IJIRSS
International Journal of Innovative Research & Scientific Studies

ISSN: 2617-6548

URL: www.ijirss.com

A hybrid explainability framework for recommender systems using SHAP with natural language interpretations

Noushad Rahim M[1*], Mohamed Basheer K.P[2]

[1]Dept. of Computer Science, Sullamussalam Science College, Areekode, University of Calicut, Kerala, India.
[2]Dept. of Computer Science, Amal College of Advanced Studies, Nilambur, University of Calicut, Kerala, India.

Corresponding author: Noushad Rahim M (*Email: mannayilnoushadrahim@gmail.com*)

Abstract

Recommender systems based on deep neural networks are widely recognized for their high predictive accuracy, but their opaque nature limits transparency and interpretability, crucial qualities needed to ensure accountability, fairness, and trust, especially as the right to explanation is increasingly regarded as a legal and ethical obligation in high-stakes decision-making. This study presents a hybrid explainability framework for recommender systems that combines SHAP (Shapley Additive Explanations) with natural language justifications to enhance the explainability of recommender system outputs. The framework integrates both local and global SHAP-based explanations. Local explanations analyze instance-level Shapley values to identify features that positively influence individual recommendations, while global Shapley values, computed at the item level, capture the dominant features characterizing the recommended item, which are then translated into item-level textual descriptions. These two explanation layers are then used to generate coherent, human-readable justifications for recommended items using large language models. The approach is implemented in a career recommendation engine based on aptitude profiles from the Occupational Information Network (O*NET). Expert evaluations on a five-point Likert scale yielded mean scores between 3.7 and 4.4 across various user profiles, indicating moderately high acceptance of the explanations generated by the framework. The findings suggest that the framework enhances transparency, fairness, accountability, and user trust in the recommendation process.

Keywords: Black-box models, Explainable AI, Hybrid explainability framework, Natural language explanations, Recommender systems, Responsible artificial intelligence, SHAP, Transparency.

DOI: 10.53894/ijirss.v8i6.9669
Funding: This study received no specific financial support.
History: Received: 10 July 2025 / **Revised:** 13 August 2025 / **Accepted:** 15 August 2025 / **Published:** 4 September 2025

Competing Interests: The authors declare that they have no competing interests.
Authors' Contributions: All authors contributed equally to the conception and design of the study. All authors have read and agreed to the published version of the manuscript.
Transparency: The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.
Publisher: Innovative Research Publishing

1. Introduction

The opacity and complexity of machine learning models based on Deep Neural Networks (DNNs) raise significant concerns regarding transparency, accountability, and trustworthiness. This is particularly important for AI models such as recommender systems when employed in critical domains. It is essential to provide clear and judicious explanations to support informed decision-making. Regulatory standards like the General Data Protection Regulation (GDPR) [1] also highlight the need for explainable and responsible use of artificial intelligence.

Recommendation systems are classified as content-based filtering, collaborative filtering, etc. However, new approaches, such as Neural Collaborative Filtering (NCF), have been developed [2] and DNN-driven content filtering methods [3] have evolved. Due to their advantages, such as predictive power and accuracy [4]. Deep Neural Networks (DNNs) are commonly used in these recommender systems. However, they present challenges due to their black-box nature, as shown in Figure 1 and complicate prediction interpretability [5, 6]. As illustrated in Figure 2 accuracy accuracy-interpretability trade-offs exist in such models [7]. Despite their high accuracy, DNN-based machine learning systems may also produce misclassifications [8]. Thus, transparency, explainability, and fairness are essential for building trustworthy AI systems [9-11].

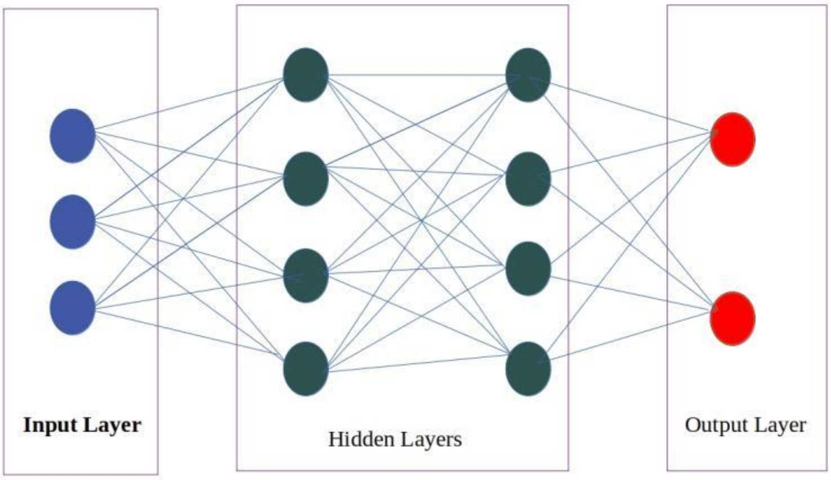

Figure 1.
Internal Architecture of DNN.

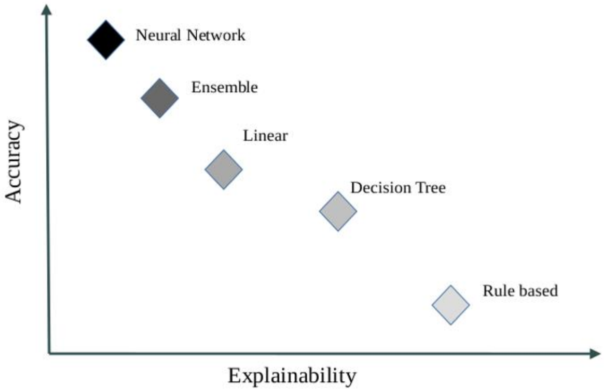

Figure 2.
ML models: accuracy versus interpretability.

To address the need for transparency in artificial intelligence systems, a range of explainable AI (XAI) techniques have been developed. These techniques are categorized along multiple dimensions: model-specific versus model-agnostic, global versus local interpretability, and intrinsic versus post-hoc approaches [12].

Model-specific explainability methods are tailored to the internal architecture of a given model. For instance, Layer-Wise Relevance Propagation (LRP) [13] is designed specifically for neural networks and offers insights by decomposing predictions layer by layer. In contrast, model-agnostic methods are applicable across different learning algorithms, regardless of their internal structure. Prominent examples include Local Interpretable Model-Agnostic Explanations (LIME) [5, 6] and SHapley Additive exPlanations (SHAP) [14], which provide post-hoc interpretations that are versatile and widely adopted.

Explainability can also be categorized as global or local. Global interpretability focuses on understanding the overall behavior of the model by ranking features according to their average influence on predictions, but local interpretability provides explanations for individual predictions by highlighting feature contributions for specific instances [15]. SHAP is particularly attractive as it supports both global and local interpretability.

Post-hoc explainability refers to explanatory techniques applied after model training. Post-hoc techniques employ visualization, feature attribution, and rule extraction strategies to interpret black-box models. In contrast, intrinsic explainability utilizes a model's internal structure, allowing direct interpretability of predictions. Models such as decision trees, linear regression, and rule-based systems exemplify this category and are often referred to as white-box models [16].

Post-hoc methods such as LIME and SHAP offer model-agnostic interpretability. LIME generates local surrogate models by perturbing input data [15] and SHAP utilizes cooperative game theory to assign additive feature importance values [14]. Grad-CAM [17] is another method, predominantly used in CNN-based architectures, which highlights class-discriminative regions of input images using gradient-based localization. While these tools enhance the transparency of complex models, they do so retrospectively. In contrast, ante-hoc models are designed with interpretability as a primary objective. Recent works [18-20] highlight the growing emphasis on combining these strategies to balance accuracy with interpretability in real-world decision-support systems.

1.1. Research Gap

Even though existing explainability methods such as SHAP, LIME, and Grad-CAM, can provide explanations for their predictions, these explanations are typically interpretable only by experts in the specific domain and are not easily understandable to general users. While LIME and SHAP provide feature attributions and visual representations of their interpretations [14]. Grad-CAM, primarily used in CNN-based computer vision tasks, highlights the regions of an image that contribute to a classification [17]. None of these methods can generate explanations in natural language; however, textual explanations can be indirectly generated using rule-based or template-based approaches [21].

1.2. Research Questions (RQ)

RQ1: How can a data-driven explanation be generated for ML-based models' predictions that can be provided in natural language that a common user can understand without much technical knowledge?

RQ2: How can SHAP XAI's local and global explainability features be used to develop an explainability framework to generate natural language justifications for the recommendations made by recommender systems?

1.3. Contributions of the Proposed Work

This study aims to develop a hybrid explainability framework for recommender systems by integrating SHAP (Shapley Additive Explanations) with textual justifications. Leveraging the local and global interpretability features of SHAP, the framework not only analyzes feature contributions at the level of individual recommendations and the feature importance of recommended items but also utilizes them complementarily to transform the resulting SHAP-based explanations into structured, human-readable narratives using the capabilities of large language models. By generating clear natural language explanations tailored for non-expert users, the study aims to bridge the gap between complex model outputs and user understanding, fostering trust, transparency, and informed decision-making in recommendation contexts.

2. Related Works

The integration of explainability into machine learning-based recommendation systems has become a critical research focus across multiple domains. Recent advancements to enhance the transparency of these opaque models include post-hoc interpretability techniques (e.g., Local Interpretable Model-agnostic Explanations (LIME), SHapley Additive Explanations (SHAP), Knowledge Graph-based reasoning), and model-intrinsic methods (e.g., Explainable Boosting Machines). This section critically reviews existing explainability approaches in recommender system research to identify methodological gaps and limitations and to support the development of a novel hybrid explainability framework aimed at improving transparency, user trust, and interpretability in recommendation tasks.

Gedikli et al. [22] investigated various aspects of explanations in recommender systems, such as efficiency, effectiveness, transparency, persuasiveness, and satisfaction. The study, conducted in the movie domain, highlights that transparency, user satisfaction, and trust are critical for user acceptance of recommendations. The findings emphasize the importance of providing both personalized and item-level explanations to enhance transparency and build long-term trust in these black-box models. These insights offer valuable guidance for designing explanation frameworks in any domain, fostering the development of systems that are transparent, trustworthy, and sustainable for long-term user engagement.

In their article, Afchar et al. [23] analyze how different explainability approaches, specifically feature-based, example-based, and graph-based explanations, can be integrated into various aspects of recommendation systems to enhance transparency. The study emphasizes the role of explanations in fostering user trust, supporting exploratory behaviors, and improving the interpretability of complex model outputs, thus bridging the gap between technical reasoning and human understanding. The article does not propose a framework for explainability but provides insights into various aspects of explainability in terms of different approaches and their usability. It also discusses how various algorithms, such as LIME, SHAP, and Knowledge Graphs, can be leveraged to address these aspects.

Yang et al. [24] introduced a Contextualised Knowledge Graph Embedding (CKGE) model for an explainable training course recommendation system. The framework constructed meta-graphs for each talent-course pair using entities such as skills, talents, and courses, and encoded semantic relationships using a Transformer architecture adapted for knowledge graphs. The system offered explainability by exploring meta-paths within the graph structure, providing semantic reasoning between user profiles and recommended courses. However, while effective in tracing logic chains, the framework did not natively support natural language generation of explanations.

Tran [25] proposed an empirical framework for job recommendation that combines both local and global explainability. LIME and Kernel SHAP were used to generate local feature attributions, while Explainable Boosting Machines (EBM) and Differentially Private EBM (DPEBM) provided global transparency through inherently interpretable model structures. The system leveraged textual data from job descriptions using TF-IDF and Latent Dirichlet Allocation (LDA), and incorporated XGBoost and Factorization Machines for ranking. Although model fidelity metrics were introduced to evaluate explanation quality, the system primarily mapped feature importances to raw text or topics, without developing full natural language narratives for broader user accessibility.

Shams et al. [26] proposed an explainable crop recommendation algorithm, LIME-based XAI-CROP, which considers a range of contextual parameters, including soil properties, weather conditions, and historical crop yield data. The system applies the LIME algorithm by perturbing samples from the validation dataset, training local surrogate models, and generating feature importance explanations for individual predictions. While this method offers instance-level interpretability, it remains limited to technical users due to its reliance on numeric attributions without contextualized textual narratives.

In the healthcare domain, Tang et al. [27] developed a prostate cancer treatment recommender system using SHAP to identify critical predictive features such as tumour stage, cancer stage, age, core positive percentage (CPP), prostate-specific antigen (PSA), and Gleason score. SHAP beeswarm plots were employed to visualize feature contributions across three different models. Their analysis demonstrated that SHAP explanations aligned well with domain expertise, guiding model selection. However, while the visual insights were clinically valid, the explanations lacked accessibility for non-specialists.

Collectively, existing literature demonstrates substantial progress in integrating explainability into recommender systems across various domains. However, a significant gap remains in producing human-readable, natural language explanations that non-technical users can easily comprehend. Most current methods, whether based on SHAP, LIME, or knowledge graphs, primarily offer visual or structural justifications that often require technical interpretation [28].

The proposed work addresses this gap by introducing a novel hybrid explainability framework that combines SHAP-based local and global interpretations with natural language justifications, thereby enhancing interpretability and fostering greater user trust, particularly among non-expert audiences. We selected SHAP over other methods due to its theoretical grounding in cooperative game theory, Lundberg and Lee [14], which guarantees faithful and additive feature attributions. In contrast, LIME relies on fitting local surrogate linear models, which may fail to accurately capture the underlying decision boundaries of complex systems. Moreover, SHAP inherently supports both local (instance-specific) and global (population-level) explanations, enabling complementary insights into personalized individual recommendations and overall model behavior [29]. These properties make SHAP particularly suitable for developing a comprehensive textual explanation framework.

3. Proposed Methodology

This section presents the theoretical foundation and technical architecture of the proposed hybrid explainability framework. Building on insights from the literature review, the framework is designed to enhance transparency and foster trust Akhtar et al. [30] in the recommendations generated by the system. Operating at both the individual instance and the recommended class levels, the framework adopts a feature-based explanation strategy. Specifically, local explanations are generated using SHAP to attribute feature contributions at the user profile level, whereas global explanations are derived by employing class-level aggregation of SHAP values for the recommended items. The framework aims to generate natural language explanations grounded in data-driven insights by systematically interpreting both local and global SHAP outputs. Through systematic feature importance analysis, numerical SHAP values are translated into accessible, human-understandable justifications, making the recommendations both interpretable and transparent for end users.

3.1. Interpreting the Model Recommendations Using SHAP

SHAP (SHapley Additive exPlanations) applies Shapley values from cooperative game theory Lundberg and Lee [14] to interpret machine learning model predictions, it distributes a model's prediction among input features, ensuring local accuracy and satisfying fairness axioms such as efficiency and symmetry. SHAP's strengths lie in its ability to operate in both model-agnostic and model-specific settings [31]. SHAP provides local explanations for individual predictions and global insights by aggregating feature contributions across datasets [32, 33]. Moreover, aggregating global SHAP values at the class level enables the generation of data-driven feature importance descriptions for specific classes, i.e., recommended clusters.

The SHAP explainer uses Shapley values to interpret predictions by calculating average feature contributions across all possible subsets, providing a unified measure of feature importance. The Shapley value ϕ_i is calculated using Equation 1.

$$\phi_i \ = \ \sum_{S \subseteq N\setminus\{i\}} \frac{|S|!\,(|N|-|S|-1)!}{|N|!} \ * \ (f_{S\cup\{i\}}(x_{S\cup\{i\}}) \ - \ f_{S(x_S)})) \qquad (1)$$

where:

ϕ_i is the Shapley value for feature *i*,

N is the set of all features,

S is any subset of *N* that does not include feature *i*,

$f_{SU(i)}$ is trained with that feature *i* present, and f_S is trained with the feature withheld.

Figure 3 visualises how the Shapley values of each feature of input x contribute to changes in the expected model prediction *E[f(z)]* to reach the predicted output *f(x)* [14] . *E[f(z)]*, called base value, represents the average prediction when no feature information is provided to the model.

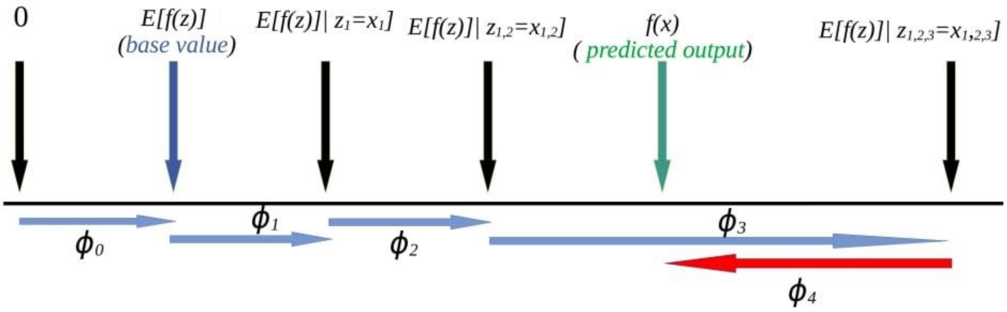

Figure 3.
Illustrates feature shifts to the final output using Shapley values.

The appeal of SHAP lies in its ability to provide both local and global interpretability for machine learning models. Local interpretability offers explanations for individual predictions by attributing the model output to the marginal contributions of input features, based on their Shapley values. In the context of recommender systems, it explains why a specific item is recommended to a particular user. The prediction is decomposed as the sum of the marginal contributions of all features, as described in Equation 2.

$$f(x) = \phi_0 + \sum_{i=1}^{M} \phi_i \qquad (2)$$

where:

ϕ_0 is the expected value of the model when no feature value is supplied,
ϕ_i is the Shapley value of the i^{th} feature, M is the set of all features.

Global explanations are obtained by aggregating Shapley values across the dataset, Saleem et al. [34] providing feature importance scores that characterize the model's overall behavior and identify the most influential features. They offer insights into how different features drive the model's predictions across the entire dataset. In multi-class problems, global explanations can also be used to identify significant features at the class level [35], enabling the description of each class's behaviour in terms of its most relevant features. The global importance of feature i across the dataset with N instances is computed as illustrated in Equation 3.

$$\text{Global Importance of Feature, } i \;=\; \frac{1}{N}\sum_{j=1}^{N} |\phi_i^{(j)}| \qquad (3)$$

where:
$\phi_i^{(j)}$ is the Shapley value of feature i for instance j.

3.2. The Architecture of the Proposed Explainability Framework

The proposed system introduces a novel mechanism that utilises SHAP's local and global interpretability features (i.e., numerical Shapley values) to generate comprehensive, human-readable textual explanations for the recommendations. Local explanations are generated by leveraging Shapley values to attribute feature contributions for individual user instances, providing personalised and feature-driven justifications for each recommendation. In parallel, global explanations are derived by aggregating Shapley values across instances at the class level and are used to generate feature-driven descriptions of the recommended class. A key novelty of the approach lies in synthesising local and class-level global insights, enabling the construction of unified textual explanations that combine personalised reasoning with broader class-level behavioural characteristics. The proposed architecture enables a holistic explanation by combining user-specific justifications (why the system recommends a particular class for the user) with class-level behavioural insights (why the class is generally recommended based on its data-driven profile), enhancing both interpretability and trust. The architecture of the system is illustrated in Figure 4.

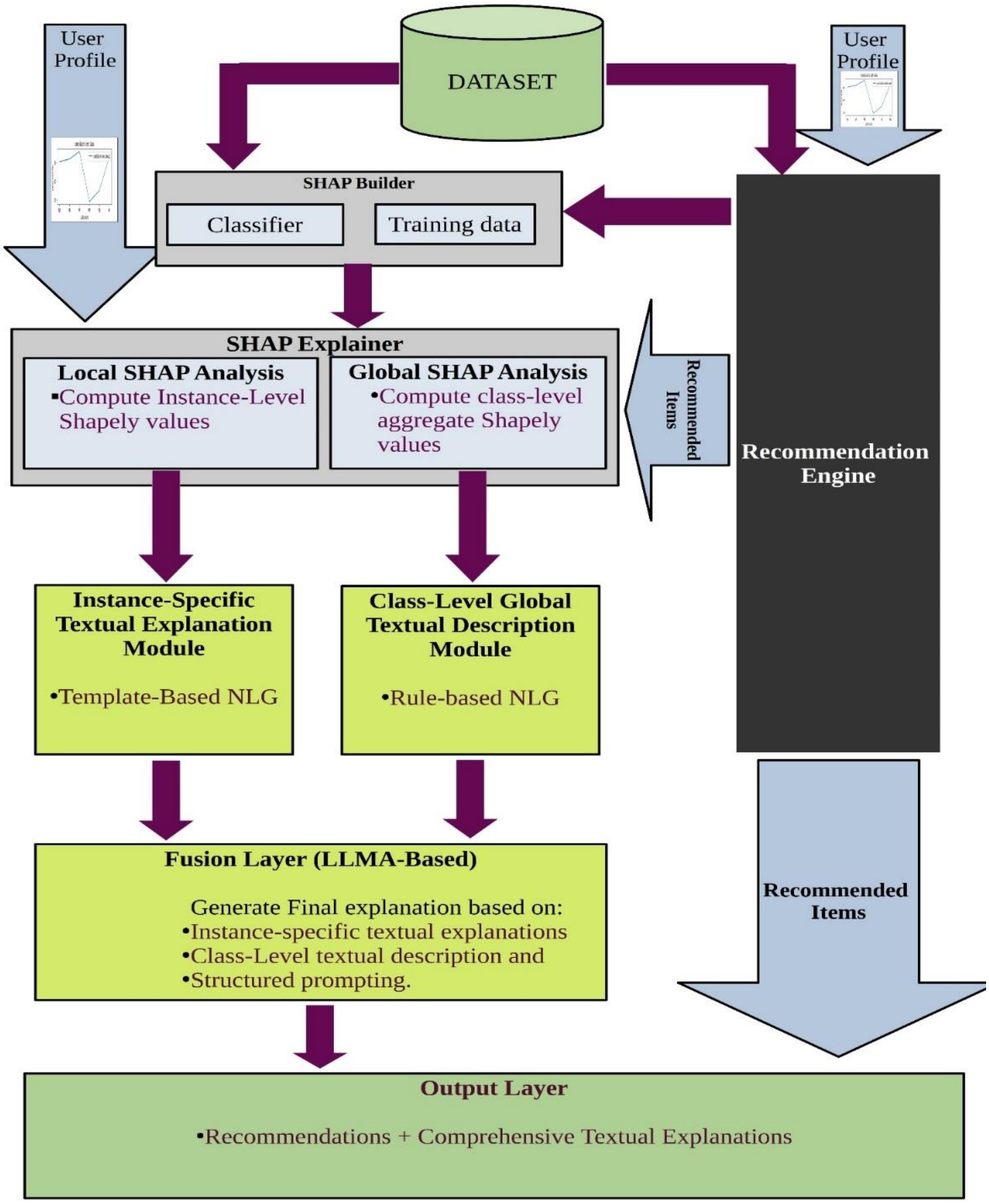
User Profile
DATASET
User Profile
SHAP Builder
Classifier
Training data
SHAP Explainer
Local SHAP Analysis
▪Compute Instance-Level Shapely values
Global SHAP Analysis
•Compute class-level aggregate Shapely values
Recommended Items
Recommendation Engine
Instance-Specific Textual Explanation Module
•Template-Based NLG
Class-Level Global Textual Description Module
•Rule-based NLG
Fusion Layer (LLMA-Based)
Generate Final explanation based on:
•Instance-specific textual explanations
•Class-Level textual description and
•Structured prompting.
Recommended Items
Output Layer
•Recommendations + Comprehensive Textual Explanations

Figure 4.
The hybrid explainability architecture of the proposed recommendation system.

As illustrated in the architecture Figure 4, the SHAP explainer is built on top of a recommendation engine and its underlying training dataset. A model-agnostic explainer, such as Kernel Explainer, is proposed to initialize the SHAP module. The choice of explainer can be determined based on the underlying classifier model used in the recommendation engine; for example, Deep Explainer is more suitable for neural network models implemented in TensorFlow or Keras, whereas Kernel Explainer can be used for any model when internal access is limited. The explainer applies SHAP's cooperative game theory framework to fairly allocate the contribution of each feature to the model's prediction. These contributions, known as Shapley values, are estimated using appropriate approximation methods, for instance, weighted linear regression in the case of Kernel Explainer [36].

3.3. Generating Local SHAP-Based Textual Explanation Using Template-Driven NLG

This module is the first layer of the textual explainability framework of the proposed system. The goal of local interpretability is to explain the rationale behind the model's recommendation for a given user instance. In the proposed framework, this is achieved through an instance-specific textual explanation module exploiting the SHAP explainer's local interpretability feature.

In a multi-class model, SHAP generates instance-specific explanations in terms of the marginal contribution of each feature for the predicted class. For interpretability, the explanation is produced for the class, $c = argmax_c(f_c(x))$. SHAP returns a vector of numerical values known as Shapley values, represented as $\phi^{(c)}=[\phi_1^{(c)}, \phi_2^{(c)}, \ldots, \phi_M^{(c)}]$ for M features. Typically, SHAP provides local interpretations by utilizing these numerical Shapley values in visual formats such as force plots, decision plots, or waterfall plots. A significant drawback of these plots is the requirement for technical expertise to interpret them.

In the proposed system, Shapley values are utilized to generate explanatory text that clarifies why a specific recommendation was made for a given user profile. The explanation module employs a template-driven natural language generation (NLG) approach [37], which interprets the instance-specific SHAP values corresponding to the predicted class. It calculates the proportional contribution of each feature by normalizing the absolute SHAP values, allowing for the identification and ranking of the most influential features. Based on the sign and magnitude of each feature's standardized value and its SHAP impact, the module fills predefined templates to construct clear, human-readable explanations that describe how individual features increased or decreased the likelihood of the recommended class. The complete procedure is detailed in Algorithm 1.

3.3.1. Generating Class-Level Textual Descriptions with Global SHAP Values and Rule-Based NLG

This is the second layer of the proposed textual explainability framework. The objective of this module is to provide a data-driven explanation for why a particular class is generally recommended, based on its features, using class-level global Shapley values. The important features of a model can be identified through global Shapley values [38], which is employed in the present work to determine feature importance at the item level (i.e., class level). The resulting output is a textual description of the class, derived from its most influential features. This class-level explanation, together with the instance-specific textual explanation discussed in the previous section, is then used to construct the final comprehensive explanation for the model's recommendations. Algorithm 2 is used to identify the key features that have a directly proportional influence on the likelihood of the class being recommended. Since Pearson correlation has been effectively used for feature selection in various contexts [39] the proposed framework adapts this approach to identify important features by computing the Pearson correlation coefficient between each feature's value and its corresponding Shapley value. Features with positive correlations are retained, as they indicate a consistent positive contribution to the class prediction. The selected features are then ranked based on the strength of their association.

Algorithm 1: Generate_Textual_Explanation_for_Local_SHAP_Values

```
Input:
        feature_names: List of M feature names
        instance_values: Feature values of the instance (length M)
        shap_values: SHAP values for a single instance (array of size M)
        predicted_class_idx: Index of the predicted class
Output:
        recommendation_explanation: A natural language explanation detailing proportional feature contributions
Steps:
  1.    Extract feature_impact ← shap_values corresponding to the predicted class.
  2.    Compute abs_shap_values ← absolute values of feature_impact feature-wise.
  3.    Compute total_impact ← sum of all abs_shap_values.
  4.    If total_impact ≠ 0:
          a. Calculate percentages ← (abs_shap_values / total_impact) × 100
        Else:
          a. Set percentages ← vector of zeros.
  5.    Sort features by descending order of percentages, storing their indices in sorted_indices.
  6.    Initialise an empty list explanation_list.
  7.    For each index i in sorted_indices:
          a. Retrieve feature ← feature_names[i]
          b. Retrieve value ← instance_values[i]
          c. Retrieve impact ← feature_impact[i]
          d. Retrieve percentage ← percentages[i]
          e. If impact > 0 and value > 0:
               Append:
"The high feature (value: value, impact: impact) contributed percentage%, increasing the likelihood of recommending this career cluster."
          f. Else if impact > 0 and value < 0:
               Append:
  "The low feature (value: value, impact: impact) contributed percentage%, increasing the likelihood of recommending this career cluster."
          g. Else:
               Append:
  "The feature (value: value) with low (impact: impact) contributed percentage%, decreasing the likelihood of recommending this career cluster."
  8.    Optionally sort explanation_list alphabetically for consistency.
  9.    Concatenate all entries in explanation_list to form recommendation_explanation.
  10.   Return recommendation_explanation.
```

Once the important features of the class are identified based on the global Shapley values, as described in Algorithm 2 this information is converted into a natural language description using a rule-based NLG technique, as outlined in Algorithm 3. Each feature's correlation value is categorized into qualitative significance levels using predefined thresholds (e.g., ≥ 0.5 for "very strong significance," ≥ 0.3 for "high importance," etc.). The threshold applied in this algorithm is an adapted version of Cohen's rule of thumb for interpreting Pearson correlation coefficients [40]. The resulting feature-driven description provides insights into why the class is generally recommended, based on its most significant features.

Instead of relying on a domain-specific class-level description, this data-driven methodology enables the automated generation of class-level textual descriptions without the need for domain expert intervention, making the approach adaptable to a wide range of recommendation systems. This feature importance-based description of the class, together with the instance-specific local textual explanation generated based on Algorithm 1 provides a robust foundation for generating a comprehensive final textual explanation of the recommendation made by the recommendation engine.

Algorithm 2: Identifying Class-Level Important Features Using Global Shapley Values

```
Input:
    Class_level_shap_values:   SHAP values for all samples belonging to a given class
    Feature_values: Feature values for the samples of the class
    Feature_names: List of feature names
Output:
    Top_features: Top features with the positive correlation between feature values and SHAP values
Steps:
    1.  Initialise an empty list: Correlations.
    2.  For each feature index i=1 to M:
        a. Extract the SHAP values and feature values for the feature.
        b. Compute the Pearson correlation coefficient between these two vectors.
        c. If the correlation is positive, append it to Correlations.
    3.  Create a map with each feature name and corresponding correlation value.
    4.  Sort the map in descending order of correlation.
    5.  Return the map containing the features and their significance.
```

Algorithm 3: Generating Class-Level Textual Description Based on Class-Level Global Shapley Values

```
Input:
        Feature_importance_map: Important features and their degree of importance
Output:
        Class_level_description: Textual summary describing significant features of the class
Steps:
  1.  Initialize Calss_level_description
  2.  For each record from Feature_importance_map
      a.  Feature = record[feature]
      b.  feature_impact =record[feature_importance]
      c.  If feature_impact>=0.5
                desc = "Feature {feature} has very strong significance as far as the class is concerned."
          Else if feature_impact >=0.3
          desc = "Feature {feature} has high importance as far as the class is concerned."
          Else if feature_impact >= 0.0
          desc = "Feature {feature} has moderately high significance as far as the class is concerned."
          Calss_level_description.append(desc)
   3.  Return Class_level_description.
```

3.3.2. LLM-Assisted Generation of Comprehensive Textual Explanations Using Local and Class-Level SHAP Insights

This is the final layer of the explainability framework. The layer, acting as a fusion layer, provides a comprehensive explanation for the recommendation made by the system. By employing a large language model (LLM) on the Shapley value-based instance-specific textual explanations and the class-level feature-driven textual descriptions generated using Algorithms 1, 2, and 3, this layer generates a meaningful final explanation for the recommendations made by the system.

The fusion layer integrates a systematic prompt to guide LLM reasoning based on insights from instance-specific textual explanations and class-level feature-driven descriptions derived through the use of local and global SHAP-based interpretations. This novel methodology, grounded in reasoning informed by SHAP-derived insights, enables the generation of meaningful explanations through a constrained LLM.

Proposed Prompt for the Explanation Generation via LLM:

Context*:*

Two types of feature-based textual explanations are provided:

Instance-specific justification: A textual explanation for the recommendation derived from the feature contribution *scores corresponding to an individual user profile.*

Class-Level Feature Summary: A description of characteristic feature patterns associated with the recommended class.

***Inputs**:*

Instance_Specific_Justification: (instance_specific_recommendation_explanation)
Class_Level_Feature_Summary: (class_level_description)

***Objective**:*

Generate a final explanation that evaluates how the individual-level justification aligns with or contradicts the class-level feature summary.

***Constraint**:*

Constrain the explanation strictly to the provided inputs. Do not use information beyond the given content.

3.3.3. Experimental Setup: Implementation of the Explainability Framework in a Career Recommender System

To demonstrate the applicability of the proposed hybrid explainability framework, this section illustrates its end-to-end implementation within a career recommender system. It covers the design of the career recommendation engine, the process of generating user profiles, local explanation generation using SHAP, construction of class-level descriptions based on global SHAP values, integration of a language model for natural language generation, and the final step of producing comprehensive explanations for the system's recommendations.

The dataset used to develop the career recommendation engine was constructed using ability profile data from the Occupational Information Network (ONET) database [41]. The original ONET profiles consist of 52 ability-related features spanning approximately 1,000 career roles. These features were mapped to six core aptitude dimensions: Numerical Ability (NA), Abstract Reasoning (AR), Verbal Reasoning (VR), Mechanical Reasoning (MR), Spatial Relations (SR), and Verbal Ability (VA) aligned with the domains assessed by the Differential Aptitude Tests (DAT). During preprocessing, irrelevant and duplicate career entries were removed, resulting in a curated dataset comprising 612 unique careers, each represented by a six-dimensional aptitude profile.

The career recommendation engine is implemented using a Deep Neural Network (DNN) designed for multi-class classification. The model architecture consists of three hidden layers with ReLU activation functions, followed by a SoftMax output layer that produces a probability distribution over the career classes. The model accepts a six-dimensional aptitude vector as input and generates a ranked list of the most suitable career options for a given user profile.

User profiles are generated using an aptitude assessment tool specifically the Kerala Differential Aptitude Tests (KDAT), developed by the Directorate of General Education, Government of Kerala [42]. Each candidate is evaluated across the same six aptitude dimensions used to represent the training data. The resulting aptitude profile is then fed into the trained DNN model, which outputs personalized career recommendations based on the candidate's aptitude strengths.

The SHAP explainer, implemented using Deep Explainer, is constructed with the career recommendation engine and its underlying training dataset to generate both visual and textual explanations. The Deep Explainer was used due to its model-agnostic nature and the model being built on a DNN. This facilitates transparency and enhances the trustworthiness of the system's recommendations.

Local interpretability is achieved after the candidate's aptitude profile, derived from KDAT, is fed into the recommendation engine and a corresponding career recommendation is generated. The user profile and the predicted class are then passed to the SHAP local explainer, which computes the Shapley values representing the marginal contribution of each aptitude dimension, such as Numerical Ability (NA), Abstract Reasoning (AR), or Spatial Relations (SR), to the predicted career cluster. These Shapley values are subsequently translated into human-readable explanations using the framework's template-based natural language generation (NLG) module, as illustrated in Algorithm 1.

Once the local explanation is generated in natural language, the class-level global textual description module the second layer of the explainability framework, analyzes the recommended career class to determine its class-specific feature importance. This is achieved by identifying features such as Numerical Ability (NA), Abstract Reasoning (AR), and others whose class-level global SHAP values are positively correlated with their corresponding feature values. These positive correlations indicate the extent to which higher feature values consistently contribute to the model's prediction of the given class, thus reflecting their importance within that class. The most influential features, ranked based on the strength of their correlation, are then translated into a textual description using a rule-based NLG approach. This process, illustrated in Algorithms 3 and 4, produces a human-readable summary of class-level characteristics, explaining that the career class is generally recommended due to the consistent contribution of these significant features.

The fusion layer is implemented using a large language model (LLM), specifically Google's Gemini model, to generate the final explanation in natural language for the career recommendation. The instance-specific textual explanation detailing why the system recommended a particular career class and the class-level textual description highlighting why that career class is generally recommended are fed into the fusion layer of the explainability framework. The prompt is customized for the career recommendation context while preserving the structural constraints defined in the framework, ensuring that the reasoning is grounded in the local and global textual explanations derived from Shapley values. This helps minimize hallucinations and enables the generation of a coherent and comprehensive final explanation.

3.3.4. Expert Evaluation Methodology

An expert evaluation was conducted as part of the validation process for the proposed recommendation engine and the hybrid explanation framework. This evaluation involved 16 distinct user profiles, covering approximately 33% of the career clusters present in the dataset. Ten expert career counsellors from the Department of General Education, Government of Kerala, were selected for the evaluation. Each expert was provided with 16 user profiles, containing scores from aptitude tests across six aptitude domains, the system-generated recommended career clusters, and the corresponding natural language explanations. The experts were instructed to assess both the relevance of the recommended career clusters and the quality of the accompanying explanations. They rated each on a five-point Likert scale [43].

1 – Not satisfactory
2 – Slightly satisfactory
3 – Moderately satisfactory
4 – Satisfactory
5 – Highly satisfactory

These ratings were collected for both the recommendations and their textual explanations to evaluate the system's overall effectiveness and interpretability.

3.3.5. Ethical Considerations

The expert evaluation was conducted between 17 April and 28 May 2025, following ethical approval granted on 10 April 2025. All participants were professional career counsellors who voluntarily participated in the expert evaluation of system-generated explanations. Before participation, they received detailed information about the study through the introduction section of the Google Form. The form included a mandatory consent question confirming their willingness to participate, which had to be acknowledged before proceeding. No personal, sensitive, or health-related data were collected, and the study involved no risk or intervention. This consent procedure was reviewed and approved by the Ethics Committee of Sullamussalam Science College, Areekode.

4. Results and Discussions

This section discusses the performance of the underlying career recommendation engine, validates the SHAP explainers using local accuracy metrics, analyzes SHAP visual plots for local and global interpretability, examines instance-level and recommended class-level explanations, and reviews the final comprehensive natural language explanations generated by the explainability framework for a given recommendation.

4.1. Evaluation of the Career Recommendation Engine

The career recommender system, with key evaluation metrics such as accuracy, precision, recall, and F1-score all exceeding 99%, demonstrated excellent performance, as shown in Table 1. This remarkable consistency underscores the model's robust and reliable predictive capabilities across various performance dimensions. Such strong performance affirms the suitability of the recommendation engine for evaluating the proposed hybrid explainability framework, as SHAP-based explainers rely on both the classifier and its corresponding training dataset for generating meaningful explanations.

Table 1.
Performance Metrics of the Career Recommendation Engine.

Evaluation Metric	Performance
Accuracy	0.99220
Precision	0.99223
Recall	0.99221
F1-Score	0.99220

To illustrate the system's effectiveness, consider the following user profile:

$$USER\_PROFILE_{(NA, AR, VR, MR, SR, VA)} = [65, 60, 66, 25, 24, 59]$$

For this profile, the recommendation engine predicted a career cluster, labelled 200, as shown in Figure 5. The careers within this cluster typically require strong numerical, abstract, and verbal reasoning skills, aptitudes reflected in the user's profile. The alignment between the candidate's aptitude and the characteristics of the predicted career cluster demonstrates the system's ability to generate relevant and contextually appropriate recommendations based on domain knowledge. This example serves as a representative case illustrating the model's consistency and interpretability.

4.2. Evaluating the SHAP Explainer

To evaluate the validity of the generated SHAP explainer, local accuracy metrics were used, as defined in Equation 4.

$$\text{Local Accuracy} = f(x) = g(x) = \phi_0 + \sum_{i=0}^{M} \phi_\mathrm{i} \qquad (4)$$

where,

f(x): The predicted value of the recommender,
g(x): The approximation of the model prediction via SHAP Explainer,

ϕ_0: The base-value of the model.,
ϕ_{i}: Shapley value of feature i,
M: Total number of features present in the model.

For the *USER_PROFILE*$_{(NA,\ AR,\ VR,\ MR,\ SR,\ VA)}$ *= [65, 60, 66, 25, 24, 59]*, and the corresponding recommended career class with label 200, as shown in Figure 5, we evaluated the explainer's validity by conducting a local accuracy analysis, as illustrated below.

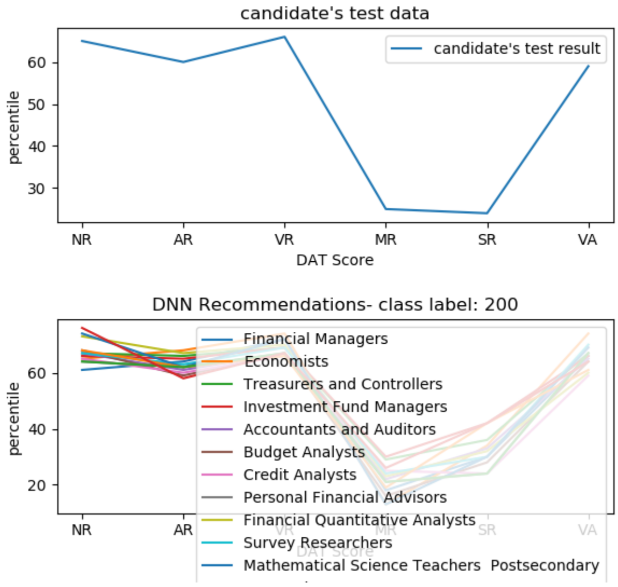

Figure 5.
Recommended careers for the candidate profile.

From the SHAP Explainer, we have,
Predicted value, f(x)=200
SHAP explainer's base value for the model,
ϕ_0 = 557.1110703812317
Shapley values of the features for the predicted class
= [-52.00565738, -45.4643695, -9.70420332, -68.92581867, -165.69422043, -15.31680108]

Applying Equation 4
g(x) = base value + Sum (Shapley values)
= 557.1110703812317+-357.11107
= 200
= f(x)

This result, f(x)=g(x)=200, confirms that the explainer accurately approximates the model's output, ensuring local faithfulness and transparency in interpreting decisions for this specific input. This validates the generated explainer and supports its use in effectively explaining the model's recommendations.

4.3. Local Explanation Evaluation

This section will analyze the explanation generated for the instance-specific recommendation made by the career recommender system. The explanation is generated using a SHAP force plot, followed by a textual explanation provided by the instance-specific textual explanation module of the proposed explainability framework.

The user profile used for validating local accuracy, along with the corresponding recommended career class labelled 200 (as illustrated in Figure 5), is considered here for evaluating the local explanation.

USER_PROFILE$_{(NA,\ AR,\ VR,\ MR,\ SR,\ VA)}$ *= [65, 60, 66, 25, 24, 59]*
After applying standardization, the transformed feature values are:
[1.69090851, 0.19389258, -0.05227239, -0.90342015, -1.79073424, -0.11755611]

The Shapley values of the features for the given user profile for the predicted class are:
[0.44310025, 0.07924637, 0.0609078, 0.09494073, 0.24430919, 0.05990119].

And, the expected SHAP value (base value) for the recommended class is: 0.01759431

For this user profile and the predicted career class, SHAP visualization tools specifically, a force plot are used to explain the model's decision. The resulting explanation is presented in Figure 6.

Figure 6 demonstrates how the force plot explains the model's prediction for the user profile. It specifically shows how higher values of NA and AR, along with lower values of MR and SR, significantly contribute to shifting the model's prediction from its base value (0.01759) to the predicted logit value (1.008). The explainer effectively highlights that features such as Numerical Ability (NA) and Abstract Reasoning (AR) are key aptitudes for the recommended career cluster, as depicted in Figure 5, where almost all careers require mathematical and problem-solving skills, while Mechanical Reasoning (MR) and Spatial Relations (SR) are less relevant. If MR and SR had higher positive values, the recommendation engine would have suggested a different career cluster for the candidate.

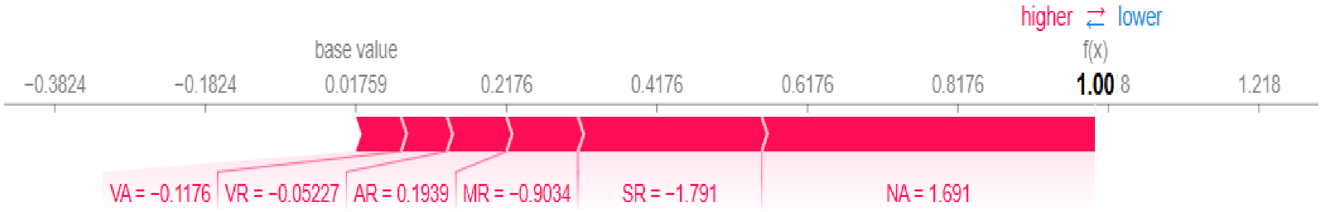

Figure 6.
Force plot showing marginal contribution of features to the model's prediction.

4.4. Shapley Value-Driven Instance-Level Textual Explanation Analysis

By applying Algorithm 1 to the standardized feature values:

1.6909, 0.1939, -0.0523, -0.9034, -1.7907, -0.1175

and the corresponding Shapley values:

[0.44310025, 0.07924637, 0.0609078, 0.09494073, 0.24430919, 0.05990119],

for the user profile:

USER_PROFILE $_{(NA,\ AR,\ VR,\ MR,\ SR,\ VA)}$ *= [65, 60, 66, 25, 24, 59]*

and the recommended career cluster depicted in Figure 5, the proposed instance-specific textual explanation module generated the following explanation:

- *The high Numerical Ability skill (value: 1.6909,*
- *impact: 0.4431) contributed 45.1%, increasing the likelihood of recommending this career cluster.*
- *The high Abstract Reasoning skill (value: 0.1939, impact: 0.0792) contributed 8.1%, increasing the likelihood of recommending this career cluster.*
- *The low Mechanical Reasoning skill (value: -0.9034, impact: 0.0949) contributed 9.7%, increasing the likelihood of recommending this career cluster.*
- *The low Spatial Relation skill (value: -1.7907, impact: 0.2443) contributed 24.9%, increasing the likelihood of recommending this career cluster.*
- *The low Verbal Ability skill (value: -0.1175, impact: 0.0599) contributed 6.1%, increasing the likelihood of recommending this career cluster.*
- *The low Verbal Reasoning skill (value: -0.0523, impact: 0.06091) contributed 6.2%, increasing the likelihood of recommending this career cluster.*

This textual explanation is consistent with the visual interpretation presented in Figure 6 and aligns with established domain knowledge in differential aptitude testing. It correctly informs the candidate that their high scores in Numerical Ability and Abstract Reasoning are the primary factors behind the recommendation of the suggested career cluster. Conversely, Mechanical Reasoning (MR) and Spatial Ability (SA) are less influential in this context; their lower values contribute positively to the recommendation. Had MR and SA been higher, a different career cluster might have been suggested. This explanation offers a more accessible and meaningful interpretation for typical users, such as students, who may lack the technical background required to interpret graphical outputs.

4.5. Shapley Value Driven Class Level Global Explanation

For the recommended career cluster presented in Figure 5 by the recommendation engine, for the user profile, *USER_PROFILE* $_{(NA,\ AR,\ VR,\ MR,\ SR,\ VA)}$ = *[65, 60, 66, 25, 24, 59]*, a feature importance analysis was carried out based on the Shapley values of the features of the cluster. SHAP Beeswarm is used to illustrate the result, as shown in Figure 7.

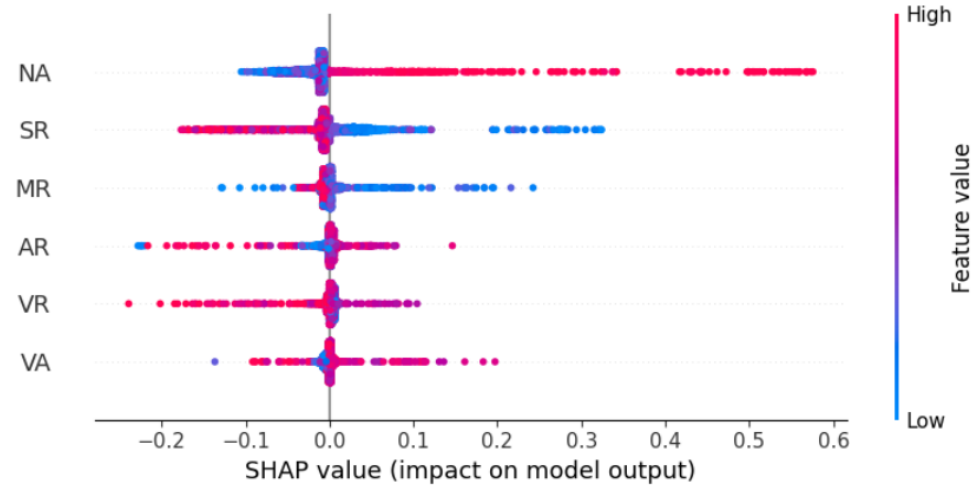

Figure 7.
Summary beeswarm plot depicting the feature importance for the career cluster, labelled 200.

From the beeswarm plot (Figure 7), it is evident that Numerical Ability (NA) is the most influential feature for the career cluster labelled 200 (illustrated in Figure 5). The plot further indicates that Abstract Reasoning (AR) and Verbal Ability (VA) are also significant features for this cluster. Meanwhile, Spatial Relations (SR) and Mechanical Reasoning (MR) appear to have minimal influence. This data-driven insight aligns with established domain knowledge in differential aptitude testing. Furthermore, the career analysis of the cluster illustrated in Figure 5 reveals that Numerical Ability and Abstract Reasoning are the most important aptitudes across all careers within this cluster. These Shapley value-based, data-driven insights are utilized to generate the class-level textual description for the predicted class in the following section.

4.6. Analysis of Class-Level Textual Descriptions Generated from Global Shapley Value Interpretations

As shown in Figure 7, interpreting this class-level explanation requires technical expertise, particularly in understanding beeswarm plots. Such visualizations may not be accessible to typical users, such as student candidates, who may lack the necessary background to comprehend the rationale behind the career recommendation. The class-level textual description module of the proposed framework addresses this limitation through Algorithms 2 and 3. Algorithm 2 identifies the feature importance of the career cluster based on the positive contribution of features and their corresponding Shapley values, while Algorithm 3 translates these insights into a natural language textual description of the career cluster.

For the recommended career cluster depicted in Figure 5, feature importance was analyzed using Algorithm 2, which leverages the class-level global Shapley values of the cluster. Table 2 presents the Pearson correlation between each feature and its corresponding Shapley value, considering only positive correlations to determine feature importance.

Table 2.
Correlation Between Feature Values and Corresponding SHAP Values.

Feature	Correlations (SHAP vs Feature values)
Numerical Ability	0.55
Verbal Ability	0.21
Abstract Reasoning	0.10

Finally, the module employing Algorithm 3 translates the feature importance information, illustrated in Table 2, into a natural language description of the cluster as shown below:

The feature Numerical Ability has very strong significance as far as this cluster is concerned.
The feature Verbal Ability has moderately high significance as far as this cluster is concerned.
The Feature Abstract Reasoning has a moderately high influence as far as this cluster is concerned.

This textual description of the career cluster is largely accurate concerning the careers represented in the cluster. All the identified important features are relevant to the careers within this cluster as per the established domain knowledge. However, in the case of Abstract Reasoning, its significance appears to be greater than what is indicated by the Shapley-based explanation. To enhance the accuracy of the description, this insight can be augmented with the domain knowledge from the respective field.

4.7. Final Explanation via Fusion Layer

The Fusion layer is designed to generate a comprehensive natural language explanation for the recommendations made by the recommendation engine, leveraging Large Language Model-Assisted (LLMA) reasoning. It operates over two distinct yet complementary textual explanations: one derived from the instance-specific explanation module and the other from the class-level description module based on Shapley values. The ultimate objective is to deliver a data-driven explanation that integrates both personalized and class-level insights in a human-readable format, using the reasoning capabilities of an LLM constrained by Shapley value-based inputs and a systematically crafted prompt.

For the user profile, *USER_PROFILE* $_{(NA,\ AR,\ VR,\ MR,\ SR,\ VA)}$ *= [65, 60, 66, 25, 24, 59]*, and the corresponding recommended career cluster presented in Figure 5 by the recommendation engine, based on the instance-specific textual explanation (presented in section 4.4) and class-level textual description (presented in section 4.6), the fusion layer of the system generated the following explanation for the career recommendation made by the recommendation engine, as shown below:

Final Explanation Generated by the Framework for the System's Recommendation:

"The recommendation is well supported by the alignment between the student's aptitude profile and the career cluster's requirements. The student demonstrates a significantly high score in Numerical Ability (NA), which is identified as a very strong determinant for this career cluster, making it the most influential match. Additionally, the student's Abstract Reasoning (AR) score moderately exceeds the mean, satisfying the cluster's requirement where AR holds only mild significance. Though Verbal Ability (VA) is slightly below average, it remains within a reasonable range for a feature with moderate relevance in this career cluster. The remaining attributes, Mechanical Reasoning (MR), Spatial Relations (SR), and Verbal Reasoning (VR) are comparatively low, but their reduced values positively contribute to the recommendation. This suggests that these dimensions are either less critical or inversely associated with the target cluster. Therefore, the recommendation is justified by a strong correspondence between the most influential student features and the defining characteristics of the recommended career cluster".

The final explanation generated by the fusion layer is largely consistent with established domain knowledge. While Verbal Reasoning (VR) appears to be a strong trait in the recommended career cluster, the Shapley value-based assessment suggests that VR holds limited relevance for the recommended career cluster. This observation stands in contrast to domain expectations, where VR is typically considered more important. This limitation arises from the system's exclusive reliance on Shapley value-based, data-driven explanation generation, without incorporating domain expertise specific to the field. Nevertheless, the remaining aspects of the explanation align well with both the data-driven insights and expert understanding, supporting the overall validity of the recommendation. Finally, the Fusion Layer successfully addressed the issue of hallucinations limiting reasoning based on given input by utilizing the prompt design.

4.8. Comparative Analysis with Existing Natural Language Explanation Frameworks

The proposed interpretability framework generates natural language explanations using data-driven approaches, providing a clear rationale for how and why the system makes recommendations. This offers significant advantages over existing explainability frameworks, as shown in Table 3, which presents a comparison between the proposed framework and existing explanation methodologies.

Table 3.
Comparison of the proposed explainability framework with existing approaches.

Study	Explanation method	Output format	Limitations	The proposed work
Yang et al. [24]	Contextualized Knowledge Graph + Transformer over meta-paths	Graph reasoning (meta-paths)	Context-aware but not user-friendly explanation	Context-aware and user-friendly explanations
Tran [25]	LIME, SHAP (local), EBM/DPEBM (global), TF-IDF, LDA	Feature Importance + Topic keywords	Designed for technical interpretability rather than end-user accessibility	Provides a user-friendly explanation with feature importance analysis.
Shams et al. [26]	LIME-based XAI with contextual features	Numeric feature importances	Limited to expert users; lacks a textual narrative	Provides natural language explanations.
Tang et al. [27]	SHAP for feature attribution	Visual explanations (SHAP plots)	Not accessible to non-specialists	Uses SHAP's local and global interpretability features to provide textual explanations.

4.9. Expert Evaluation of the System's Recommendations and Corresponding Explanations

The expert evaluation conducted on 16 user profiles, covering approximately 33% of the career clusters in the dataset and involving ten career counsellors, confirmed the acceptability of the explanations generated by the framework in support of the recommendations produced by the recommender engine. The mean rating values on a five-point Likert scale ranged from 3.4 to 4.6 for the recommendations and from 3.7 to 4.4 for the corresponding explanations across the 16 profiles, indicating a generally positive response. An average standard deviation of 0.80 for both recommendation and explanation ratings suggests moderate variability in expert opinions across different user profiles. Figure 8 illustrates the distribution of mean expert satisfaction scores on a five-point Likert scale for the recommendations and corresponding explanations across the 16 user profiles.

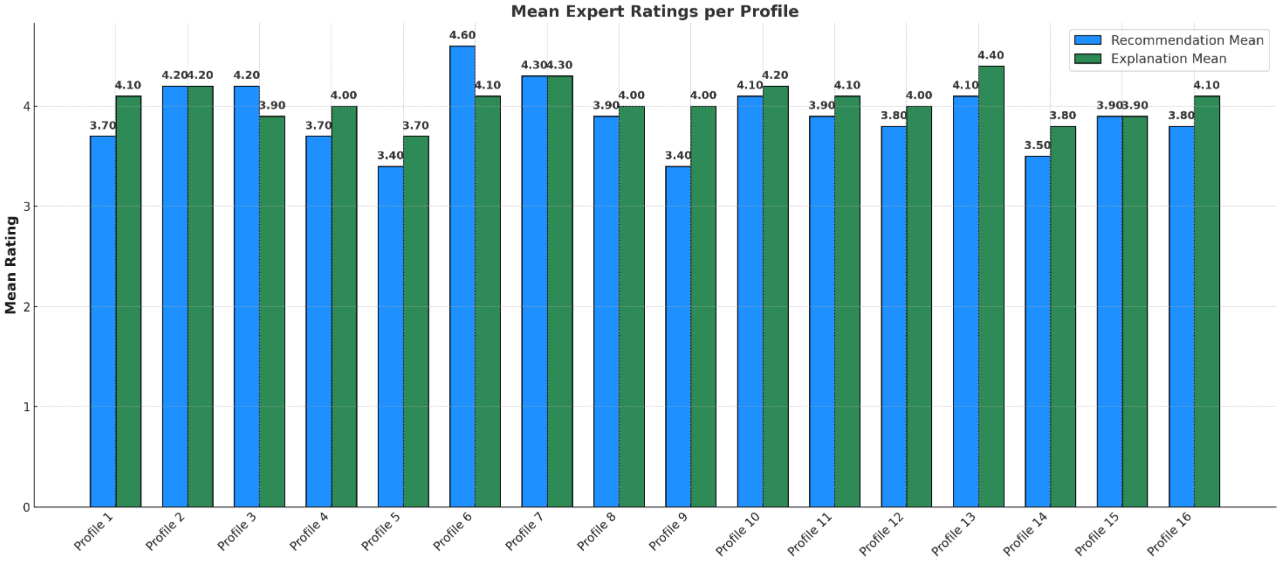

Figure 8.
Mean expert ratings per profile on a scale of 1 to 5 (1: Not Satisfactory, 5: Highly Satisfactory)

5. Limitations of the Study

While the proposed system demonstrates the ability to generate automated textual explanations for recommendations made by the recommender engine, the data-driven approach based on Shapley values may not always align with established domain knowledge in specific fields. This limitation is illustrated in Table 4, which is based on the *USER_PROFILE* $_{(NA,\ AR,\ VR,\ MR,\ SR,\ VA)}$ *= [65, 60, 66, 25, 24, 59]*, and the corresponding recommended career cluster shown in Figure 5. This limitation should be addressed by incorporating domain-specific insights, at least within the fusion layer of the framework. Another constraint is the framework's dependency on carefully crafted prompts to generate factual explanations from the underlying SHAP-based outputs; any discrepancy in prompt design may lead to hallucinations or misinterpretations in the generated reasoning.

Table 4.
Alignment of SHAP-Derived Feature Importance with Expert Domain Knowledge.

Feature	Student Score (in percentile)	Shapley Impact (%)	Cluster-Level Feature Importance (as per Shapley values	Domain Expectations
NA	65	45.1	Very Strong	Very Strong
AR	60	8.1	Moderately High	Moderately High
VR	66	6.2	Low	Strong
MR	25	9.7	Low	Low
SR	24	24.9	Low	Low
VA	59	6.1	Moderately High	Moderately High

6. Conclusion

This study introduces a hybrid explainability framework for recommender systems that integrates local and global SHAP-based feature attributions with constrained large language model (LLM) reasoning to generate automated, human-readable explanations. Applied to a career recommendation context, the framework successfully provides both instance-specific and class-level justifications, expressed in natural language without requiring domain expert input. Expert evaluation across diverse user profiles, covering approximately 33% of the career clusters, yielded mean Likert scores ranging from 3.7 to 4.4 on a scale of 1 to 5, indicating moderately high acceptance of the generated explanations. These findings affirm the framework's capacity to deliver interpretable, scalable, and transparent justifications that support trust and usability in decision-making. A key limitation lies in the potential misalignment of data-driven explanations with domain expectations in certain cases, due to the absence of expert knowledge. Future enhancements will explore the integration of domain-informed refinements and optimize the fusion layer to reduce dependence on large language models for explanation fine-tuning. Overall, this work contributes a generalizable and adaptable approach to natural language explanation generation in recommender systems, with promising applicability across diverse domains.

References

[1] A. J. Wulf and O. Seizov, "Please understand we cannot provide further information": Evaluating content and transparency of GDPR-mandated AI disclosures," *AI & SOCIETY*, vol. 39, no. 1, pp. 235-256, 2024. https://doi.org/10.1007/s00146-022-01424-z

[2] M. Ibrahim, I. S. Bajwa, N. Sarwar, F. Hajjej, and H. A. Sakr, "An intelligent hybrid neural collaborative filtering approach for true recommendations," *IEEE Access*, vol. 11, pp. 64831-64849, 2023.

[3] B. Suvarna and S. Balakrishna, "Enhanced content-based fashion recommendation system through deep ensemble classifier with transfer learning," *Fashion and Textiles*, vol. 11, no. 1, p. 24, 2024. https://doi.org/10.1186/s40691-024-00382-y

[4] S. F. Ahmed *et al.*, "Deep learning modelling techniques: Current progress, applications, advantages, and challenges," *Artificial Intelligence Review*, vol. 56, no. 11, pp. 13521-13617, 2023.

[5] Z. Abou El Houda, B. Brik, and L. Khoukhi, "Why should i trust your ids?": An explainable deep learning framework for intrusion detection systems in internet of things networks," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 1164-1176, 2022.

[6] K. Amarasinghe and M. Manic, "Explaining what a neural network has learned: Toward transparent classification," presented at the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 2019.

[7] A. A. Freitas, *Automated machine learning for studying the trade-off between predictive accuracy and interpretability in International cross-domain conference for machine learning and knowledge extraction*. Cham: Springer International Publishing, 2019.

[8] L. Schwinn, R. Raab, A. Nguyen, D. Zanca, and B. Eskofier, "Exploring misclassifications of robust neural networks to enhance adversarial attacks," *Applied Intelligence*, vol. 53, no. 17, pp. 19843-19859, 2023. https://doi.org/10.1007/s10489-023-04532-5

[9] C. Novelli, M. Taddeo, and L. Floridi, "Accountability in artificial intelligence: What it is and how it works," *AI & SOCIETY*, vol. 39, no. 4, pp. 1871-1882, 2024. https://doi.org/10.1007/s00146-023-01635-y

[10] V. Viswan, N. Shaffi, M. Mahmud, K. Subramanian, and F. Hajamohideen, "Explainable artificial intelligence in Alzheimer's disease classification: A systematic review," *Cognitive Computation*, vol. 16, no. 1, pp. 1-44, 2024.

[11] V. Hassija *et al.*, "Interpreting black-box models: A review on explainable artificial intelligence," *Cognitive Computation*, vol. 16, no. 1, pp. 45-74, 2024. https://doi.org/10.1007/s12559-023-10179-8

[12] C. Molnar *et al.*, "General pitfalls of model-agnostic interpretation methods for machine learning models," presented at the International Workshop on Extending Explainable AI Beyond Deep Models and Classifiers (pp. 39-68). Cham: Springer International Publishing, 2020.

[13] Y.-J. Jung, S.-H. Han, and H.-J. Choi, "Explaining CNN and RNN using selective layer-wise relevance propagation," *IEEE Access*, vol. 9, pp. 18670-18681, 2021.

[14] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[15] A. M. Salih *et al.*, "A perspective on explainable artificial intelligence methods: SHAP and LIME," *Advanced Intelligent Systems*, vol. 7, no. 1, p. 2400304, 2025. https://doi.org/10.1002/aisy.202400304

[16] Q. Ai and R. Narayanan, L., "Model-agnostic vs. model-intrinsic interpretability for explainable product search," in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 2021.

[17] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," *International Journal of Computer Vision*, vol. 128, no. 2, pp. 336-359, 2020. https://doi.org/10.1007/s11263-019-01228-7

[18] V. Vimbi, N. Shaffi, and M. Mahmud, "Interpreting artificial intelligence models: A systematic review on the application of LIME and SHAP in Alzheimer's disease detection," *Brain Informatics*, vol. 11, no. 1, p. 10, 2024. https://doi.org/10.1186/s40708-024-00222-1

[19] V. Jahmunah, E. Y. K. Ng, R.-S. Tan, S. L. Oh, and U. R. Acharya, "Explainable detection of myocardial infarction using deep learning models with Grad-CAM technique on ECG signals," *Computers in Biology and Medicine*, vol. 146, p. 105550, 2022. https://doi.org/10.1016/j.compbiomed.2022.105550

[20] D. Macha, M. Kozielski, Ł. Wróbel, and M. Sikora, "RuleXAI—A package for rule-based explanations of machine learning model," *SoftwareX*, vol. 20, p. 101209, 2022. https://doi.org/10.1016/j.softx.2022.101209

[21] J. P. Poli, W. Ouerdane, and R. Pierrard, "Generation of textual explanations in XAI: The case of semantic annotation," presented at the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 2021.

[22] F. Gedikli, D. Jannach, and M. Ge, "How should I explain? A comparison of different explanation types for recommender systems," *International Journal of Human-Computer Studies*, vol. 72, no. 4, pp. 367-382, 2014. https://doi.org/10.1016/j.ijhcs.2013.12.007

[23] D. Afchar, A. Melchiorre, M. Schedl, R. Hennequin, E. Epure, and M. Moussallam, "Explainability in music recommender systems," *AI Magazine*, vol. 43, no. 2, pp. 190-208, 2022. https://doi.org/10.1002/aaai.12056

[24] Y. Yang, C. Zhang, X. Song, Z. Dong, H. Zhu, and W. Li, "Contextualized knowledge graph embedding for explainable talent training course recommendation," *ACM Transactions on Information Systems*, vol. 42, no. 2, pp. 1-27, 2023. https://doi.org/10.1145/3597022

[25] T. H. A. Tran, "Explainable artificial intelligence in job recommendation systems," Master's Thesis, University of Twente, 2023.

[26] M. Y. Shams, S. A. Gamel, and F. M. Talaat, "Enhancing crop recommendation systems with explainable artificial intelligence: A study on agricultural decision-making," *Neural Computing and Applications*, vol. 36, no. 11, pp. 5695-5714, 2024. https://doi.org/10.1007/s00521-023-09391-2

[27] S. Tang *et al.*, "Prostate cancer treatment recommendation study based on machine learning and SHAP interpreter," *Cancer Science*, vol. 115, no. 11, pp. 3755-3766, 2024. https://doi.org/10.1111/cas.16327

[28] S. Ahmed, M. S. Kaiser, M. S. Hossain, and K. Andersson, "A comparative analysis of LIME and SHAP interpreters with explainable ML-based diabetes predictions," *IEEE Access*, vol. 13, pp. 37370-37388, 2024.

[29] M. Setzu, R. Guidotti, A. Monreale, F. Turini, D. Pedreschi, and F. Giannotti, "Glocalx-from local to global explanations of black box ai models," *Artificial Intelligence*, vol. 294, p. 103457, 2021. https://doi.org/10.1016/j.artint.2021.103457

[30] M. A. K. Akhtar, M. Kumar, and A. Nayyar, *Transparency and accountability in explainableAI: Best practices in towards ethical and socially responsible explainable AI: Challenges and opportunities*. Cham: Springer Nature Switzerland, 2024.

[31] M. Chromik, "Making SHAP rap: Bridging local and global insights through interaction and narratives," in *Human-Computer Interaction–INTERACT 2021: 18th IFIP TC 13 International Conference, Bari, Italy, August 30–September 3, 2021, Proceedings, Part II 18 (pp. 641-651). Springer International Publishing*, 2021.

[32] F. Xu, H. Uszkoreit, Y. Du, W. Fan, D. Zhao, and J. Zhu, "Explainable AI: A brief survey on history, research areas, approaches and challenges," in *Natural Language Processing and Chinese Computing: 8th cCF International Conference, NLPCC 2019, Dunhuang, China, October 9–14, 2019, Proceedings, part II 8 (pp. 563-574). Springer International Publishing*, 2019.

[33] R. Dwivedi *et al.*, "Explainable AI (XAI): Core ideas, techniques, and solutions," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1-33, 2023.

[34] R. Saleem, B. Yuan, F. Kurugollu, A. Anjum, and L. Liu, "Explaining deep neural networks: A survey on the global interpretation methods," *Neurocomputing*, vol. 513, pp. 165-180, 2022. https://doi.org/10.1016/j.neucom.2022.09.129

[35] M. Bacevicius and A. Paulauskaite-Taraseviciene, "Machine learning algorithms for raw and unbalanced intrusion detection data in a multi-class classification problem," *Applied Sciences*, vol. 13, no. 12, p. 7328doi: https://doi.org/10.3390/app13127328.

[36] SHAP, "Shap.KernelExplainer," 2024. https://shap.readthedocs.io/en/latest/generated/shap.KernelExplainer.html

[37] P. Pino, D. Parra, C. Besa, and C. Lagos, "Clinically correct report generation from chest x-rays using templates," presented at the In International Workshop on Machine Learning in Medical Imaging (pp. 654-663). Cham: Springer International Publishing, 2021.

[38] Y. Liu, Z. Liu, X. Luo, and H. Zhao, "Diagnosis of Parkinson's disease based on SHAP value feature selection," *Biocybernetics and Biomedical Engineering*, vol. 42, no. 3, pp. 856-869, 2022. https://doi.org/10.1016/j.bbe.2022.06.007

[39] H. Gong, Y. Li, J. Zhang, B. Zhang, and X. Wang, "A new filter feature selection algorithm for classification task by ensembling pearson correlation coefficient and mutual information," *Engineering Applications of Artificial Intelligence*, vol. 131, p. 107865, 2024. https://doi.org/10.1016/j.engappai.2024.107865

[40] J. Cohen, *Statistical power analysis for the behavioral sciences*. New York: Routledge, 2013. https://doi.org/10.4324/9780203771587

[41] ONET Resource Center, "Abilities," ONET OnLine, 2022. https://www.onetonline.org/find/descriptor/browse/1.A

[42] Directorate of Higher Secondary Education, "Career guidance portal – kerala differential aptitude test (K-DAT)," Directorate of Higher Secondary Education, 2023. https://www.careerguidance.dhse.kerala.gov.in/

[43] A. Joshi, S. Kale, S. Chandel, and D. K. Pal, "Likert scale: Explored and explained," *British Journal of Applied Science & Technology*, vol. 7, no. 4, pp. 396-403, 2015. https://doi.org/10.9734/BJAST/2015/14975