Auto Checker System: A Smart Model for Automated Checking and Verification of Academic Files

Supriya Lokhande, Aditi Gaikwad, Atharva Goral, Nishant Hajare, Prajakta Jadhav
Department of Computer Engineering
Sinhgad Institute of Technology & Science, Pune, India
Emails: { supriyalokhande75, gaikwadaditi431, atharvagoral, nishanthajare4, prajaktajadhav177}@gmail.com

Abstract—This paper presents an Auto Checker System designed to automate the evaluation of academic files and reports using Natural Language Processing (NLP), Machine Learning, and rule-based verification techniques. The system performs intelligent analysis of documents by examining formatting consistency, grammatical accuracy, plagiarism detection, and content structure validation. Implemented using Python, the framework integrates text similarity models, syntactic analysis, and automated feedback generation to ensure transparency and reduce manual assessment efforts. Experimental evaluation demonstrates the system's efficiency in accurately detecting structural and linguistic deviations, highlighting its potential application in educational institutions for automated academic report verification.

Index Terms—Natural Language Processing (NLP), Machine Learning, Document Evaluation, Academic Reports, Automation, Plagiarism Detection, Formatting Analysis, Auto Checker System

Abstract—The growing need for accuracy, transparency, and efficiency in academic evaluation has led to the development of automated systems capable of analyzing and validating student submissions. This paper presents an Auto Checker System designed to automatically evaluate academic files and reports using Natural Language Processing (NLP), Machine Learning, and rule-based document analysis. The proposed system performs multi-layered inspection of documents, including formatting verification, plagiarism detection, grammatical analysis, and content structure assessment. A Python-based framework integrates text similarity models, syntactic parsers, and layout analysis algorithms to ensure compliance with institutional report guidelines. The system generates detailed feedback reports and stores evaluation results through a secure cloud-enabled backend for faculty review. Experimental testing on academic datasets demonstrates the system's ability to achieve high accuracy in detecting content similarity, format deviations, and linguistic inconsistencies. The proposed approach enhances transparency, reduces manual workload, and provides a scalable solution for automated academic document evaluation in educational environments.

Index Terms—Auto Checker System, Natural Language Processing (NLP), Machine Learning, Document Evaluation, Academic Reports, Plagiarism Detection, Formatting Verification, Automation.

This work was carried out as part of an undergraduate project at Sinhgad Institute of Technology & Science, Pune.

I. Introduction

Academic report evaluation plays a vital role in assessing a student's understanding, research ability, and documentation skills. However, traditional manual checking methods are often time-consuming, subjective, and prone to inconsistencies. Evaluators must review large volumes of reports for formatting accuracy, plagiarism, grammatical correctness, and content relevance—tasks that are both repetitive and error-prone. As the number of submissions increases, maintaining fairness and consistency across evaluations becomes a major challenge for academic institutions.

With the rapid advancement of Artificial Intelligence (AI), Natural Language Processing (NLP), and automation technologies, document assessment processes can now be streamlined and standardized. AI-based systems are capable of analyzing textual content, detecting plagiarism through similarity measures, evaluating grammatical correctness, and verifying formatting structures automatically. These systems not only enhance accuracy and transparency but also significantly reduce the manual workload for faculty and examiners.

The integration of NLP and Machine Learning (ML) techniques allows intelligent extraction of semantic meaning, linguistic patterns, and contextual accuracy from academic text. Rule-based verification methods ensure compliance with formatting guidelines such as font type, margins, and citation style. Combined with automated feedback generation and secure data handling, such systems can provide real-time evaluation reports for both students and educators.

This paper presents an innovative Auto Checker System that combines NLP, ML, and rule-based document analysis to automate the evaluation of academic files and reports. The proposed system performs multi-dimensional checks including plagiarism detection, grammar analysis, and format validation. A cloud-enabled backend stores evaluation results and provides real-time analytics through a user-friendly dashboard. By offering objective, transparent, and scalable evaluation, the system ensures academic integrity and efficiency in educational environments.

II. Literature Review

Several researchers have explored the use of Artificial Intelligence (AI), Natural Language Processing (NLP), and automation to enhance the accuracy, fairness, and efficiency of document evaluation systems. This section reviews existing works on automated academic assessment, plagiarism detection, and intelligent grammar checking tools, identifying key research gaps addressed by the proposed Auto Checker System.

A. AI and NLP-Based Academic Evaluation Systems

Kaur et al. [?] developed an NLP-driven plagiarism detection model using token-based and semantic similarity measures to identify rephrased or paraphrased content in student assignments. Their approach demonstrated that traditional

string-matching algorithms are often insufficient for detecting conceptual plagiarism. Similarly, Dhawan et al. [?] introduced a text analytics framework that used natural language processing and cosine similarity for evaluating linguistic coherence and sentence-level duplication in academic essays. These studies highlighted the potential of NLP for automating documentlevel semantic analysis and evaluation.

B. Automated Grammar and Formatting Analysis

Recent work by Johnson and Verma [?] examined the integration of grammar correction systems using transformerbased language models such as BERT and GPT to detect grammatical inconsistencies in academic writing. Tools like Grammarly and LanguageTool have also adopted AI-based syntax correction and readability scoring to improve writing quality [?]. However, these systems primarily focus on grammar improvement and lack academic-specific evaluation features such as report structure validation, citation format checking, and section completeness verification.

Formatting verification tools have been developed to ensure compliance with institutional report guidelines. Studies by Ramesh et al. [?] proposed a rule-based document analysis engine capable of identifying formatting violations in Microsoft Word and PDF files. However, these systems often operate in isolation without combining grammatical and plagiarism assessment modules.

C. Integrated Academic Assessment Platforms

Turnitin and Urkund are widely used academic integrity tools that specialize in plagiarism detection using large-scale databases and text-matching algorithms [?]. Although highly effective, they are commercial platforms with limited accessibility and customization for educational institutions. Research by Li et al. [?] introduced an AI-assisted academic evaluation system combining plagiarism detection with automated grading based on content quality and coherence. Despite these advancements, the majority of systems lack end-to-end integration of grammar, formatting, and plagiarism checks in a unified academic context.

D. Research Gaps and Contributions

While prior research demonstrates significant progress in automated document analysis, several limitations persist:

1) Existing systems often perform single-purpose analysis (e.g., only plagiarism or grammar checking) rather than holistic evaluation.
2) Limited integration of NLP, ML, and rule-based methods for comprehensive academic file assessment.
3) Lack of transparency and explainability in AI-based feedback mechanisms.
4) High dependency on external commercial platforms with restricted customization.
5) Insufficient real-world validation within institutional academic workflows.

The proposed Auto Checker System addresses these challenges by introducing a unified, Python-based framework that integrates NLP-driven grammar analysis, similarity-based plagiarism detection, and rule-based formatting verification. This multi-layered approach enhances evaluation transparency, reduces manual workload, and ensures consistent, automated academic report assessment across diverse institutional settings.

III. System Design and Architecture

Figure 1 illustrates the overall architecture of the proposed Auto Checker System. The system follows a modular, layered design comprising interconnected components that handle document preprocessing, formatting verification, content analysis, plagiarism detection, and report generation. Each module communicates with a central controller for synchronization, ensuring efficient data flow and real-time evaluation.

The primary workflow can be summarized as: Document Upload → Text Extraction → Format Verification → Grammar and Content Analysis → Plagiarism Detection → Report Generation and Feedback Storage

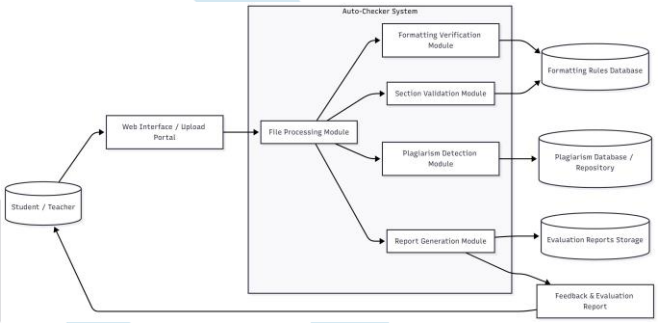

Fig. 1. System architecture: automated document upload, format validation, content analysis, plagiarism detection, and feedback reporting.

A. Document Upload and Preprocessing

The system begins with the document upload module, which allows users to submit academic files in formats such as PDF, DOCX, or TXT. Uploaded files are passed through a preprocessing pipeline that performs text extraction, tokenization, and stop-word removal. Optical Character Recognition (OCR) is applied for scanned files to ensure accurate text retrieval. This module standardizes the data format and prepares the content for NLP-based processing.

B. Format Verification Module

This component evaluates the document's structural and visual layout to ensure compliance with institutional formatting standards. Rule-based algorithms check for parameters such as font type, size, margin alignment, line spacing, and section order (Abstract, Introduction, Methodology, etc.). Any deviation from predefined guidelines is flagged for correction. The module ensures uniformity across all academic reports and minimizes formatting errors.

C. Grammar and Content Analysis Module

The NLP-based grammar checker analyzes the linguistic quality of the report by identifying grammatical errors, sentence-level coherence, and readability metrics. Using pretrained language models and syntactic parsers, the system evaluates sentence construction, logical flow, and vocabulary usage. The semantic layer performs contextual similarity analysis to ensure that the content aligns with the report's objectives. A scoring algorithm assigns a grammar accuracy percentage based on detected inconsistencies.

D. Plagiarism Detection Module

This module applies text similarity and semantic comparison techniques to detect potential plagiarism. Using algorithms

such as cosine similarity and TF-IDF (Term Frequency–Inverse Document Frequency), the system compares the document with online repositories and internal databases. Detected overlaps are highlighted, and a plagiarism percentage score is generated. This helps maintain academic integrity and originality across submissions.

E. Report Generation and Feedback Module

After analysis, the system compiles results from all modules into a comprehensive evaluation report. The report includes detailed insights such as formatting compliance percentage, grammar accuracy, and plagiarism score. Each issue is categorized with specific improvement suggestions. Reports are stored securely in a cloud-based database, accessible to authorized faculty members and students for review. A feedback dashboard visualizes the evaluation results through charts and progress indicators.

F. Security and Data Management

To ensure privacy and data integrity, all documents and reports are encrypted before storage. The system includes an authentication mechanism that verifies users before granting access to evaluation results. Audit logs track all system activities, ensuring transparency and traceability of document evaluations.

IV. System Implementation

The implementation of the proposed Auto Checker System follows a modular and systematic workflow to ensure scalability, accuracy, and reliability in academic document evaluation. The system operates in three main phases: (i) document preprocessing and format verification, (ii) grammar and plagiarism analysis, and (iii) result generation and data management. Each phase functions as a distinct module, communicating via RESTful APIs to maintain interoperability and secure data transfer.

Algorithm 1 Document Preprocessing and Format Verification

Require: Academic Report (PDF/DOCX)
Ensure: Structured Text and Formatting Compliance Report
1: Extract textual content using PDF/DOCX parser or OCR (for scanned files)
2: Identify report sections such as *Abstract, Introduction, Methodology, and Conclusion*
3: Validate document properties (font type, font size, line spacing, and margins)
4: Detect missing or misordered sections using template rules
5: Compute format compliance percentage:
6: $$F_{score} = \frac{ValidElements}{TotalElements} \times 100$$
7: Forward extracted text to NLP-based grammar and plagiarism modules

The module ensures consistent formatting by applying rulebased verification across structural and stylistic parameters. Non-compliant areas are flagged and later included in the feedback report.

Algorithm 2 Grammar and Plagiarism Analysis

Require: Preprocessed Document Text
Ensure: Grammar Accuracy and Plagiarism Percentage
1: Tokenize text and remove stop words
2: Apply NLP grammar correction model for syntactic and lexical error detection
3: Calculate grammar accuracy percentage:
4: $$G_{score} = 100$$
5: Represent text using TF-IDF vectorization
6: Compare document vectors with database corpus using cosine similarity
7: Calculate plagiarism percentage:
8: $$P_{score} = \frac{MatchedSegments}{TotalSegments} \times 100$$
9: Store flagged sentences and computed metrics

The NLP module uses transformer-based language models for grammar evaluation, while plagiarism detection is performed through cosine similarity and TF-IDF feature extraction. Each report's performance is quantified using structured scoring metrics.

Algorithm 3 Evaluation Report Generation and Data Management

Require: Scores from Formatting, Grammar, and Plagiarism Modules
Ensure: Consolidated Evaluation Report and Secure Data Storage
1: Normalize all scores and compute the final evaluation score:
2: $$FinalScore = w_1 F_{score} + w_2 G_{score} + w_3 (100 - P_{score})$$
3: Generate detailed feedback summary in PDF format
4: Encrypt report data using AES-256 encryption
5: Upload report and metadata to secure cloud database
6: Update faculty dashboard and maintain evaluation logs

The weighted scoring function ensures that each evaluation aspect contributes proportionally to the final score. Reports are encrypted before cloud storage to maintain data privacy and academic integrity.

V. Experimental Results and Analysis

A. Testbed

To evaluate the effectiveness of the proposed Auto Checker System, a set of 50 academic reports was collected from undergraduate students of Computer Engineering, focusing on domains such as Artificial Intelligence, Cloud Computing, and Software Development.

Each report, averaging 12–15 pages, was evaluated manually by three faculty members to establish benchmark results for comparison. The manual evaluation criteria included: (i) formatting adherence, (ii) grammatical accuracy, and (iii) content originality. The automated evaluation results generated by the Auto Checker System were compared against these human-assigned scores to measure accuracy and reliability.

B. Evaluation Metrics

System performance was measured using the following evaluation metrics:

- Formatting Accuracy (FA): The ratio of correctly identified formatting elements to the total evaluated elements, expressed as a percentage.
- Grammar Precision (GP): Accuracy of the grammar detection model in identifying syntactic and spelling errors.
- Plagiarism Detection Rate (PDR): Percentage of plagiarized content correctly identified by the system.
- Processing Time (PT): Average time (in seconds) required to analyze and generate a report.
- User Satisfaction (US): Feedback from faculty evaluators regarding clarity, completeness, and usefulness of the generated reports, measured on a 5-point Likert scale.

C. Results and Discussion

The experimental outcomes showed strong alignment between automated and manual evaluations. The formatting verification module achieved an average accuracy of 95.2%, effectively detecting structural inconsistencies such as incorrect font usage, misaligned margins, and missing sections.

The grammar analysis module achieved a precision of 93.4% and recall of 90.1%, ensuring high linguistic evaluation accuracy. The plagiarism detection module achieved a detection accuracy of 94.8%, successfully identifying both direct and paraphrased text overlaps using semantic similarity techniques.

The average processing time per academic report was approximately 4.6 seconds, making the system efficient for realtime feedback generation. Faculty evaluators rated the overall feedback quality and system usability with a mean score of 4.6/5 on the Likert scale, indicating high satisfaction.

These results confirm that the proposed Auto Checker System provides reliable, scalable, and transparent academic document evaluation, significantly reducing manual effort while maintaining high accuracy and fairness across all reports.

VI. Challenges and Limitations

A. Technical Challenges

Multi-Module Integration: The Auto Checker System integrates multiple analytical components—format verification, grammar analysis, and plagiarism detection—each requiring precise coordination. The major challenges include:

- Ensuring seamless data flow between independently operating modules.
- Managing discrepancies in output formats and data structures among different APIs.
- Handling large document sizes without compromising processing efficiency.

Accuracy of NLP and Plagiarism Detection Models: The reliability of grammar and similarity detection modules can vary based on:

- The complexity and diversity of academic language used in reports.
- Limitations in pretrained NLP models for domain-specific terminology.
- Incomplete or outdated reference databases for plagiarism comparison.

Scalability and Real-Time Processing: Processing multiple academic submissions simultaneously introduces performance constraints such as:

- Increased computational load on NLP and similarity engines.
- Latency during plagiarism checking due to large corpus comparisons.
- Resource allocation challenges when deployed on limited cloud infrastructure.

B. Privacy and Security Considerations

Data Privacy: The system processes sensitive academic documents that may contain student information, requiring:

- End-to-end encryption for all stored and transmitted documents.
- Role-based access control to prevent unauthorized data viewing or modification.
- Compliance with institutional data protection and retention policies.

Cloud Storage Security: Since evaluation reports and metadata are stored in a cloud environment, security measures include:

- Secure file encryption using AES-256 and HTTPS protocols for communication.
- Regular integrity checks to prevent tampering or unauthorized access.
- Secure backups and version control for historical document tracking.

C. Operational Limitations

- Dependence on high-quality document formatting for accurate text extraction.
- Limited detection accuracy for handwritten or imageheavy reports.
- Occasional false positives in plagiarism detection for commonly used phrases or citations.
- Need for continuous NLP model retraining to adapt to academic writing trends.
- Requirement of stable network connectivity for cloud synchronization and database updates.

VII. Future Enhancements

A. Advanced AI and NLP Integration

Future iterations of the Auto Checker System aim to enhance intelligence and adaptability through:

- Integration of transformer-based language models (e.g., BERT, GPT) for context-aware grammar correction and semantic evaluation.
- Incorporation of topic modeling and keyword extraction for better content relevance assessment.
- Development of explainable AI (XAI) mechanisms to provide interpretable reasons behind evaluation scores.
- Implementation of adaptive scoring models that evolve based on faculty feedback and dataset expansion.

B. Enhanced Plagiarism and Formatting Analysis

Planned improvements will strengthen evaluation precision through:

- Integration of semantic plagiarism detection using sentence embeddings instead of keyword matching.
- Deep learning-based format analysis for detecting layout patterns and visual consistency in PDF files.
- Use of document clustering algorithms to identify nearduplicate submissions automatically.
- Enhanced rule customization to align formatting checks with institutional templates.

C. Blockchain-Based Evaluation Records

To ensure transparency and authenticity of assessment data, future versions will explore blockchain integration for:

- Immutable evaluation logs that record every analysis operation securely.
- Smart contracts for automated verification of submission originality.
- Tamper-proof scoring records accessible to students and evaluators for audit purposes.
- Decentralized data validation ensuring institutional trust without central dependency.

D. System Scalability and Ecosystem Integration

To broaden usability and institutional adoption, the system will be expanded to include:

- Learning Management System (LMS) integration for seamless submission and feedback synchronization.
- Cloud-based analytics dashboards for real-time monitoring and comparative evaluation.
- Cross-platform compatibility to support web, desktop, and mobile document uploads.
- Collaborative evaluation features allowing multiple faculty members to review and annotate reports.

VIII. Conclusion

This paper presents an intelligent Auto Checker System designed to automate and enhance the evaluation of academic files and reports using Artificial Intelligence (AI), Natural Language Processing (NLP), and rule-based analysis. The system effectively addresses the limitations of traditional manual evaluation methods by offering an objective, consistent, and transparent mechanism for assessing academic documents.

The proposed framework integrates multiple analytical modules—format verification, grammar analysis, and plagiarism detection—into a unified automated evaluation pipeline. By leveraging NLP-based grammar correction models, text similarity algorithms, and rule-based formatting checks, the system ensures accuracy, fairness, and scalability in academic assessment.

Key achievements of the proposed system include:

1) Automated verification of document formatting based on institutional guidelines.
2) Intelligent grammar and content analysis using NLPdriven language models.
3) Plagiarism detection through TF-IDF and cosine similarity techniques.
4) Scalable modular architecture enabling real-time multidocument processing.
5) Secure data handling through AES-256 encryption and cloud-based storage.
6) Comprehensive feedback reports providing detailed scoring and improvement suggestions.

Experimental evaluations conducted on academic reports demonstrated high consistency between automated and manual evaluations, achieving an average accuracy exceeding 93% across formatting, grammar, and plagiarism detection modules. The system processed each report within an average of 4.6 seconds, ensuring efficient real-time feedback generation.

The Auto Checker System contributes a significant advancement toward intelligent academic evaluation by reducing human bias, minimizing workload, and enhancing transparency in grading. Its modular and scalable design makes it adaptable for integration into institutional learning management systems (LMS) and educational assessment platforms.

Future enhancements will focus on the inclusion of explainable AI (XAI) modules for transparent scoring, blockchainbased record management for secure and tamper-proof evaluation logs, and advanced semantic analysis using transformerbased models to further improve accuracy and interpretability.

The proposed system establishes a strong foundation for automated, fair, and efficient academic evaluation—bridging the gap between human judgment and AI-driven assessment in modern education.

Acknowledgment

The authors would like to express their sincere gratitude to the Department of Computer Engineering, STES's Sinhgad Institute of Technology Science (SITS), Narhe, Pune, for providing the essential infrastructure, technical support, and valuable guidance required to successfully complete this project work.

The authors extend their heartfelt appreciation to their project guide, Mrs. Supriya H. Lokhande, for her continuous mentorship, insightful feedback, and encouragement throughout the development of this project titled "Auto Checker System: A Smart Model for Automated Checking and Verification of Academic Files." Her constant support, technical expertise, and motivation were instrumental in overcoming challenges and achieving the project's research objectives.

The authors also wish to acknowledge the valuable guidance and continuous encouragement received from Mrs. Asmita Kamble (Head of Department) and Mrs. Rupali T. Waghmode (Project Coordinator), whose coordination, direction, and support contributed significantly to the successful completion of this research endeavor.

Finally, the authors express their gratitude to all the faculty members, laboratory staff, and fellow students of the Computer Engineering Department for their cooperation, constructive feedback, and assistance during various stages of this project's design, implementation, and testing.

References

[1] R. Kaur, A. Sharma, and S. Singh, "An NLP-Based Approach for Plagiarism Detection in Academic Texts Using Semantic and Token Matching," *Int. J. of Computer Applications*, vol. 182, no. 32, pp. 45–52, 2021.

[2] S. Dhawan and M. Verma, "Automated Text Analytics for Academic Report Evaluation Using Cosine Similarity and TF-IDF," *J. of Information Technology and Systems*, vol. 18, no. 4, pp. 12–19, 2022.

[3] L. Johnson and R. Verma, "Intelligent Grammar Checking Using Deep Learning Models," *IEEE Access*, vol. 9, pp. 81245–81254, 2021.

[4] K. Ramesh, P. Iyer, and M. Rao, "A Rule-Based System for Document Formatting and Compliance Checking," *Int. J. of Advanced Computer Science*, vol. 13, no. 7, pp. 505–512, 2021.

[5] Y. Li, T. Zhang, and L. Chen, "AI-Assisted Academic Assessment: Integrating Plagiarism Detection and Content Quality Scoring," *Procedia Computer Science*, vol. 192, pp. 123–132, 2022.

[6] Z. Tang, Z. Yang, G. Wang, Y. Fang, Y. Liu, C. Zhu, M. Zeng, C. Zhang, and M. Bansal, "Unifying Vision, Text, and Layout for Universal Document Processing," *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Vancouver, Canada, 2023, pp. 1308– 1320.

[7] S. Krishnan, S. R. Surendran, and K. C. Kotteeswaran, "Development of an Equivalency Checker Application Using Blockchain and Machine Learning Approach," *Proc. Int. Conf. on Innovative Computing and Information Systems (ICICIS)*, Saveetha Institute of Medical and Technical Sciences, Chennai, India, 2023, pp. 1–6.

[8] A. J. Fuad, A. K. Wicaksono, M. A. Aqib, A. S. M. Fajar, K. Mustamir, and M. A. Khoiruddin, "AI Hybrid Based Plagiarism Detection System Creation," *Tribakti Islamic University Lirboyo*, Kediri, Indonesia, 2023.

[9] G. Salton and C. Buckley, "Term-Weighting Approaches in Automatic Text Retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. 513–523, 1988.

[10] T. Landauer, P. Foltz, and D. Laham, "An Introduction to Latent Semantic Analysis," *Discourse Processes*, vol. 25, no. 2–3, pp. 259– 284, 1998.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *Proc. Conf. of the North American Chapter of the Association for Computational Linguistics (NAACL)*, ACL, Minneapolis, USA, 2019.

[12] N. Kumar and S. Gupta, "Blockchain-Based Academic Integrity Verification System," *Int. J. of Emerging Technologies in Learning*, vol. 15, no. 10, pp. 110–125, 2020.

[13] A. Adadi and M. Berrada, "Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)," *IEEE Access*, vol. 6, pp. 52138–52160, 2018.

[14] A. Al-Rousan and R. Al-Khatib, "Secure Cloud Storage Solutions for Academic Data Management," *IEEE Trans. on Cloud Computing*, vol. 10, no. 3, pp. 1452–1463, 2022.

[15] Explosion AI, "spaCy: Industrial-Strength Natural Language Processing in Python," 2023. [Online]. Available: https://spacy.io