[6] Bhor, S., Gupta, V., Nair, V., Shinde, H., Kulkarni, M.S. (2021). Resume parser using natural language processing techniques. *Int*. J. Res. Eng. Sci, 9.

[7] Gunaseelan, B., Mandal, S., Rajagopalan, V. (2020). Automatic Extraction of Segments from Resumes using Machine Learning, *2020 IEEE 17th* India Council International Conference (INDICON), New Delhi, India, p. *1-6,* doi: 10.1109/INDICON49873.2020.9342596.

[8] Maddumage, C., Senevirathne, D., Gayashan, I., Shehan, T., Sumathipala, S. Intelligent Recruitment Sys tem, *2019 IEEE 5th* International Conference for Convergence in Technology (I2CT), Bombay, India, 2019, p. *1-6,* doi:10.1109/I2CT45611.2019.9033836.

[9] Kaló, Á. Z., Sipos, M. L. (2021). Key-Value Pair Searching System via Tesseract OCR and Post Processing, *2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI), Herl'any, Slovakia, 2, p. 000461-000464,* doi:10.1109/SAMI50585.2021.9378680.

[10] Kuhn, J. (2019). Computational text analysis within the Humanities: How to combine working practices from the contributing fields?. *Lang Resources & Evaluation V. 53, P. 565–602*.