Neural Intelligence in Ayurveda: A Deep Learning-Driven Ecosystem for Herbal Discovery and Trade

Asavari Desai[1], Riya Desai[2], Sakshi Powale[3], Swati Sonavane[4], Dr. Divya Tamma[5]

[1,2,3,4] UG Scholar, Dept. of AI&DS, Rajiv Gandhi Institute of Tech., Mumbai, Maharashtra, India

[5] Assistant Professor, Dept. of AI&DS, Rajiv Gandhi Institute of Tech., Mumbai, Maharashtra, India

[1]asavaridesai325@gmail.com, [2]riadesai3110@gmail.com, [3]strategic.sakshi@gmail.com, [4]truptisonavane847@gmail.com, [5]divya.tamma@mctrgit.ac.in

***Abstract*: Medicinal plants play a crucial part in the Ayurvedic medical system,effective methods for information retrieval and identification are essential. This research intends to design a deep learning-based model employing Convolutional Neural Networks (CNNs) for the real-time identification of medicinal plants. The project strives to create an accessible and user-friendly platform. that not only identifies plants through image recognition but also provides comprehensive information about various Ayurvedic botanicals. In addition to plant identification, the system features a knowledge base that offers detailed insights into the properties and uses of these plants. An integrated e-commerce component allows users to purchase Ayurvedic products directly, enhancing accessibility to herbal remedies. By utilizing deep learning techniques, our project seeks to enhance Ayurvedic research, ensuring easy access to valuable information and products while encouraging the use of medicinal plants in holistic health practices.**

Keywords: Plant information retrieval, Medicinal plant identification, Deep learning, Convolutional Neural Networks (CNN), E-commerce, Ayurvedic products.

I. INTRODUCTION

The Ayurvedic medical system, which has its origins in ancient Indian practices, promotes a holistic approach to health through the utilization of therapeutic herbs. As interest in natural remedies continues to rise globally, the need for precise and reliable information about these plants has become increasingly important. However, the overwhelming amount of information available online can make it difficult to find relevant and trustworthy data. Conventional search engines often fall short when handling specialized queries related to Ayurvedic medicine, highlighting the necessity for more effective solutions. To address these challenges, our research aims to create a comprehensive platform that employs deep learning techniques for the real-time identification of medicinal plants using Convolutional Neural Networks (CNNs). This innovative platform allows users to identify Ayurvedic plants by simply uploading an image, providing immediate access to detailed information about each species.

In addition to plant identification, our system features an advanced search engine and a dedicated knowledge page that offers extensive insights into the characteristics and applications of various medicinal herbs. This ensures that users can quickly and efficiently access high-quality information tailored to their needs.

Furthermore, the platform includes an e-commerce component that enables users to buy and sell Ayurvedic products directly. This functionality not only enhances accessibility to herbal remedies but also supports sustainable practices within the herbal medicine community. Through this research initiative, we strive to develop a user-friendly resource that enriches knowledge about Ayurveda while facilitating easy access to medicinal plants and related products for a wider audience.

II. LITERATURE SURVEY

The increasing global demand for natural and Ayurvedic medicine has emphasized the necessity of accurate and efficient medicinal plant identification. However, the complexity of botanical classification, combined with the limitations of traditional search engines, makes it challenging to retrieve reliable information on medicinal plants. The Sanjeevani project addresses these challenges by leveraging deep learning, focused web crawling, and e- commerce integration to create an innovative solution for medicinal plant identification and commerce.

This literature survey aims to examine existing methodologies, algorithms, and frameworks relevant to the Sanjeevani system. By analyzing focused web crawlers, machine learning techniques, and e- commerce models, this survey provides a foundation for understanding current research gaps and how the Sanjeevani platform advances the field. The study also highlights setbacks encountered in current approaches, proposing potential solutions that enhance the system's scalability, accuracy, and efficiency.

Nisha Pawar and Dr. K. Rajeswari's study [1], focuses on creating a targeted web crawler that will recover web pages about medicinal plants and the illnesses they cure. Motivating the research is the abundance of unstructured online material about medicinal plants, especially in relation to the plantbased medicines used in the Indian Ayurvedic system. Conventional search engines, however, frequently have trouble effectively filtering and retrieving pertinent content in this field. To address these issues, the authors suggest a tailored web crawler that produces more concentrated search results by navigating the medicinal

plants domain.
The proposed crawler's methodology is built upon a manually developed medicinal plant thesaurus and the Naive Bayes classification algorithm. The Naive Bayes method evaluates the relevance of each web page by analyzing four key lexical features: title text, anchor text, meta description, and URL tokens. The system begins with a collection of seed URLs and expands by following links. Ranking web pages according to their relevance to the user's query, cosine similarity is applied to these features. Additionally, query expansion significantly enhances the accuracy and precision of search results by linking user queries to relevant keywords in the thesaurus, enabling the crawler to perform searches more effectively.
The experimental findings show how successful this system is. The crawler used the Naive Bayes classification algorithm to retrieve relevant web pages with great precision, achieving a classification accuracy of 90%. When compared to other algorithms, such as decision trees and multilayer perceptrons, Naive Bayes performed better. Furthermore, using a manual thesaurus to expand queries increased retrieval accuracy by 20% when compared to general-purpose crawlers. The paper emphasizes the potential for further development, suggesting the incorporation of more advanced machine learning algorithms and the expansion of the thesaurus to cover a broader range of medicinal plants and diseases, laying the groundwork for future research in specialized information retrieval systems.
The study [2] describes TRES (Tree REinforcement Spider), an advanced system for optimizing focused web crawling. Focused crawlers seek out appropriate web pages on a certain domain while avoiding non-relevant ones. The crawling environment is modelled as a Markov Decision Process (MDP) by the authors, who employ Reinforcement Learning (RL) to improve the crawler's decision-making strategies. Tree-Frontier, an innovative decision-tree-based method, is employed in the framework to efficiently manage the enormous state and action spaces associated with focused crawling. Tree-Frontier's discretization of the state-action space allows the RL agent to evaluate fewer actions at each step, considerably lowering computing complexity while maintaining a high harvest rate (the proportion of relevant pages retrieved).
The TRES framework incorporates several innovative components, including a keyword expansion strategy and a deep learning classifier, KwBiLSTM, for page relevance evaluation. Starting with a small set of seed keywords, the framework uses word embeddings to expand this set, improving its ability to guide the crawler. The KwBiLSTM classifier predicts the relevance of web pages by leveraging semantic features and keyword appearance, serving as the reward function in the RL setup. This combination of RL, keyword expansion, and a deep classifier allows TRES to prioritize URLs effectively and achieve better results than traditional methods that rely solely on classifiers or brute-force evaluations.
Experimental results show that TRES surpasses state-of-the-art crawlers like ACHE and SeedFinder in terms of both harvest rate and computational efficiency. Tested across domains such as Sports, Food, and Hardware, TRES achieved superior performance with fewer computational resources by using Tree-Frontier to reduce the number of evaluated actions per step. Its ability to adapt to different environments, coupled with the effectiveness of its keyword expansion and classification components, positions TRES as a highly efficient solution for focused web crawling, setting a new standard in the field.
The research [3] describes a novel focused web crawler that uses deep convolutional networks (CNN) to parse Turkish language accurately. The architecture is organized into numerous parallel components, including the control, crawler, link extractor, link sorter, and natural language processing (NLP) units. This framework facilitates the efficient crawling and processing of large amounts of Turkish web information. The natural language processing unit, driven by CNN, can accurately classify Turkish online pages, outperforming seven cutting-edge CNN models for text categorization. Notably, the web crawler can extract and sort out links, hence enhancing the classification accuracy of Turkish text by incorporating the most appropriate word embeddings, such as Bidirectional Encoder Representations from Transformer.
In terms of technique, the crawler was tested with three datasets: One consists of 50,000 Turkish web pages retrieved by the crawler, along with two publicly available datasets with 28,567 and 22,431 pages, respectively. To improve classification, the paper investigates numerous sophisticated embedding approaches, with BERT proving the most effective for Turkish text. This finding emphasizes the need of proper word representation for web page classification. The CNN-based technique was tested against other models such as VdCNN, CharCNN, and TurkishCNN, demonstrating the proposed system's resilience, particularly its ability to manage noise and the dynamic nature of Turkish web material. The research emphasizes that preprocessing methods like as stemming, lemmatization, and normalization helped to significantly improve performance.
The contribution of this study lies in filling a crucial gap in Turkish web processing, providing a focused crawler specifically tailored for the language. This system could enhance Turkish search engines and contribute to the creation of Turkish-specific datasets. Furthermore, the application of parallel processing methods to improve scalability and processing speed makes it an adaptable tool for crawling huge Turkish web data. The report also identifies potential areas for further research, such as processing dynamic information and supporting AJAX, which could enhance the system's capabilities.
In order to guide topical web crawlers, Gautam Pant and Padmini Srinivasan's paper [4] compares three distinct classification schemes: Support Vector Machines (SVM), Neural Networks (NN), and Naive Bayes (NB). These crawlers seek out relevant webpages based on a certain topic, using classification algorithms to prioritize which URLs to follow. The paper tackles the absence of systematic comparisons of various systems by running experiments on over 100 topics to evaluate their performance in a dynamic web context.
The research focuses on three classifiers: Support Vector Machines, Naive Bayes, and Neural Networks. For Naive Bayes, two versions were tested using normal density estimation and kernel density estimation. For SVM, three polynomial kernels (1st, 2nd, and 3rd-degree) were evaluated, while different architectures of Neural Networks were tested using varying numbers of hidden nodes and learning rates. Additionally, the study examines the effectiveness of these classifiers in situations with limited training

data (seed URLs) and whether adaptation during crawling, by adding pseudoexamples, improves performance.

The major findings show that Naive Bayes performs poorly when compared to Neural Networks and SVM due to skewed posterior probabilities, limiting its capacity to efficiently rank URLs. In contrast, SVM and Neural Networks outperformed Naive Bayes, with SVM having a modest edge in training efficiency. Although SVM and Neural Networks performed similarly in terms of precision and recall, they explored different regions of the web and returned different groups of relevant pages.

When crawlers were initialized with fewer seed URLs, their performance fell, but Neural Networks and SVM surpassed Naive Bayes, exhibiting improved generalization with limited training dataset. The researchers also investigated adaptive crawling by retraining classifiers with pseudoexamples, however this method did not significantly increase performance. The findings indicate that the pseudo-examples may not have been sufficiently informative, or that the classifiers did not benefit from retraining throughout the crawl.

In conclusion, the reviewed studies highlight significant advancements in the formation of focused web crawlers and machine learning methods for detecting and retrieving relevant medicinal plant data. From Nisha Pawar and Dr. K. Rajeswari's use of Naive Bayes classification and thesaurus expansion to enhance search accuracy, to TRES's integration of reinforcement learning and deep learning classifiers for optimized crawling, each approach contributes to improving the efficiency and precision of web crawling in specialized domains.

The incorporation of deep convolutional networks for language-specific crawlers further demonstrates the potential for specialized, high-performance systems. Additionally, the comparative analysis of classification algorithms like Neural Networks, SVM, and Naive Bayes emphasizes the importance of selecting the most suitable method based on the crawling context. Despite these advances, challenges remain in scalability, adaptability, and the ability to handle large, dynamic datasets, which offer opportunities for further research and development in this field.

III. METHODOLOGY

The Sanjeevani system effectively bridges the gaps identified in the literature survey by integrating deep learning, AI-powered web crawling, and e- commerce authentication to enhance medicinal plant identification and retrieval. Traditional focused web crawlers, such as Naïve Bayes-based classifiers and RL-based TRES models, struggle with adaptability, scalability, and computational efficiency. Sanjeevani overcomes these limitations by using SBERT-based text similarity ranking, and CLIP-powered image- text relevance scoring. Unlike static keyword-based methods, this approach enables semantic adaptability, ensuring more precise and context- aware retrieval of medicinal plant data. Moreover, the integration of CNNs for plant image classification significantly improves real-time plant identification, addressing the lack of accurate AI- powered recognition systems in previous research. The hybrid model combining image and text analysis reduces misclassification risks, allowing users to confidently identify and purchase medicinal plants.

Beyond improving retrieval and classification, Sanjeevani also enhances trust in e-commerce transactions, an area where existing platforms like fall short. By implementing AI-driven seller authentication and image-based verification, Sanjeevani ensures that only correctly identified medicinal plants are listed for sale, minimizing fraudulent or misidentified listings. Furthermore, the community knowledge-sharing feature fosters an interactive platform where experts and users can discuss medicinal properties, creating a bridge between traditional Ayurvedic wisdom and AI- driven solutions. Unlike previous models, which suffer from high computational costs and dataset limitations, Sanjeevani optimizes efficiency using MiniLM for fast text ranking and cloud-based adaptive crawling, allowing for continuous dataset expansion and improved accuracy over time. This synergy of AI, deep learning, and e-commerce integration makes Sanjeevani a transformative innovation in the field of medicinal plant research, retrieval, and commerce. This double-moving approach ensures both visual and textual interpretation of textual data. The goal is to provide fast, intelligent search results tailored to the user's intentions. Finally, Sanjeevani creates a seamless way to ensure product acquisition from system identification.

A. System Architecture

The Sanjeevani architecture comprises three core modules: Plant Information Retrieval, Plant Identification, and User Interface. Figure 1 illustrates a framework for gathering and categorizing plant data for easy access and identification. It describes a two-pronged approach that is initiated by a user's query. The "plant information retrieval" section explains how a user's text-based query initiates both image retrieval (using an unsplash api and subsequent ranking with a clip model) and web search (followed by result ranking and image query explanation). Simultaneously, the "plant identification" section demonstrates a process where a plant leaf image is preprocessed and then fed into a convolutional neural network for identification against a plant leaf dataset. Finally, the "user interface" section demonstrates how a user-uploaded plant image is analyzed by an identification model to determine the plant's name, which is then utilized to display pertinent plant information retrieved from the initial part of the system. This architectural design aims to create a comprehensive plant information system that can be accessed through both textual queries and image inputs.

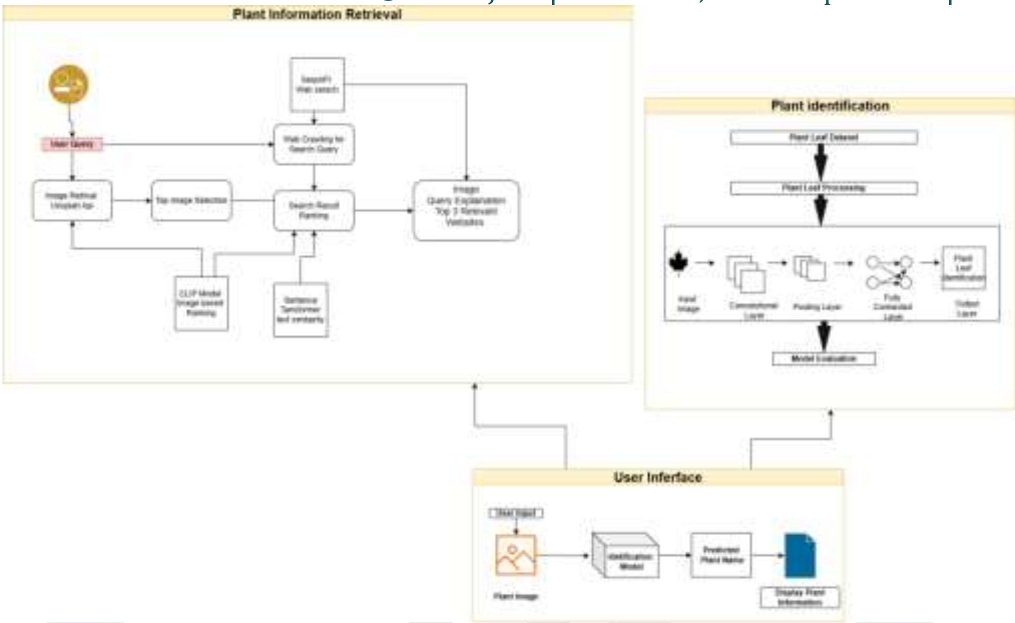

Fig. 1 Plant Information Retrieval

1) Plant Information Retrieval: The plant information retrieval model consists of the following steps:

- User Query: Receives a text-based query related to a plant.
- Image Retrieval (Unsplash API): Fetches high-quality plant images.
- Top Image Selection: Selects the most relevant images.
- CLIP Model (Image-Based Ranking): Ranks images based on relevance.
- SerpAPI Web Search: Conducts a web search for plant-related information.
- Web Crawling: Extracts valuable plant- related data from websites.
- Search Result Ranking: Uses a Sentence Transformer model to rank results.
- Image Query Explanation & Top 3 Websites: Summarizes and presents the top three relevant websites for plant- related details.

2) Plant Identification (Deep Learning): The plant identification model consists of the following steps:

- Dataset Preparation: Uses a Plant Leaf Dataset.
- CNN-Based Classification:
- Convolutional + Pooling Layers extract image features.
- Fully Connected Layer performs classification.
- Softmax Output Layer predicts the medicinal plant species.
- Model Evaluation: 77.8% accuracy on test data.
- Improved performance under variable lighting conditions.

3) User Interface & E-Commerce Integration: The plant identification model consists of the following steps:

- User uploads an image, and the system predicts the plant name.
- Retrieves relevant web pages and links.
- E-commerce page: Allows users to register, browse, and purchase products.
- Manages product listings, transactions.

B. Components of the Project and Technologies Used

1) User Interface (UI) and Interaction System: The user interface (UI) acts as the primary access point, enabling users to interact with the system through a web or mobile application. The UI enables functionalities such as uploading plant images, searching for plant information, and accessing the e-commerce marketplace. A responsive and intuitive interface is essential to ensure smooth user experience.

The front-end is developed using React.js providing dynamic and interactive components. It is connected to the backend through RESTful APIs, enabling seamless communication with the database and machine learning models. The backend is built using Flask, handling requests, processing images, and retrieving plant-related data. The UI also includes an image upload system, which forwards the image to the identification model for classification. Furthermore, search queries related to plant information are passed to the retrieval module for fetching relevant data. The UI supports user authentication, allowing registered users to buy or sell plant products in the e-commerce system.

2) Plant Identification Module: A deep learning-based medicinal plant identification model leveraging Convolutional Neural Networks (CNNs) is proposed to effectively classify plant species from images. The system achieves robust performance through image preprocessing, data augmentation, and an optimized CNN architecture. The results demonstrate high accuracy, making the model suitable for real-world applications in botany and healthcare.

Medicinal plants have been extensively used in traditional and modern medicine. However, manual identification is prone to errors due to morphological similarities among species. To enhance classification accuracy and scalability, an automated plant identification system is developed using a deep learning approach. A structured medicinal plant image dataset was utilized, comprising images labeled into different species. The images were preprocessed using resizing (128x128 pixels), normalization, and data augmentation techniques to enhance model generalization.

The CNN model was developed using TensorFlow/Keras and comprises multiple convolutional, pooling, and dense layers. Its key architectural components include Conv2D layers for extracting spatial features, MaxPooling layers to reduce dimensionality while preserving important information, Fully Connected layers for classification, and activation functions like ReLU for feature extraction and softmax for multi-class classification. The dataset was split into training and testing sets at an 80:20 ratio. The training process utilized categorical cross-entropy loss, the Adam optimizer, and early stopping to mitigate overfitting. The model's performance was assessed based on accuracy, precision, recall, and F1-score.

The model Assessed high classification accuracy, demonstrating effective learning of medicinal plant features. CNNs provide a scalable and efficient method for automated plant identification. Challenges such as class imbalance and image variations were mitigated through data augmentation and hyperparameter tuning.

3) Information Retrieval Module: The AI Agent serves as a crucial component in an intelligent plant information retrieval and identification system. This component is designed to enhance user experience by efficiently retrieving relevant images and textual information related to plants. It utilizes advanced machine learning models, incorporating Computer Vision techniques and Natural Language Processing (NLP), to analyze user queries and provide ranked search results. The primary functions of this AI Agent include retrieving high-quality plant images, ranking websites based on their content relevance, and generating well-structured explanations to summarize search results. By integrating multiple APIs and deep learning models, the AI Agent ensures that the retrieved information is both contextually relevant and visually representative of the searched plant.

The AI Agent primarily leverages machine learning models to perform text and image-based ranking. The Sentence Transformer model (all-MiniLM-L6-v2) is employed to analyze and compare text by converting search queries and web content into high-dimensional vector representations. This enables the system to rank search results according to their semantic relevance to the user’s query. Furthermore, OpenAI’s CLIP (Contrastive Language–Image Pretraining) model is integrated to match textual queries with image embeddings. This model enables the AI Agent to determine how closely an image matches the given search term, ensuring that only the most relevant images are presented to users.

The AI Agent retrieves data from multiple online sources using various APIs. SerpAPI (Google Search API) is employed to fetch search results from Google, which are then processed and ranked based on relevance. To provide high-quality plant images, the system uses the Unsplash API, which allows it to search for free stock images related to the given query. The images retrieved from Unsplash are ranked using the CLIP model to determine their contextual accuracy. Additionally, the AI Agent uses the Requests library in Python to communicate with these APIs and fetch data efficiently.

To rank and display relevant plant images, the AI Agent first constructs a query based on the user’s input and fetches the top 10 images from Unsplash. Each image is then scored using the CLIP model, which compares the image embedding with the textual query embedding. The image with the highest score is selected, downloaded, and displayed along with attribution to the photographer. This ensures that users receive visually relevant and high-quality plant images that correspond to their search queries.

The textual search and ranking system follow a structured process. When a user searches for plant- related information, the AI Agent queries Google Search through SerpAPI and retrieves organic search results. Each search result snippet is then processed using the Sentence Transformer model, which calculates a similarity score between the snippet and the original query. Additionally, the AI computes a CLIP score to evaluate the relevance of the search result in terms of both text and visual content. These scores are combined using a weighted formula—70% text similarity and 30% CLIP score—to rank the websites effectively. The AI Agent presents the highest-ranked search results, ensuring users receive the most relevant and credible information available online.

One of the most critical features of the AI Agent is its ability to generate query explanations. After ranking search results, the system extracts key information from the top-ranked snippets and structures them into a cohesive explanation. This explanation is formatted with a topic header and summary, ensuring that users receive a clear and concise response to their query. By summarizing multiple sources, the AI Agent provides a more comprehensive understanding of the searched topic without requiring users to browse through multiple web pages manually.

The implementation of this AI Agent is built using several advanced technologies. The system is developed primarily in Python, with libraries such as PyTorch used for deep learning computations. The Matplotlib and PIL (Pillow) libraries are used for image processing and visualization, while NumPy supports numerical computations. The integration of NLP and computer vision models ensures that the AI Agent can handle both text- based and image-based queries effectively. Furthermore, the use of cloud-based APIs like SerpAPI and Unsplash API ensures that the system retrieves real-time data dynamically, enhancing its

responsiveness and accuracy.
The AI Agent operates through a well-structured workflow. First, the user inputs a search query related to a specific plant. The system then initiates two parallel processes—one for retrieving and ranking images and another for fetching and ranking textual search results. For the image retrieval process, the AI queries Unsplash for relevant plant images, processes them using the CLIP model, and selects the most relevant image based on the computed similarity score. Meanwhile, for the textual search process, the system fetches Google search results, ranks them based on semantic similarity and visual relevance, and generates a structured explanation summarizing the key findings. Finally, the AI Agent displays the best-matching image, top-ranked websites, and the generated query explanation in a well-organized format.

4) Sentence Transformer: The all-MiniLM-L6-v2 model is a Sentence Transformer designed to generate high-quality sentence embeddings with efficiency. It is based on the MiniLM architecture and uses a 6-layer Transformer encoder. The model is designed for semantic similarity, clustering, and retrieval tasks, offering a good balance between performance and speed. It maps sentences into a 384-dimensional dense vector, making it lightweight and suitable for real-time applications. Trained on a diverse dataset, it achieves state-of-the-art performance across various natural language processing (NLP) tasks while remaining computationally efficient.
In this step, a Web Page Downloader retrieves relevant web pages from the internet and stores them as extracted pages for further processing. The Feature Extraction component then analyzes these pages, extracting key details such as titles, metadata, and main content. A deep learning-based classification algorithm determines the relevance of these pages, ensuring that only useful information is retained for the next phase.
The AI agent enhances search functionality using sentence-transformers to generate semantic embeddings, which are numerical representations of text. By converting both user queries and retrieved text into vector formats, the system computes cosine similarity to rank search results based on relevance, ensuring accurate and meaningful retrieval.The SentenceTransformer model, all- MiniLM-L6-v2, plays a crucial role by converting sentences into 384-dimensional vectors and using cosine similarity to measure text similarity. As a transformer-based model optimized for NLP tasks, it improves search accuracy by encoding queries into numerical representations and comparing them with retrieved results, making it an efficient tool for AI-driven search ranking and information retrieval.

5) Clip Model: The CLIP (Contrastive Language-Image Pretraining) model, developed by OpenAI, is designed to connect vision and language by learning associations between images and textual descriptions. It is trained on large datasets containing paired image-text data using a contrastive learning approach. This method enables the model to bring related image-text pairs closer together in a shared feature space while distancing unrelated pairs. As a result, CLIP can interpret images within the context of natural language. One of its most notable features is its zero-shot learning ability, allowing it to perform tasks such as image classification without requiring task-specific fine-tuning. It can categorize images based on textual prompts, even for classes it has not explicitly encountered during training. CLIP is highly adaptable, with applications including text-based image search, content moderation, visual question answering, and image generation when combined with models like DALL•E. Overall, CLIP facilitates a seamless understanding of visual and textual data, making It is a powerful resource for various multimodal applications.
At this stage, the AI agent retrieves and ranks high-quality images according to their relevance to the search query using CLIP-based ranking. The agent first sends a request to the Unsplash API with a plant- focused query, and the API returns a set of related images. The AI agent then uses the CLIP model to extract text features from the search query and visual features from the images. It calculates cosine similarity scores between the text embeddings and the image embeddings. The image with the highest similarity score is chosen and presented to the user. This process involves the use of the Unsplash API for fetching images, the CLIP model for image- text matching, and the Requests library for handling API requests.

Algorithm: Score = (0.7∗Text Similarity) + (0.3∗Image Similarity) Results are ranked based on relevance scores.

6) Cosine Similarity: Cosine similarity is a measure used to determine the similarity between two vectors by computing the cosine of the angle they form. It measures how closely the vectors align, where a score of 1 indicates identical vectors, 0 signifies no similarity, and -1 denotes completely opposite vectors. This measure is commonly used in machine learning models to compare feature vectors, such as text or image representations, which have been transformed into vectors.

The cosine similarity between two vectors, A and B, is calculated using the following formula:

$$Cosine\ Similarity = \frac{A \cdot B}{\|A\| \|B\|}$$

Where:-

- A·B represents the dot product of vectors A and B.
- ‖A‖ and ‖B‖ denote the magnitudes (or lengths) of vectors A and B, respectively.

In models like CLIP & Sentence Transformer, cosine similarity is used to compare the vector representations (embedding) of text and images. The higher the cosine similarity score, the more closely related the text and image are, which aids in tasks like retrieving relevant images based on a text query.

7) E-commerce Module: The e-commerce module is integrated into the system to facilitate the buying and selling of plant- based products. Users can register as buyers or sellers, list products for sale, browse the marketplace, and make purchases. The system includes features such as product listings, a shopping cart, checkout functionality, and payment gateway integration.

Sellers can upload images and descriptions of their products, such as medicinal plants, herbal extracts, seeds, and gardening tools. The system allows buyers to search for products based on plant names or categories. The search engine within the e- commerce module enables users to find relevant products efficiently. To ensure authenticity, the e-commerce module includes seller verification. Sellers are required to register and verify their identity before listing products.

IV. RESULT & DISCUSSION

The "Sanjeevani" website is an online platform dedicated to natural healing and ayurvedic remedies. It provides information about medicinal plants, herbal products, and wellness solutions rooted in ancient herbal traditions. The website includes sections for learning about herbs, shopping for natural products. Each section serves a distinct purpose, providing users with a seamless blend of education, exploration, and e-commerce focused on medicinal plants and wellness practices.

The herbology section is a fundamental component of the website, providing a carefully selected assortment of medicinal plants accompanied by images, scientific names, and succinct explanations of their health advantages. It encompasses a plant identification system, which enables users to upload plant images for instant classification and medicinal information, utilizing advanced artificial intelligence and deep learning models. By incorporating an AI search agent, this experience is further enhanced as it retrieves relevant content based on user queries, presents summarized information, visuals, and ranked sources to ensure credibility and ease of understanding. In addition to the informative sections, the sanjeevani natural store serves as a dedicated platform for purchasing a wide range of herbal products. It encompasses a diverse range of products, including herbs and supplements, essential oils, herbal teas, natural skincare items, and wellness kits. The store provides convenient filters and product cards that display images, prices, ratings, and discounts, guaranteeing a hassle-free and tailored shopping experience for customers. By combining these elements, sanjeevani becomes a comprehensive destination for individuals seeking natural remedies and holistic wellness solutions.

Fig. 2 Home page

Figure 2 illustrates the homepage of the Sanjeevani website. The homepage presents the site's branding and theme, emphasizing the "Ancient Wisdom of Natural Healing." The background features Sanskrit inscriptions, reinforcing the traditional and Ayurvedic essence of the platform. A prominent "Explore Plants" button invites users to discover medicinal plants, while the navigation bar provides access to different sections such as Home, Herbology, Shop, Community, and Contact.

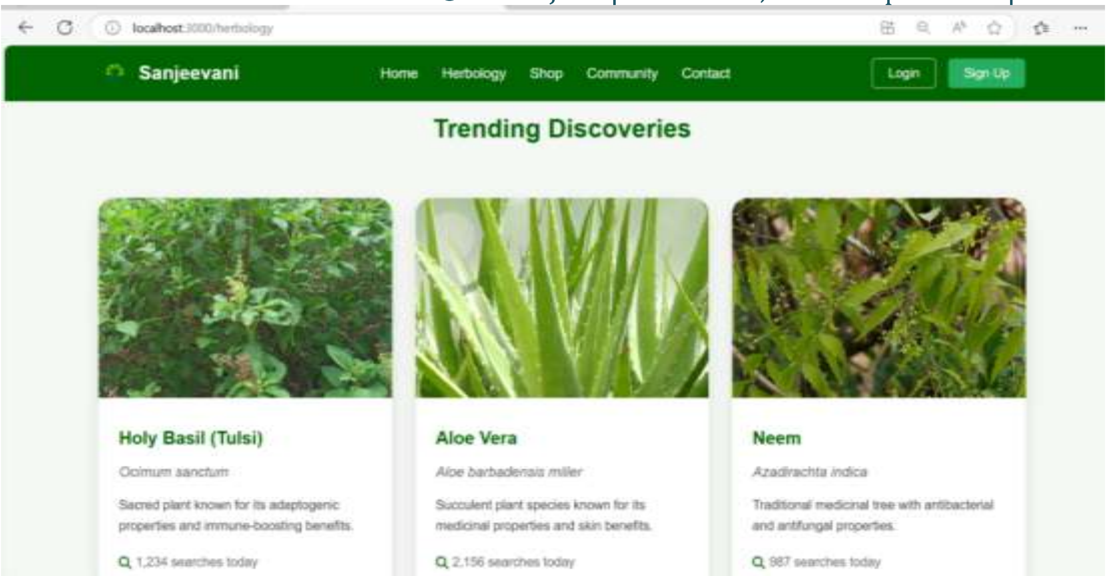

Fig. 3 Herbology page

Figure 3 illustrates the Herbology section. This section showcases trending medicinal plants with their images, scientific names, and brief descriptions of their benefits. For example, plants like Holy Basil (Tulsi), Aloe Vera, and Neem are highlighted, along with the number of searches they have received. This section aims to educate users about commonly used medicinal plants and their healing properties. Additionally, the Herbology page features a plant identification tool and an AI search agent, allowing users to identify plants through image uploads and receives instant information about their medicinal uses.

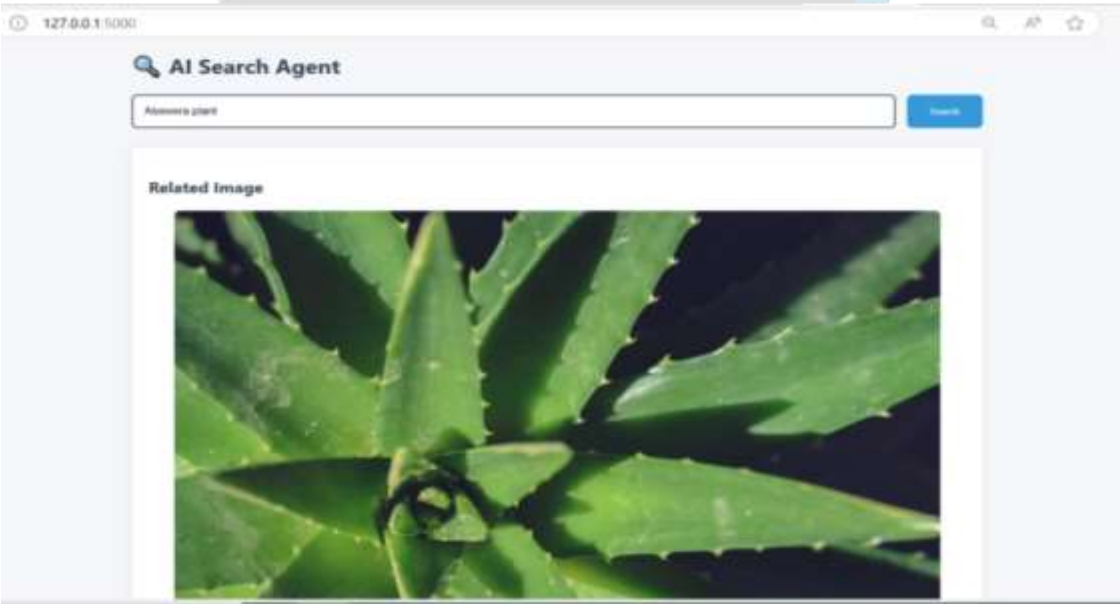

Fig. 4.1 AI Search Agent

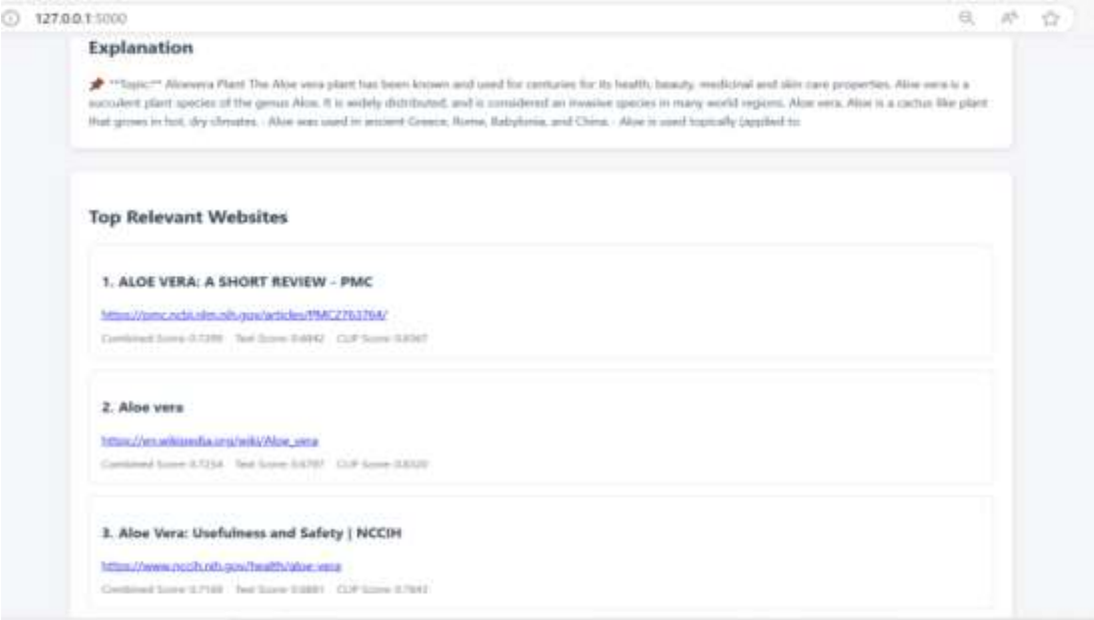

Fig. 4.2 AI Search Agent

Figure 4.1 and 4.2 illustrate that the AI-powered search agent efficiently retrieves and presents relevant information based on user queries. When a user searches for a term like "Aloe Vera Plant," the system responds with a related image, a concise textual summary, and a ranked list of the most relevant websites. The displayed image provides a visual representation of the searched topic, improving user engagement. The textual summary offers key details, including the plant's medicinal benefits, geographical distribution, and historical importance.

In addition, the system ranks websites based on their relevance, assigning scores derived from a combination of textual and visual relevance metrics. These scores are determined using text similarity measures and CLIP-based image-text correlation, allowing users to evaluate the credibility of the sources. By integrating AI-driven natural language processing (NLP) and computer vision techniques, the system enhances search accuracy and provides a more informative user experience

Fig. 5 Plant Identification Page

Figure 5 showcases the AI-driven plant identification system accurately classifies and predicts the given plant image as "Aloe Vera." Utilizing deep learning-based image recognition models, the system processes the input image to determine the plant species with precision. The result is displayed at the top, confirming the identified plant name, while the corresponding image is shown alongside the prediction to help users validate the classification's accuracy.

Additionally, the system provides an option to "Try another image," allowing users to upload different plant images for identification. The model's effectiveness in recognizing plant species based on visual characteristics highlights its potential applications in agriculture, medicine, and botany.

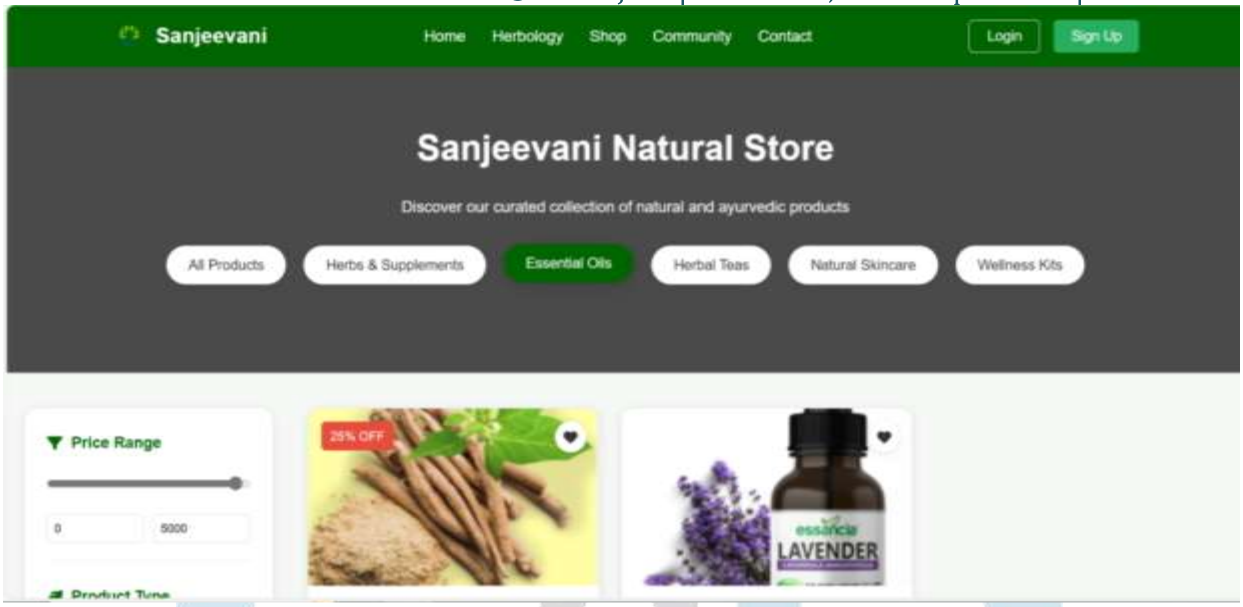

Fig.6.1 Ecommerce Page

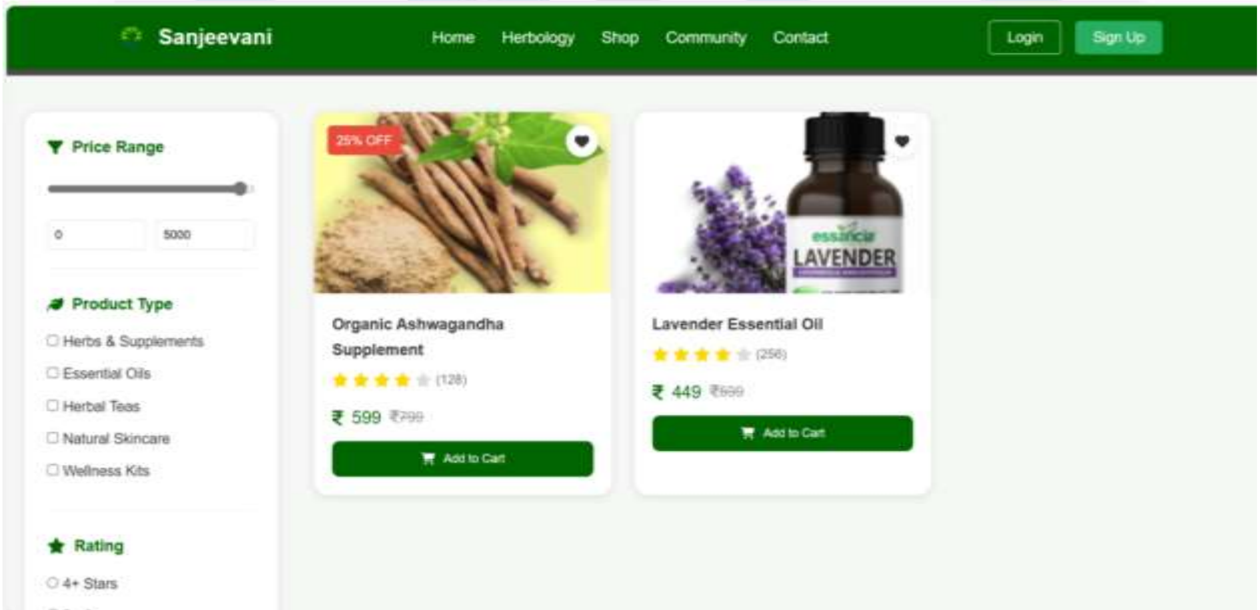

Fig.6.2 Ecommerce Page

Figures 6.1 and 6.2 illustrate the shopping section of the Sanjeevani Natural Store, offering a seamless browsing and purchasing experience for natural products. Users can explore a well-organized product catalog featuring categories such as Herbs & Supplements, Essential Oils, Herbal Teas, Natural Skincare, and Wellness Kits. The store provides filters on the left sidebar, including price range, product type, and rating options, allowing users to refine their searches efficiently. Each product card displays an image, price, discount, rating, and an "Add to Cart" button for easy purchasing. The interface is designed for user-friendly navigation, enabling visitors to discover and buy herbal products effortlessly based on their preferences.

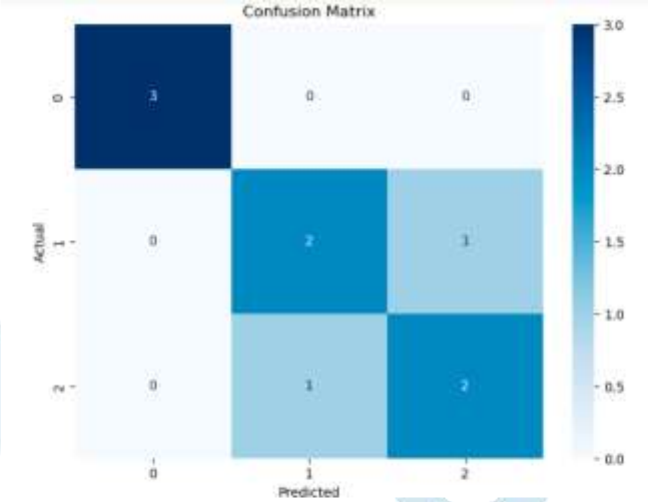

Fig.7.1 Plant Identification model Confusion matrix

```
Accuracy: 0.78
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         3
           1       0.67      0.67      0.67         3
           2       0.67      0.67      0.67         3

    accuracy                           0.78         9
   macro avg       0.78      0.78      0.78         9
weighted avg       0.78      0.78      0.78         9
```

Fig.7.2 Plant Identification model Classification report

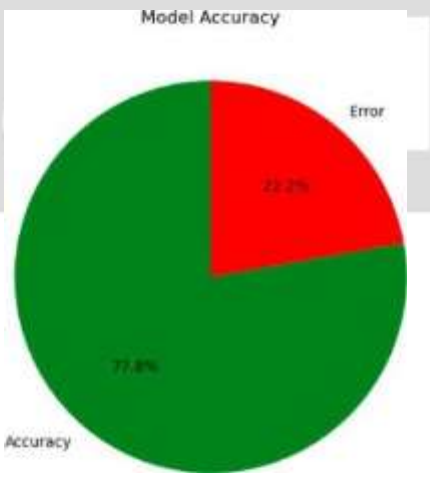

Fig.7.3 Plant Identification model Accuracy

Figure 7.1, 7.2 and 7.3 illustrates the results for plant identification system. The plant identification system, powered by AI and deep learning image recognition, demonstrates its capabilities through the provided visual data. The confusion matrix offers a snapshot of the model's performance, highlights its capability to correctly and incorrectly categorize plant species. A more granular view is presented in the classification report, which details the precision, recall, and F1- score for each class, culminating in a 78% overall accuracy. This accuracy is further depicted in the pie chart, where the model's successful predictions account for 77.8%, leaving 22.2% as errors.These metrics collectively illustrate the system's ability to classify plant species, reinforcing its applicability in fields such as agriculture, medicine, and botany as intended by its design.

V. CONCLUSION

In this research, we addressed the critical need for an efficient and reliable system for identifying medicinal plants and accessing related information. The "Sanjeevani" system leverages deep learning, AI-driven web crawling, and e-commerce integration to provide a comprehensive platform that enhances the accessibility and understanding of Ayurvedic medicine.

Our deep learning-based plant identification module, powered by CNNs, achieved a classification accuracy of 77.8%, demonstrating the effectiveness of our approach in accurately identifying plant species from images. This high accuracy supports the system's potential for real-world applications in fields such as botany, healthcare, and agriculture.

The AI-driven information retrieval module effectively retrieves and ranks relevant plant information from diverse online sources. By integrating NLP and computer vision techniques, the system provides users with accurate textual summaries, related images, and ranked website links, enhancing the overall search experience.

Furthermore, the integration of an e-commerce platform within the system facilitates the buying and selling of medicinal plant products, promoting accessibility and supporting sustainable practices within the herbal medicine community.

The Sanjeevani system addresses the limitations of traditional search engines and web crawlers by employing advanced techniques such as SBERT- based text similarity ranking and CLIP-powered image-text relevance scoring. This innovative approach ensures more precise, context-aware retrieval of information and enhances trust in e- commerce transactions.

In conclusion, The Sanjeevani system showcases a remarkable progression in the utilization of AI and deep learning in the field of Ayurvedic medicine. It offers a user-friendly platform that not only identifies medicinal plants and provides comprehensive information but also fosters a community-driven approach to knowledge sharing and commerce. This research lays the groundwork for future developments in this area, with the potential to transform how people access and utilize traditional medicinal knowledge.

REFERENCES

[1] N. Pawar, K. Rajeswari and A. Joshi, "Implementation of an efficient web crawler to search medicinal plants and relevant diseases," 2016 International Conference on Computing Communication Control and automation (ICCUBEA), Pune, India, 2016, pp. 1-4, doi: 10.1109/ICCUBEA.2016.7860006.

[2] ALkaraleh, Saed, and Hatice Meltem Nergiz Şirin. "A topic-specific web crawler using deep convolutional networks." The International Arab Journal of Information Technology, Vol. 20, No. 3, May 2023.

[3] Kontogiannis, Andreas, et al. "Tree-based focused web crawling with reinforcement learning." arXiv preprint arXiv:2112.07620 (2022).

[4] Gautam Pant and Padmini Srinivasan. 2005. Learning to crawl: Comparing classification schemes. ACM Trans. Inf. Syst. 23, 4(October 2005),430–462. https://doi.org/10.1145/1095872.1095875

[5] Elaraby, M. & Mohamed, Sherihan & Moftah, Hossam & Rashad, Magdi. (2019). A new architecture for improving focused crawling using deep neural network. Journal of Intelligent & Fuzzy Systems. 37. 1-13. 10.3233/JIFS-182683.

[6] Dhanith, P. & B, Surendiran & Raja, S.. (2020). A Word Embedding Based Approach for Focused Web Crawling Using the Recurrent Neural Network. International Journal of Interactive Multimedia and Artificial Intelligence. In Press. 1.10.9781/ijimai.2020.09.003.

[7] S. Patel, D. Agarwal, and A. Verma, "A Comparative Study of Focused Web Crawling Techniques Using Deep Learning," 2018 IEEE International Conference on Data Science and Advanced Analytics (DSAA), vol. 6, pp. 563-570, 2018.

[8] B. Yu, H. He, and J. Xu, "An Adaptive Focused Crawler Based on Topic-Related Graph Model,"2019 IEEE International Conference on Knowledge Engineering and Applications (ICKEA), vol. 2, pp. 287-292, 2019.

[9] L. Sun and J. Han, "A Novel Approach to Focused Web Crawling using Supervised Learning," 2017 IEEE International Conference on Artificial Intelligence and Web Mining (AIWM), vol. 1, pp. 178-183, 2017.