Article

Consolidating Access to Candidate Data for Recruitment Headhunting: Leveraging Explainable Machine Learning

Mncedisi Mncwabe1, Dr. Thulane Paepae2

|  |
| --- |
| Received: date  Revised: date  Accepted: date  Published: date  **Citation:** To be added by editorial staff during production.  **Copyright:** © 2025 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/). |

1 Department of Electrical and Electronic Engineering Science; University of Johannesburg; 201591994@student.uj.ac.za

2 Mathematics and Applied Mathematics; University of Johannesburg; tpaepae@uj.ac.za

**\*** Correspondence: 201591994@student.uj.ac.za

**Abstract:** In the recruitment space, headhunting is one of the processes used by recruiters to hire candidates for open job positions. To identify the best candidates for the job during a headhunting process, recruiters individually visit multiple job portals such as LinkedIn, Pnet, Indeed and provide certain search criteria, such as job title, location, education, and other parameters. They then sift through the search results and manually save the candidates matching the provided search parameters into datasets from each job portal to further contact the candidates. This process is time-consuming and delays the process of finding suitable candidates quicker for the role. This study presents a unified search to consolidate candidate information from several job portals and a machine learning recommendation framework to enhance this recruitment process. This study leverages existing professional data aggregation APIs to collect, standardize and unify candidate data from multiple job portals and transform it into a structured format suitable for machine learning modelling. Our approach employs TF-IDF (Term Frequency-Inverse Document Frequency) embeddings, cosine similarity and regression models such as Ridge, Gradient Boosting and Random Forest to effectively match and rank candidates against job descriptions, incorporating Shapash for explainability of candidate rankings. The results from the consolidation of candidate data show an improvement in data quality and comprehensiveness of candidate information, enhancing the likelihood of finding the best match for the job quicker. Ridge regression achieved the highest performance compared to Random Forest and Gradient Boosting in candidate matching accuracy, with an R² score of 0.955 and RMSE of 0.021 on training data, and an R² score of 0.924 and RMSE of 0.024 on test data. Furthermore, the findings from Shapash analysis provide insights and features to understand the contribution of features to each candidate’s ranking, thereby providing transparency to recruiters. The inclusion of explainability of candidate ranking against job descriptions in this study advances the current literature on AI-based recruitment systems. The originality of this study lies in its combination of professional data aggregation APIs to unify candidate data from various job portals and the use of explainable machine learning using the Shapash framework. This methodology not only offers transparency in candidate ranking but it is also based on real-world and up-to-date candidate data.

**Keywords:** Candidate Ranking; Recruitment Headhunting; Explainable Machine Learning; Data Aggregation APIs; TF-IDF; Ridge Regression; Random Forest

1. Introduction

The AI-based systems in the recruitment industry are rapidly evolving. However, during the headhunting process to fill an open job position, manual searches for potential candidates across various job portals remain a relevant practice. Typically, recruiters navigate multiple job portals such as LinkedIn, Pnet and Indeed entering job titles and other search parameters in each job portal to source potential candidates matching the specified search parameters. Additionally, they manually compile the results from each job portal and subsequently contact the candidates. This approach has shortcomings such as time consumption and delay in identifying ideal candidates for the role quicker. While some studies, such as the study by (Bogers & Mesut, 2021), explore how recruiters manually find candidates data across various job portals for headhunting, specifically focusing on their information-seeking behavior and engagements with the job portals, it does not necessarily suggest how recruiters can unify access to candidate data from multiple job portals. Another study by (Peicheva, 2022) explores how recruiters use the Applicant Tracking Systems (ATS) to manage incoming job applications, however the functionality of these systems is not meant to consolidate candidate data profiles from various job portals when recruiters are headhunting candidates.

There has been a significant attention in explainability and interpretability of machine learning models in recent years in AI-based systems across various industries. Even though machine learning models can be highly accurate, their complexity can pose a challenge, especially for non-technical users such as recruiters, to understand how certain decisions were made by the model (Xianlong, 2024). The importance of building machine learning algorithms that are not only accurate but also explainable is underscored by (Beretta, et al., 2024) who emphasize the need for transparency to build trust and ensure fairness in AI-based recruitment. Some studies use machine learning to enhance the recruitment process by matching and ranking candidate profiles to job descriptions based on similarity scores. However, the studies in existing literature either only report on accuracy based on cosine similarity without explanations into the factors influencing the candidate rankings or use the SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) evaluation metrics for explainability. For instance, a study by (Aram Khasro, et al., 2025) demonstrated high accuracy rates using SVM and LSTM models achieving 94% and 92% accuracy in matching and ranking candidates against job descriptions, respectively, but did not incorporate explainability in order to understand why certain candidates were a better match than others. Another study by (Magham, 2024) explores the role of explainable machine learning in AI-driven recruitment using the SHAP and LIME evaluation metrics. While SHAP and LIME are established tools for machine learning algorithm explainability, their technical nature can be complex and not easily understandable by non-technical users (Xianlong, 2024).

Considering this gap in current research, this study proposes a unified approach to consolidate candidate data from several job portals and a machine learning recommendation framework to enhance this recruitment process while providing explainable candidate rankings. The aim of this study is to develop a methodology that effectively tackles the difficulties of manually accessing candidate data from various job portals for headhunting in the recruitment industry and integrate machine learning recommendation engines to rank and match candidates to job descriptions with explainability frameworks understandable by non-technical audiences. Firstly, this study aims to achieve this by utilizing existing professional data aggregation Application Programming Interfaces (APIs) that provide publicly accessible data about employees such as their experience history, skills, education and certifications from various job portals worldwide. Secondly, although candidate matching and ranking has been enhanced using machine learning techniques by existing studies, there remains a gap in the explainability of these AI-driven recommendations. Existing studies prioritize accuracy without providing clear insights into why certain candidates are ranked and recommended as the most ideal candidates for the role over others. We employ machine learning regression methods such as Ridge, Gradient Boosting, Random Forest and TF-IDF embeddings, to efficiently match candidates to job descriptions. Additionally, we utilize Shapash, a Python library designed to make machine learning model outcomes more understandable, especially to non-technical audiences (Read the Docs, 2025), to extract feature importance and explain candidate rankings, thereby enhancing the transparency of our recommendations. By addressing the inefficiencies of manually visiting multiple job portals to obtain candidate information during the headhunting process and incorporating explainable machine learning in matching candidates to job descriptions, this research contributes to the broader field of AI-powered recruitment.

2. Literature Review

The literature review in this study provides insights into existing research and theoretical frameworks relevant to this study, highlighting key contributions and identifying gaps that this research seeks to address. For the literature review, the following studies were cited.

2.1 Review

The study by (Bogers & Mesut, 2021) explores how recruiters find candidates data across various job portals for headhunting, specifically focusing on their information-seeking behavior and engagements with the job portals. Their study's results demonstrate that recruiters utilize keyword searches such as job titles in each job portal's search engine. Their study did not specifically address how to unify access to candidate data from various job portals.

The study by (Aram Khasro, et al., 2025) uses a BERT-based contextual embeddings methodology to enhance job applicant matching. Their study used synthetic data sourced from Kaggle and demonstrated high accuracy rates using SVM, LSTM and MLP models achieving 94%, 92% and 90% accuracy, respectively. However, this study did not incorporate explainability to understand why some job applicants would be a better match to a job than others.

Furthermore, (Magham, 2024) explores the application of explainable machine learning in mitigating bias within AI-driven recruitment using the SHAP and LIME evaluation metrics. They highlight how organizations can enhance transparency in their recruitment processes by implementing explainable machine learning strategies. While SHAP and LIME are established tools for machine learning algorithm explainability, their technical nature can be complex and not easily understandable by non-technical users (Xianlong, 2024).

Another study by (Kumar et al, 2023) combines web scraping and crawling methods to collect job data and utilizes a hybrid recommendation system that combines content-based and collaborative filtering to recommend jobs to candidates based on their profiles. The study achieved an average of 59.78% increase in candidate match scores, showcasing the success of integrating various recommendation methods. However, their system does not aggregate candidate data, rather, it recommends jobs to candidates. Moreover, the research employs web scraping, according to (Mydyti & Ware, 2025), scraped data can be inconsistent and incomplete due to frequent website updates, leading to errors and outdated information. The research does not provide model explainability to clarify why some candidates are ranked higher or lower than others.

A use of Stochastic Gradient Descent (SGD) classifier by (Kitichalermkiat et al.,2022) explores the classification of job positions into standardized occupational Thailand categories. In their study they build an algorithm that suggests industries sectors that might be suitable for a candidate based on their previous employment, not necessarily matching a candidate to a job description. They use data from a single government source in Thailand, which may limit generalizability. The study achieved a reported accuracy of 99.89%, demonstrating the effectiveness of using SGD for job classification.

In their study, (Delecraz et al., 2022) discuss the transparency and explainability of machine learning models in the context of recruitment. An XGBoost model is employed to match candidates with job descriptions, while SHAP values are utilized for providing model explainability. The research utilizes job application data from an internal platform called Gojob, but it does not necessarily consolidate candidate data from various job portals. The study indicates that their machine learning model achieved a True Positive Rate Parity of under 5% across sensitive characteristics and effectively employed 43% of NEET individuals in temporary positions.

The effectiveness of using machine learning in making hiring decisions is underscored by (Smelyakov et a., 2023). They focus on predicting labeled data with a recruitment status indicating whether a candidate was hired (Y) or not hired (N). The study focuses on the interpretability of decision tree algorithms with a reported accuracy of 73-78%. While the data used in this study includes comprehensive candidate features, it does not necessarily highlight how candidate data can be consolidated from multiple job portals.

A methodology for automating resume shortlisting by leveraging text processing and TF-IDF techniques to improve efficiency is proposed by (Borgave et al., 2023). Their study explores the use of tokenization, stop-word removal, stemming and calculating similarity scores between job descriptions and resumes using TF-IDF and cosine similarity. The KNN (k-Nearest Neighbors) classifier is then used to classify resumes based on similarity scores. The study reports an accuracy of 80.22%, with precision, recall and F1-score of 0.85, 0.75 and 0.80, respectively. The paper lacks model interpretability to clarify why some candidates are ranked higher or lower than others and does not highlight how candidate data can be consolidated from multiple job portals.

**Table 1.** Summary of Literature Review.

|  |  |  |  |
| --- | --- | --- | --- |
| **Authors** | **Summary** | **Results** | **Limitations** |
| (Bogers & Mesut, 2021) | Focuses on information-seeking behavior and engagements of recruiters with the job portals. | Demonstrates that recruiters utilize keyword searches such as job titles in each job portal's search engine. | Does not address how to consolidate access to candidate data from various job portals. |
| (Aram Khasro, et al., 2025) | Used a BERT-based contextual embeddings methodology to enhance job applicant matching. | Achieved high accuracy rates of 94% for SVM, 92% for LSTM, and 90% for MLP. | Lacks model explainability. |
| (Magham, 2024) | Explores the application of explainable machine learning in mitigating bias within AI-driven recruitment. | Enhanced transparency in AI-driven recruitment processes. | Uses SHAP and LIME values, which can be complex to non-technical users. |
| (Kumar et al., 2023) | Combines web scraping and crawling methods to collect job data and utilizes a hybrid recommendation system to recommend jobs to candidates based on their profiles. | Achieved an average of 59.78% increase in candidate match scores. | Scraped data can be inconsistent and outdated. Does not consolidate candidate data from multiple job portals and lacks model explainability. |
| (Kitichalermkiat et al.,2022) | Explores the classification of job positions into standardized occupational Thailand categories. | Achieved a reported accuracy of 99.89%. | Focused on data from a single government source in Thailand, which may limit generalizability. |
| (Delecraz et al., 2022) | Builds an ML model to match candidates to job offers and provides an explainable model to internal recruiters using XGBoost for job matching. | True Positive Rate Parity of under 5% across sensitive characteristics and effectively employed 43% of NEET individuals in temporary positions. | Does not explicitly consolidate candidate data from various job portals. |
| (Smelyakov et al., 2023) | Focuses on predicting labeled data with a recruitment status indicating whether a candidate was hired (Y) or not hired (N). | Reported accuracy of 73-78%. | Does not explicitly consolidate candidate data from various job portals. |
| (Borgave et al., 2023) | Proposes a methodology for automating the shortlisting of resumes by utilizing TF-IDF and text processing techniques to increase efficiency. | Achieved an accuracy of 80.22%, precision of 0.85, recall of 0.75 and an F1-score of 0.80. | Lacks model and does not highlight how candidate data can be consolidated from various job portals. |

2.2. Data Aggregation and APIs

The research conducted by (Mydyti & Ware, 2025) emphasizes ethical issues like breaches of data privacy regulations and discrepancies that may arise with web scraping, particularly when obtaining sensitive data. They further note that scraped data can be inconsistent and incomplete due to frequent website updates, leading to errors and outdated information. In our study we leverage existing professional data aggregation APIs. These APIs provide access to publicly available employee data sourced and aggregated from various job portals, including experience history, skills, education and certifications, ensuring that the candidate data is real-world, up-to-date and relevant to the current job market. This methodology addresses the need for consolidated candidate data source that enhances the efficiency and scalability of candidate searches.

2.3. Natural Language Processing

Natural Language Processing (NLP) refers to a computer program's capability to comprehend human language in both its spoken and written forms. It is a branch of Artificial Intelligence (TechTarget, 2024). In our paper we use NLP techniques such as TF-IDF to calculate embeddings (converting text data to numerical vectors) to calculate the similarity scores between candidate profiles and job descriptions. Other NLP techniques such as tokenization, stop-word removal and stemming are used in this study to clean and extract relevant features from candidate profiles textual data, making it easier to match and rank candidates to job descriptions based on their qualifications, experience and skills.

2.4. Machine Learning

Machine Learning is a subset of Artificial Intelligence dedicated to enabling computer programs to replicate human learning process and complete tasks autonomously (IBM, 2021). In this study we employ machine learning models such as Ridge Regression, Random Forest and Gradient Boosting to predict the similarity score between a candidate profile and job descriptions. These models are trained on TF-IDF embeddings and provide insights into the features that influence candidate rankings. The use of these models enhances the precision of candidate-job matching and facilitates the identification of the most relevant candidates for specific roles.

2.5. Explainable AI (XAI)

Explainable AI is essential for ensuring transparency, trust and interpretability of machine learning models. It addresses the challenges of understanding the decisions made by machine learning models (Amit Ganatra et al., 2024). In this study we employ XAI techniques such as Shapash to provide summary reports and visualizations of model feature importance, offering clear and human-readable explanations for why certain candidates rank higher or lower for roles. This approach offers clear reports that non-technical users such as recruiters can easily understand.

3. Materials and Methods

To achieve the objectives of this research, the following methodology was adhered to. The proposed methodology is intended to be adaptable across any industry and job roles in the recruitment domain.

3.1 Selection of the Data Source and API Evaluation

We evaluated two official employee data aggregation APIs, Coresignal and Proxycurl (now called Enrich Layer). These are APIs that provide access to real-world, consolidated and publicly available employee data across the globe from multiple job portals such as LinkedIn, Glassdoor, GitHub and Indeed. Additionally, the use of these APIs for data collection in this study is driven by their adherence to data privacy laws like the GDPR and legal frameworks for data collection, which is advantageous over manual scraping or unofficial methods of gathering people data. While such APIs require paid subscriptions, they offer free trials with sufficient data access for research purposes, making them viable for this study.

After a comparative analysis of employee data coverage between the two APIs, Coresignal was selected as the data source for this study because of its wider range coverage of multiple job portals where employee data is sourced and aggregated, which better reflects the multiple job portal searches employed by recruiters in practice. Coresignal boasts 20+ employee data sources compared to Proxycurl’s 4 (Proxycurl, 2022).

3.2 User Interface Development

A web-based user interface (UI) was created using Streamlit to facilitate consolidated searches of candidate data. Streamlit is an open-source python framework that enables the development and sharing of custom web applications (Streamlit, 2025). The interface enables users to enter search criteria such as job titles, skills, educational institutions and certifications. The UI integrates with the Coresignal API to retrieve candidates matching the specified search criteria from various job portals.

3.3 Data Collection and Processing

Candidate data was retrieved through the Coresignal API using RESTful HTTP requests. The API returns unstructured data in JSON format that includes candidate profiles with details such as educational background, skills, educational institution names, certifications and professional experience. The raw JSON data was processed to convert the unstructured data into tabular format suitable for machine learning model training. The cleaning of the data included JSON parsing and flattening of nested structures, handling of missing values and data inconsistencies, standardization of field formats and naming conventions. The data scientist roles were specifically targeted for candidate data retrieval to establish a dataset for model training. Even though this study focuses on data scientist roles, the same methodology can be applied to other job titles and descriptions.

3.3 Natural Language Processing

The Natural Language Toolkit (NLTK) and scikit-learn libraries were used to apply Natural Language Processing (NLP) techniques to job descriptions and candidate profiles (skills, education, professional experience and certifications). The preprocessing comprised of text normalization and lowercasing, removal of stop words, removal of special characters and text tokenization.

3.3 Feature Extraction

To create numerical embeddings for candidate profiles and job descriptions, scikit-learn's TfidfVectorizer was used to implement Term Frequency-Inverse Document Frequency (TF-IDF) vectorization.

3.3 Similarity Calculation

Cosine similarity is the measure of distance between two vectors. It is measured by calculating the cosine angle between the two vectors (DataCamp, 2025). This measure is used in this study to calculate the similarity score between candidate profiles and the job description. It is calculated by measuring the cosine angle between the TF-IDF embeddings of candidate profiles and the embeddings of a job description using the cosine similarity function from scikit-learn library. This generated a continuous similarity score ranging from 0 to 1, representing the degree of match between candidate profiles and job requirements. A value of 0 indicates no similarity and 1 indicates perfect similarity. The similarity score served as the primary continuous target variable for machine learning modeling.

4. Results

This section may be divided by subheadings. It should provide a concise and precise description of the experimental results, their interpretation, as well as the experimental conclusions that can be drawn.

4.1. Subsection

4.1.1. Subsubsection

Bulleted lists look like this:

* First bullet;
* Second bullet;
* Third bullet.

Numbered lists can be added as follows:

1. First item;
2. Second item;
3. Third item.

The text continues here.

4.2. Figures, Tables and Schemes

All figures and tables should be cited in the main text as Figure 1, Table 1, etc.

**A black background with a black square

Description automatically generated with medium confidence**

**Figure 1.** This is a figure. Schemes follow the same formatting.

**Table 1.** This is a table. Tables should be placed in the main text near to the first time they are cited.

|  |  |  |
| --- | --- | --- |
| **Title 1** | **Title 2** | **Title 3** |
| entry 1 | data | data |
| entry 2 | data | data 1 |

1 Tables may have a footer.

The text continues here (Figure 2 and Table 2).

|  |  |
| --- | --- |
| C:\Users\martin\Downloads\testFigure.tif | C:\Users\martin\Downloads\testFigure.tif |
| (**a**) | (**b**) |

**Figure 2.** This is a figure. Schemes follow another format. If there are multiple panels, they should be listed as: (**a**) Description of what is contained in the first panel; (**b**) Description of what is contained in the second panel. Figures should be placed in the main text near to the first time they are cited.

**Table 2.** This is a table. Tables should be placed in the main text near to the first time they are cited.

|  |  |  |  |
| --- | --- | --- | --- |
| **Title 1** | **Title 2** | **Title 3** | **Title 4** |
| entry 1 \* | data | data | data |
| data | data | data |
| data | data | data |
| entry 2 | data | data | data |
| data | data | data |
| entry 3 | data | data | data |
| data | data | data |
| data | data | data |
| data | data | data |
| entry 4 | data | data | data |
| data | data | data |

\* Tables may have a footer.

4.3. Formatting of Mathematical Components

This is example 1 of an equation:

|  |  |
| --- | --- |
| a = 1, | (1) |

the text following an equation need not be a new paragraph. Please punctuate equations as regular text.

This is example 2 of an equation:

|  |  |
| --- | --- |
| a = b + c + d + e + f + g + h + i + j + k + l + m + n + o + p + q + r + s + t + u + v + w + x + y + z | (2) |

the text following an equation need not be a new paragraph. Please punctuate equations as regular text.

Theorem-type environments (including propositions, lemmas, corollaries etc.) can be formatted as follows:

**Theorem 1.** Example text of a theorem. Theorems, propositions, lemmas, etc. should be numbered sequentially (i.e., Proposition 2 follows Theorem 1). Examples or Remarks use the same formatting, but should be numbered separately, so a document may contain Theorem 1, Remark 1 and Example 1.

The text continues here. Proofs must be formatted as follows:

**Proof of Theorem 1.** Text of the proof. Note that the phrase “of Theorem 1” is optional if it is clear which theorem is being referred to. Always finish a proof with the following symbol. □

The text continues here.

5. Discussion

Authors should discuss the results and how they can be interpreted from the perspective of previous studies and of the working hypotheses. The findings and their implications should be discussed in the broadest context possible. Future research directions may also be highlighted.

6. Conclusions

This section is not mandatory but can be added to the manuscript if the discussion is unusually long or complex.

7. Patents

This section is not mandatory but may be added if there are patents resulting from the work reported in this manuscript.

**Supplementary Materials:** The following supporting information can be downloaded at: https://www.mdpi.com/article/doi/s1, Figure S1: title; Table S1: title; Video S1: title.

**Author Contributions:** For research articles with several authors, a short paragraph specifying their individual contributions must be provided. The following statements should be used “Conceptualization, X.X. and Y.Y.; methodology, X.X.; software, X.X.; validation, X.X., Y.Y. and Z.Z.; formal analysis, X.X.; investigation, X.X.; resources, X.X.; data curation, X.X.; writing—original draft preparation, X.X.; writing—review and editing, X.X.; visualization, X.X.; supervision, X.X.; project administration, X.X.; funding acquisition, Y.Y. All authors have read and agreed to the published version of the manuscript.” Please turn to the [CRediT taxonomy](https://img.mdpi.org/data/contributor-role-instruction.pdf) for the term explanation. Authorship must be limited to those who have contributed substantially to the work reported.

**Funding:** Please add: “This research received no external funding” or “This research was funded by NAME OF FUNDER, grant number XXX” and “The APC was funded by XXX”. Check carefully that the details given are accurate and use the standard spelling of funding agency names at https://search.crossref.org/funding. Any errors may affect your future funding.

**Institutional Review Board Statement:** In this section, you should add the Institutional Review Board Statement and approval number, if relevant to your study. You might choose to exclude this statement if the study did not require ethical approval. Please note that the Editorial Office might ask you for further information. Please add “The study was conducted in accordance with the Declaration of Helsinki, and approved by the Institutional Review Board (or Ethics Committee) of NAME OF INSTITUTE (protocol code XXX and date of approval).” for studies involving humans. OR “The animal study protocol was approved by the Institutional Review Board (or Ethics Committee) of NAME OF INSTITUTE (protocol code XXX and date of approval).” for studies involving animals. OR “Ethical review and approval were waived for this study due to REASON (please provide a detailed justification).” OR “Not applicable” for studies not involving humans or animals.

**Informed Consent Statement:** Any research article describing a study involving humans should contain this statement. Please add “Informed consent was obtained from all subjects involved in the study.” OR “Patient consent was waived due to REASON (please provide a detailed justification).” OR “Not applicable.” for studies not involving humans. You might also choose to exclude this statement if the study did not involve humans.

Written informed consent for publication must be obtained from participating patients who can be identified (including by the patients themselves). Please state “Written informed consent has been obtained from the patient(s) to publish this paper” if applicable.

**Data Availability Statement:** We encourage all authors of articles published in MDPI journals to share their research data. In this section, please provide details regarding where data supporting reported results can be found, including links to publicly archived datasets analyzed or generated during the study. Where no new data were created, or where data is unavailable due to privacy or ethical restrictions, a statement is still required. Suggested Data Availability Statements are available in section “MDPI Research Data Policies” at https://www.mdpi.com/ethics.

**Acknowledgments:** In this section, you can acknowledge any support given which is not covered by the author contribution or funding sections. This may include administrative and technical support, or donations in kind (e.g., materials used for experiments). Where GenAI has been used for purposes such as generating text, data, or graphics, or for study design, data collection, analysis, or interpretation of data, please add “During the preparation of this manuscript/study, the author(s) used [tool name, version information] for the purposes of [description of use]. The authors have reviewed and edited the output and take full responsibility for the content of this publication.”

**Conflicts of Interest:** Declare conflicts of interest or state “The authors declare no conflicts of interest.” Authors must identify and declare any personal circumstances or interest that may be perceived as inappropriately influencing the representation or interpretation of reported research results. Any role of the funders in the design of the study; in the collection, analyses or interpretation of data; in the writing of the manuscript; or in the decision to publish the results must be declared in this section. If there is no role, please state “The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results”.

Abbreviations

The following abbreviations are used in this manuscript:

|  |  |
| --- | --- |
| MDPI | Multidisciplinary Digital Publishing Institute |
| DOAJ | Directory of open access journals |
| TLA | Three letter acronym |
| LD | Linear dichroism |

Appendix A

Appendix A.1

The appendix is an optional section that can contain details and data supplemental to the main text—for example, explanations of experimental details that would disrupt the flow of the main text but nonetheless remain crucial to understanding and reproducing the research shown; figures of replicates for experiments of which representative data is shown in the main text can be added here if brief, or as Supplementary data. Mathematical proofs of results not central to the paper can be added as an appendix.

**Table A1.** This is a table caption.

|  |  |  |
| --- | --- | --- |
| **Title 1** | **Title 2** | **Title 3** |
| entry 1 | data | data |
| entry 2 | data | data |
| entry 3 | data | data |
| entry 4 | data | data |
| entry 5 | data | data |
| entry 6 | data | data |
| entry 7 | data | data |

Appendix B

All appendix sections must be cited in the main text. In the appendices, Figures, Tables, etc. should be labeled starting with “A”—e.g., Figure A1, Figure A2, etc.

Notes

1. This is a note example.
2. This is a note example.
3. This is a note example.

References

References must be arranged in alphabetical order of the first author in the Reference list and listed individually at the end of the manuscript. We recommend preparing the references with a bibliography software package, such as EndNote, Reference Manager or Zotero to avoid typing mistakes and duplicated references.

Citations and references in the Supplementary Materials are permitted provided that they also appear in the reference list here.

(Azikiwe & Bello, 2020a) Azikiwe, H., & Bello, A. (2020a). Title of the cited article. *Journal Title*, *Volume*(Issue), Firstpage–Lastpage/Article Number.

(Azikiwe & Bello, 2020b) Azikiwe, H., & Bello, A. (2020b). *Book title*. Publisher Name.

(Davison, 1623/2019) Davison, T. E. (2019). Title of the book chapter. In A. A. Editor (Ed.), *Title of the book: Subtitle* (pp. Firstpage–Lastpage). Publisher Name. (Original work published 1623) (Optional)

(Fistek et al., 2017) Fistek, A., Jester, E., & Sonnenberg, K. (2017, Month Day). *Title of contribution* [Type of contribution]. Conference Name, Conference City, Conference Country.

(Harwood, 2008) Harwood, J. (2008). *Title of the cited article*. Available online: URL (accessed on Day Month Year).

(Hutcheson, 2012) Hutcheson, V. H. (2012*)*. *Title of the thesis* [XX Thesis, Name of Institution Awarding the Degree].

(Lippincott & Poindexter, 2019) Lippincott, T., & Poindexter, E. K. (2019). *Title of the unpublished manuscript* [Unpublished manuscript/Manuscript in preparation/Manuscript submitted for publication]. Department Name, Institution Name.

(Harwood 2008) Harwood, John. 2008. Title of the Cited Article. Available online: URL (accessed on Day Month Year).

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.