# An approach to Evaluative AI through Large Language Models

Andrea Ermellino[1,*,†], Lorenzo Malandri[2,3,†], Fabio Mercorio[2,3,†], Navid Nobani[2,3,†] and Antonio Serino[4,†]

[1]*Data Science & Responsible AI, Intesa Sanpaolo, Milan, Italy*
[2]*Department of Statistics and Quantitative Methods, University of Milano-Bicocca, Italy*
[3]*CRISP Research Centre, University of Milano-Bicocca, Italy*
[4]*Department of Economics, Management and Statistics, University of Milano-Bicocca, Italy*

## Abstract

eXplainable AI (XAI) has been gaining research interest across several AI applications. However, current XAI methods often fall short of involving the user in the decision-making process, as XAI explains to the user a decision already made by the algorithm, preventing the user from evaluating alternatives. In this setting, Evaluative AI encourages balanced human-AI collaboration by addressing the issues of over- and under-reliance on AI systems and involving the user in evaluating the pros/cons of each recommendation.

In this paper, we present EADS (Evaluative AI-based Decision Support), a framework that connects Evaluative AI with conversational explanations realised via Large Language Models (LLMs). The Evaluative AI approach enables users to actively contribute with their domain knowledge and expertise to more effective and robust decision-making processes. Large Language Models enrich the explainer's output with natural language conversational explanations to present pros, cons, and neutral aspects of ML model alternatives, empowering users to evaluate options for informed, hypothesis-driven decision-making. Following the implementation of the conversational framework as per our proposed formalization, our conducted user study serves as compelling evidence of its efficacy in enhancing decision-making processes. The results of the user study demonstrate that EADS , through the fusion of human expertise and AI capabilities, presents a highly promising avenue for elevating the explainability, transparency, and overall efficacy of decision support systems across diverse domains.

## Keywords

Large Language Models, Explainable AI, Human-Centered AI



## 1. Introduction

Many state-of-the-art (SOTA) techniques in AI aim at explaining AI decisions. Depending on the underlying model and purpose, explanations can be provided either at a global level, giving insight into the overall functioning of the model, or at a local one, explaining a single output of the model. (e.g., [1, 2, 3, 4, 5]). These techniques aim to explain the model's inner workings that led to the output generation and can be divided into different categories such as counterfactual explanations, prototype and criticism, and contrastive explanations [6]. Regardless of the explanation technique and its level of presentation, the current eXplainable AI (XAI) frameworks - applied to decision support systems - provide the decision maker with a set of explanations about how an outcome recommended by the AI model was realised and the reasons for accepting or rejecting it. However, this paradigm does not involve users in the decision but only allows them to accept or reject the machine's result.

As argued by Miller [7], this approach limits engagement and, as a consequence, the trust of decision-makers or, even worse, it makes them accept the recommended decision blindly. Moreover, not knowing the full range of possibilities limits the user's ability to form his or her own opinions and possibly correct the machine's behaviour.

*27th European Conference on Artificial Intelligence, Workshop on Multimodal, Affective, and Interactive Explainable AI*
*Corresponding author.
†These authors contributed equally.
andrea.ermellino@intesasanpaolo.com (A. Ermellino); lorenzo.malandri@unimib.it (L. Malandri); fabio.mercorio@unimib.it (F. Mercorio); navid.nobani@unimib.it (N. Nobani); a.serino3@campus.unimib.it (A. Serino)
https://fabiomercorio.github.io/ (F. Mercorio)
0009-0005-1297-9185 (A. Ermellino); 0000-0002-0222-9365 (L. Malandri); 0000-0001-6864-2702 (F. Mercorio); 0000-0001-9964-097X (N. Nobani); 0009-0008-0737-8547 (A. Serino)

Following this direction, Miller [7] proposes a paradigm in which AI-based Decision Aids (DAs) do not explain a single outcome but evidence for and against possible outcomes, with the user in charge of evaluating them and making a decision. Building on top of this paradigm, our work proposes EADS (Evaluative AI-based Decision Support) framework. Our approach allows users to explore all the possible outcomes - and their explanations - through a natural language dialogue system to garner users' trust and leverage their expertise and engagement in the decision-making process. In this sense, our approach relies on the principles of *conversational explanation framework* [8, 9, 10] that exploits natural language to provide explanations and interact with the user. Specifically, we employ an LLM to generate conversational explanations that (i) make the dialogue more effective and (ii) better address the context related to the user domain, resulting in better decisions.

Motivation. Current XAI-based decision support approaches present an output to the user, who only has the choice of accepting or rejecting it based on the explanations and context provided by the agent. A system offering not pre-determined decisions but analysing the advantages and disadvantages of various alternatives would enhance the comprehensiveness of the decision-making support provided. This approach enhances explainability and allows the user to evaluate the possibilities, leading to more informed decision-making and increased user confidence in the system;

To further clarify the matter, let us consider the following mock conversation, which depicts an imaginary conversation between a user (👤) and a system (⚙) trained and deployed to classify millions of online job ads over a standard occupation taxonomy [9]:

**Classic XAI Framework**
👤: Why have you classified this job as "Data engineer"?
⚙: I saw "Data" and "Architect" words.
👤: What is the most important word for classifying this job ad?
⚙: "Architect"
👤: Which word did you not consider for the classification?
⚙: "Senior"
👤: What if the input was "Senior Python data architect"?
⚙: Then, the class would be "Data Scientist".
👤: Why instead of classifying it as "Software engineer", have you chosen "Data engineer"?
⚙: I never saw titles containing the word "architect" being labelled as "Software engineer" in the training data.
👤: Why did you ignore the word "senior"?
⚙: I consider this word as "too general".
👤: Answer me what "too general" means by considering me as a data scientist.
⚙: The TF-IDF score of the word "senior" - calculated based on the training data - is distant from the score of the rest of the terms.
👤: What if I tell you that I'm your developer?
⚙: I would say that I took that decision because the word "senior" is on the list of stopwords.

This mock example illustrates a conversational explanation framework in which the user interacts with an XAI agent- as a part of a decision support system- to understand its decision [8, 9, 10]. The user may trust the XAI agent's explanations, such as accepting the classification of "Data engineer" for a specific job ad, or may not trust it.It's important to note that the user is only partially involved in the decision-making process through interacting with the agent to understand the system's output. The same example, converted in the approach we propose, can be depicted as follows:

**Evaluative AI Framework**
👤: How could this job ad be classified?
⚙: It could be *"Data engineer"*, *"Data Scientist"* or *"Software engineer"*.
👤: Why could it be "Data engineer"?
⚙: It could be "Data engineer" because I saw the words "Data" and "Architect"
👤: Why could it be "Data scientist"?
⚙: It could be "Data Scientist" because I saw the word "Data", but I didn't see words like "Python" or "R"
👤: Why could it be "Software engineer"?
⚙: It could be "Software engineer" because I saw the word "Data", but I've never seen titles containing the word "Architect" being labelled as "Software engineer" in the training data.

In this new conversation, the user interacts with an Evaluative AI framework that provides pros and cons for each potential job classification in response to a specific job ad, subject to be classified. As can be seen from the example, at the end of the conversation, the system does not justify (i.e. explain) a definitive output but rather supports the user in the decision-making process by presenting the pros and cons of each available outcome (e.g. *Data engineer* and *Data Scientist*). Contrary to the initial example, in the Evaluative AI-based decision system, the user controls the decision-making process, and the system serves as a helpful tool by offering evaluative information. We argue that this approach is more effective for decision support because it aligns with the cognitive decision-making process that people use when making judgements and decisions (e.g., [11, 12, 13]).

Moreover, our proposed framework makes decision-making support even more robust by using not a single XAI agent but a consensus system capable of providing a more grounded evaluation.

Contribution. The contribution of EADS is two-fold:
1. we propose an approach realising Evaluative AI - built on top of [7] - to support the decision-maker, utilising the current SOTA XAI techniques;
2. we employ conversational explanations via LLM to enhance the interaction between human and agent and to increase the informative power of explanations [10];

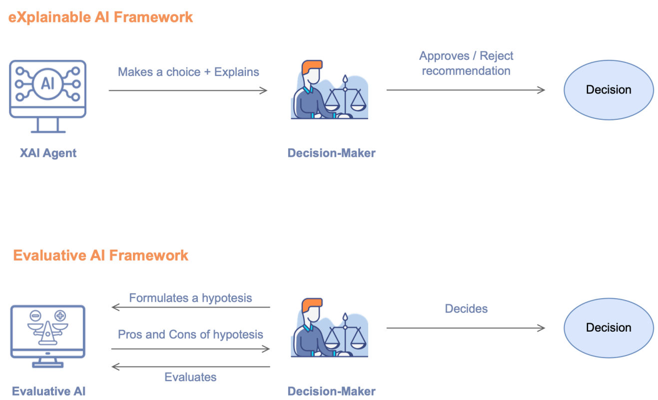


Figure 1: Comparison between eXplainable AI framework and Evaluative AI framework

## 2. Background and Related Works

eXplainable AI XAI is increasingly recognized as a critical component of AI-driven, human-centric systems across diverse sectors, including healthcare [14] to robotics [15]. XAI methods explain the internal workings and decision-making processes of otherwise opaque AI models, as highlighted by Guidotti et al. [1]. Previously viewed as a luxury, the growing complexity of AI models coupled with stringent regulatory requirements, such as the EU's General Data Protection Regulation (GDPR) which mandates a "right to explanations," has propelled XAI from an optional enhancement to a necessary feature. The GDPR emphasizes the need for "meaningful information" about the logic of automated decisions, underscoring the importance of presenting explanations in a form understandable to all users, regardless of their technical expertise. This shift not only emphasises the significance of natural language in making AI explanations accessible but also marks a pivotal evolution in the interaction between AI systems and their users.

Natural Language Explanation Although the Evaluative AI paradigm was recently introduced, the use of natural language to reinforce explanations is a popular trend. According to Sokol and Flach [16], using natural language to provide explanations is especially effective for a non-expert audience, making the information easily accessible. At the same time, narration in natural language increases the credibility of explanations and improves the user's awareness of the explanation, facilitating their approval. (see e.g., [17, 18]) This highlights the significance of natural language in making information more accessible and increasing its acceptance through a clear and effective communication approach.

Research on methodologies for Natural Language Generated (NLG) Explanations has advanced from rigid early techniques to complex and sophisticated approaches(See [10]). In its early stages, text generation was based on rule-based systems, where predefined rules set the text's structure and content [19]. Concurrently, template-based systems were developed, utilizing preset structures filled with specific data to combine flexibility and control in content generation [20, 5, 21, 22]. More recently, the adoption of deep neural networks and attention mechanisms [23] has significantly enhanced the generation of natural language explanations, surpassing the constraints of earlier methods by improving naturalness, fluency, and adaptability in various contexts [24, 25, 26].

LLMs and Prompt Engineering Recent advancements in LLMs are mitigating the constraints of requiring extensive labelled data sets and significant computational resources for task-specific tuning. Leveraging in-context learning [27] for instance, LLMs efficiently handle tasks with high accuracy [28] through one-shot and few-shot learning methods [29, 30]. These models, informed by explainers about feature contributions, provide clear and precise explanations easily understandable also by non-experts [31, 32, 33]. Despite challenges like the need for prompt engineering to ensure accuracy and avoid errors that could impact system credibility, LLMs show promise in surpassing the limitations of traditional rule and template-based approaches.

## 3. Formalising Evaluative AI

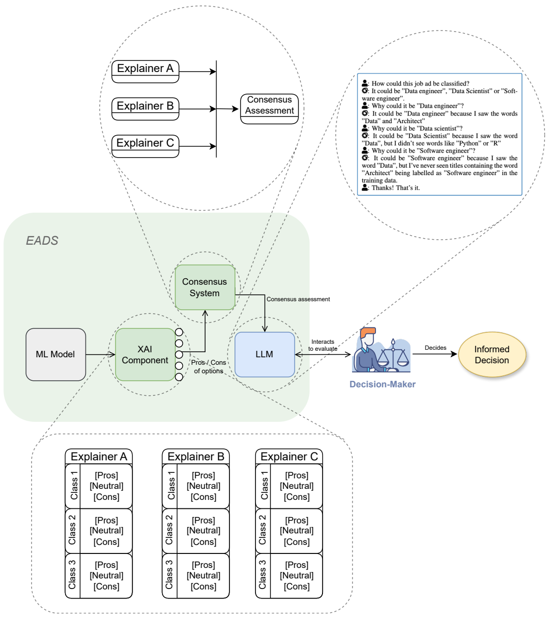


Figure 2: EADS architecture

Fig. 2 shows EADS architecture. The individual components are briefly described below:
Machine Learning Model: a machine learning "soft" classifier that provides a classification score for each class available for a given instance;
XAI Component: the component responsible for obtaining explanations of all possible outcomes provided by the machine learning classifier;
Consensus System: The XAI Component consists of $N$ distinct explainers. Its output is a synthesized interpretation, representing a collective view derived from all individual explainers. To reinforce the

decision-making process, the outputs from each explainer are integrated within a consensus system. This system quantifies the degree of agreement among the explainers.
LLM Component: The LLM component serves two primary functions. Firstly, it functions as a conversational agent, enabling users to interact and gain insights from the explanations provided by the XAI component. Secondly, it acts as an agent that enhances the context of these explanations, thereby enriching their informational value.

### 3.1. Problem Formulation

We build and formalise our framework on top of the Evaluative AI framework introduced by Miller [7]. First of all, we generalize the classification problem as described by Guidotti et al. [1]:

**Definition 3.1** (Machine Learning "Black Box" Classifier)**.** A predictor, also named model or classifier, is a function

$$b : \mathcal{X}^{(m)} \to \mathcal{Y} \tag{1}$$

which maps data instances (tuples) $x$ from a feature space $\mathcal{X}^{(m)}$ with $m$ features to a decision $y$ in a target space $\mathcal{Y}$. We write $b(x) = y$ to denote the decision $y$ predicted by $b$, and $b(\mathcal{X}) = \mathcal{Y}$ as a shorthand for $\{b(x) \mid x \in \mathcal{X}\} = \mathcal{Y}$. An instance $x$ consists of a set of $m$ attribute-value pairs $(a_i, v_i)$, where $a_i$ is a feature (or attribute) and $v_i$ is a value from the domain of $a_i$. The domain of a feature can be continuous or categorical. The target space $\mathcal{Y}$ (with dimensionality equal to one) contains the different labels (classes or outcomes), and also, in this case, the domain can be continuous or categorical. Note that, in the case of ordinal classification, labels in $\mathcal{Y}$ have an order. A predictor $b$ can be a machine-learning model, a domain expert rule-based system, or any combination of algorithmic and human knowledge processing. In the following, we denote by $b$ a black box predictor whose internals are either unknown to the observer or are known but uninterpretable by humans.

A Machine Learning "Black Box" Classifier underlies the XAI Component in implementing an Evaluative AI framework. In the XAI component, a key feature is the presence of $N$ explainers, collectively forming a consensus system. Each explainer $e_n$ within the XAI component takes as input all scores $s_j$ produced by the soft Machine Learning Black Box classifier for every label $y_i \in \mathcal{Y}$, given a single data instance $x$.

In this setup, the soft Machine Learning Black Box classifier refers to the machine learning model that outputs probabilistic scores for each class label $y_i$ when presented with the input data instance $x$. These probabilistic scores $s_j$ represent the model's confidence in the likelihood of each class $y_i$ being the correct prediction for $x$. Each explainer $e_n$ takes these scores $s_j$ as input and performs its analysis, contributing to the collective interpretation of the XAI component. The aggregation of insights from all explainers in the consensus system enhances the transparency and interpretability of the overall decision-making process of the machine learning model, promoting a comprehensive understanding of the predictions for the given data instance $x$. Burkart and Huber [6] define the explanation generation problem as follows:

**Definition 3.2** (Explanation Generation)**.** An explanator function $e$ is defined as

$$e : (\mathcal{X} \to \mathcal{Y}) \times (\mathcal{X} \times \mathcal{Y}) \to \mathcal{E} \tag{2}$$

which takes a supervised Machine Learning (SML) model (black box or interpretable) and a specific data set as input and provides an explanation belonging to the set $\mathcal{E}$ of all possible explanations as an output. There are two possible explanation generation problems:
Global Extracting a global explanation from a model that is representative of some specific data set $\mathcal{D}'$, i.e., $e(b, \mathcal{D}')$ in case of a black box model or $e(c, \mathcal{D}')$ for interpretable models;
Local Instance explanators extract an explanation for a single test input $x$ and the corresponding prediction $y$, i.e., $e(b, (x, y))$ or $e(c, (x, y))$.
Given this definition, we introduce the first type of explanation of an SML model, namely the direct interpretation of a given black box model in a post-hoc fashion. This is achieved by utilizing global, often model-agnostic, explanators. A well-known example is partial dependency plots [34].

The XAI Component produces a collective interpretation derived from its underneath explainers, implemented to provide explanations in the form of numerical scores within the range of [-1, +1]. Each explainer $e_n$ assesses the contribution of each feature in the feature space $\mathcal{X}^{(m)}$ to the classification of the instance $x$ in the class $y_i \in \mathcal{Y}$. This assessment classifies each feature as one of the following:

Pros: if the feature positively influenced the classification and the explainer assigns a score $> k$;
Cons: if the feature negatively influenced the classification and the explainer assigns a score $< -k$;
Neutral: if the explainer assigns a score between $-k$ and $+k$ the feature had no significant impact on the classification.

Where $k$ is an arbitrary threshold. Therefore, each explainer $e_n$ of the XAI Component can determine the contribution of individual features to the classification outcome. The output of the individual explainer $e_n$ is an object defined as *Evaluation*, formalised as:

**Definition 3.3** (Evaluation)**.** An evaluation $ev_n$ produced by the explainer $e_n$ is a mapping that associates, for each class $y_i \in \mathcal{Y}$, a categorization label from the set $\mathcal{L} = \{PRO, CON, NEUTRAL\}$ to each feature of the feature space $\mathcal{X}^{(m)}$. Formally:

$$ev : \mathcal{Y} \to (\mathcal{X} \to \mathcal{L}) \quad (3)$$

As a result, the $ev_n$ evaluation produced by the $e_n$ explainer contains features categorised as pro, con and neutral to the classification of the data instance $x$.

The collective interpretation of the XAI Component is derived from its explainers $e_n$ by assessing the level of *Consensus*. We adopt a Consensus measure proposed in [35] to assess the level of agreement of XAI Component individual explainers $e_n$ as follows.

Similarly, Le et al. [36] propose Weight of Evidence (WoE) as a probabilistic method for analysing the variable importance. To do so, the authors propose summing the WoE of each feature $X_i$ to calculate the WoE of the hypothesis $h$, articulated by the decision maker.

**Definition 3.4** (Consensus Measure)**.** Consensus Measure is a measure of dispersion introduced as a representation of agreement and disagreement. Building on the generally accepted Shannon entropy, this measure uses a probability distribution and the distance between categories to produce a value spanning the unit interval. The measure is applied to the Likert scale (or any ordinal scale) to determine degrees of consensus. Using this measure, data on ordinal scales can be given a value of dispersion that is both logically and theoretically sound. Since consensus is a function of shared group feelings towards an issue, this "feeling" can be captured through a Likert scale that measures the extent to which an entity agrees or disagrees with the question. The Likert scale adopted in our framework is shown in Tab. 3.4 where:

- $es_{ni}$ is the numeric score representing the impact of feature $a_i$ on the classification of data instance $x$ for the individual explainer $e_n$;
- $k$ is an arbitrary threshold;
- *Evaluation Mapping* value is the result of Eq. 3.

| Explainer Score ($es_{ni}$) | Evaluation Mapping | Likert Value |
|---|---|---|
| $es_{ni} < -k$ | CON | 1 |
| $-k < es_{ni} < k$ | NEUTRAL | 2 |
| $es_{ni} > k$ | PRO | 3 |

Table 1
Consensus Likert scale

Tastle and Wierman [35] define a set of rules that must be satisfied in a Likert scale consensus problem:

1. For a given (even) number of individuals participating in a discussion on some question of interest, if an equal number of individuals, *n/2*, separate themselves into two disjoint groups, each centred on the strongly disagree and strongly agree categories, the group is considered to have no consensus;

2. if all the participants classify themselves in the same category of the Likert scale, regardless of the category, then the consensus of the group is considered to be complete at 100%;
3. if the mix of participants is such that $n/2+1$ participants assign themselves to any one category, the degree of consensus must be greater than 0, for the balance in the group is no longer equal at the extreme categories.

Hence, a complete lack of consensus generates a value of 0, and a complete consensus of opinion yields a value of 1. Every other combination of Likert scale categories must result in a value within the unit interval. The consensus value of $a_i$ feature for a given class $y_i \in \mathscr{Y}$ is defined as:

$$Cns(a_i) = 1 + \sum_{j=1}^{n} p_j \log_2(1 - \frac{|LV_j - \mu_{LV}|}{d_{LV}}) \tag{4}$$

Where:

- $p_j$ is the probability (relative frequency) of outcome $LV_j$ (which ranges from 1 to 3);
- $LV_j$ is the outcome in Likert scale value;
- $\mu_{LV}$ is the weighted mean of $LV$ using probabilities $p$ as weights;
- $d_{LV}$ is the dimension of Likert scale adopted.

Using this measure, a collective interpretation of the XAI Component can be obtained, also quantifying the degree of agreement of the individual explainers that make up the component.

In this framework, a LLM is used to generate natural language, providing the user with textual explanations and giving them context. Following [37], the LLM is formalised as follows:

**Definition 3.5** (Large Language Model)**.** Language models are a fundamental building block of current SOTA natural language processing pipelines. While the objectives used to train these models vary, one popular choice is a next-step prediction objective. This approach constructs a generative model of the distribution

$$Pr(t_1, t_2, ..., t_n) \tag{5}$$

where $t_1, t_2, ..., t_n$ is a sequence of tokens from a vocabulary $\mathscr{V}$ by applying the chain rule of probability

$$Pr(t_1, t_2, ..., t_n) = \prod_{i=1}^{n} Pr(t_i \mid t_1, ..., t_{i-1}) \tag{6}$$

SOTA LMs use neural networks to estimate this probability distribution. We let $f_\theta(t_i \mid t_1, ..., t_{i-1})$ denote the likelihood of token $t_i$ when evaluating the neural network $f$ with parameters $\theta$.

User can interact with the LLM by giving input prompts:

**Definition 3.6** (Prompt)**.** A prompt is a sequence $t_1, ..., t_i$ tokens that condition the Large Language Model text generation process. Given an input prompt a language model can generate new text by iteratively sampling $\hat{t}_{i+1} \sim g_\theta(t_{i+1} \mid t_1, ..., t_i)$ and then feeding $\hat{t}_{i+1}$ back into the model to sample $\hat{t}_{i+2} \sim g_\theta(t_{i+2}|t_1, ..., \hat{t}_{i+1})$. This process is repeated until a desired stopping criterion is reached. Variations of this text generation method include deterministically choosing the most-probable token rather than sampling (i.e., greedy sampling) or setting all but the top-n probabilities to zero and renormalizing the probabilities before sampling (i.e., top-n sampling [38]) [37].

### 3.2. From Explaining to Supporting Decisions

We introduce EADS, offering a new approach for DA systems to transition from the conventional XAI framework to the Evaluative AI framework, as suggested by Miller [7]. One of the main limitations of current XAI approaches when applied to decision support systems is that they provide the user with explanations of a single answer (i.e. output of the black box), as seen in the first conversation example. This approach limits the decision-maker's ability to evaluate different choices. More precisely, the limitation in the first conversation example lies in the "hard classification" approach of the system, which identifies and explains a single best choice. Otherwise, we adopt a "soft classification" approach, i.e., a membership/classification score for each possible class. In this way, the evaluative system supports the decision-making process by presenting a set of possible choices with pros, cons and neutral features, enabling the decision-maker to guide the process with the system's support, resulting in a machine-in-the-loop paradigm. Moreover, we supplement our approach with two additional components to make the underlying model more understandable, accountable and transparent. The first is an LLM to enhance the interaction between the AI agent and the human decision-maker. The second one gives users more robust information thanks to a consensus system of individual explainers.

### 3.3. Framework Components

In this section, the implementation of the framework is shown step-by-step, taking up the four fundamental elements described in Sec. 3.1: 1. Machine Learning Black Box (soft) Classifier training, 2. XAI Component construction and Evaluation generation, 3. Consensus assessment, and 4. Prompt engineering.

#### 3.3.1. ML (soft) classifier training

First, a model is trained on a training dataset $D_{train}$, and its performance is evaluated on a test dataset $D_{test}$ (using the hard classification approach). Therefor, given a new data instance $x$ to be evaluated, the trained model must no longer return the single prediction $y$ (hard classification) but produces a score $s_j$ (see Sec. 3.1) for each target class $y_i \in \mathcal{Y}$, thus softly approaching the classification problem as described above (see Sec. 3.1 and 3.2).

#### 3.3.2. XAI Component construction and Eval generation

The next step is constructing the XAI Component and Evaluation generation. As described in Sec. 3.1, the component consists of $N$ individual explainers. Each one, given an instance $x$, for each class $y_i \in \mathcal{Y}$ evaluates the impact of each feature within the feature space $\mathcal{X}^{(m)}$ on the classification of $x$ in class $y_i$. The individual explainer $e_n$ expresses the impact of the feature in the form of a numerical score in the range [-1, +1].

Algorithm 1 shows how the *Evaluation* of an individual explainer $e_n$ for a given data instance $x$ is obtained. In Algorithm 1:

- *model* is the trained Machine Learning soft classifier;
- *explainer* is an individual explainer of the XAI Component; the method *explainer(model)* is used as a placeholder for the actual explainer instantiation method, the signature of which varies depending on the library used;
- $k$ is the arbitrary threshold for the Likert Mapping step;
- *pros, cons, neutral* are the sets containing categorised features as a result of the evaluation mapping step;
- the method *explain(x)* is used as a placeholder for the actual explainer $e_n$ explanation generation method, the signature of which varies depending on the library used;

**Algorithm 1:** EVALUATION

```
Input: x, model, explainer, k
Output: evaluation
pros,cons,neutral ← ∅;
evaluation ← {};
e_n ← explainer(model);
explanation ← e_n.explain(x);
for y_i in 𝒴 do
    for feature in 𝒳^(m) do
        if explanation(y_i, feature) > k then
            pros.append(feature);
        else
            if explanation(y_i, feature) < −k then
                cons.append(feature);
            else
                neutral.append(feature);
    class_evaluation ← {"PROS": pros, "CONS": cons, "NEUTRAL": neutral};
    evaluation.append({y_i: class_evaluation})
return evaluation
```

- *explanation($y_i$, feature)* is the numerical score given by the explainer $e_n$ to assess the impact of $feature$ on the classification of data instance $x$ in $y_i$ class.

XAI Component is constructed by repeating Algorithm 1 $N$ times, one for each explainer. The result of the $N$ executions is $N$ *Evaluations* that are used as input for the next implementation step.

#### 3.3.3. Consensus assessment

After constructing the XAI Component and obtaining all *Evaluations* from the component's explainers, we proceed with the consensus assessment for collective interpretation of the component.

**Algorithm 2:** LIKERT SCALE MAPPING

```
Input: evaluation
Output: likert_evaluation
likert_evaluation ← ∅;
for y_i in 𝒴 do
    for feature in 𝒳^(m) do
        if evaluation(y_i, feature) ∈ CONS then
            likert_evaluation(y_i, feature) ← 1;
        else
            if evaluation(y_i, feature) ∈ NEUTRAL then
                likert_evaluation(y_i, feature) ← 2;
            else
                likert_evaluation(y_i, feature) ← 3;
return likert_evaluation
```

Algo 2 shows the mapping of *Evaluation* elements in the Likert scale described in Def. 3.4. Repeating Algo 2 $N$ times, one for each *Evaluation*, results in a complete mapping of the output of the XAI Component into a format useful for consensus assessment.

After obtaining a complete mapping of the output of the XAI Component explainers into a format useful for consensus assessment, Algo 3 shows how to obtain the consensus score for a *feature* given a class $y_i \in \mathcal{Y}$ following Eq. 4. In Algo 3:

- *likert_evaluations* contains all the *evaluations* resulting from $N$ iterations of Algo 2;

**Algorithm 3:** CONSENSUS

**Input:** *likert_evaluations, $y_i$, feature*<br>
**Output:** *consensus_score*<br>
likert_outcome $\leftarrow [1, 2, 3]$;<br>
$p_1, p_2, p_3 \leftarrow 0$;<br>
**for** *evaluation **in** likert_evaluations* **do**<br>
    **if** *evaluation($y_i$, feature)* $== 1$ **then**<br>
        $p_1 \leftarrow p_1 + 1$;<br>
    **else**<br>
        **if** *evaluation($y_i$, feature)* $== 2$ **then**<br>
            $p_2 \leftarrow p_2 + 1$;<br>
        **else**<br>
            $p_3 \leftarrow p_3 + 1$;<br>
$N \leftarrow$ length(*likert_evaluations*);<br>
$rf_1 \leftarrow \frac{p_1}{N}$;<br>
$rf_2 \leftarrow \frac{p_2}{N}$;<br>
$rf_3 \leftarrow \frac{p_3}{N}$;<br>
$\mu_{LV} \leftarrow \sum_{j=1}^{3} rf_j \cdot likert\_outcome[j]$;<br>
$d_{LV} \leftarrow 3$;<br>
$consensus\_score \leftarrow 1 + \sum_{j=1}^{3} p_j \log_2(1 - \frac{|likert\_outcome[j] - \mu_{LV}|}{d_{LV}})$;<br>
**return** (*consensus_score*)

- *likert_outcome* is a set containing all possible outcomes of the Likert scale described in Tab. 3.4.

Repeating Algo 3 for each class $y_i \in \mathcal{Y}$ and each feature in the feature space $\mathcal{X}^{(m)}$ yields all the consensus scores needed for the next step.

Following the formalisation described in Sec. 3.1, we implemented the framework using the benchmark *Iris* dataset for a preliminary study. The black box "soft" classifier trained on a portion of the dataset was a Random Forest and the XAI Component was made by SHAP [39], LIME [40] and ELI5[1] explainers. We set the threshold $k = 0.10$ for the Evaluation generation following Algo 1. After the results were obtained, the Likert Mapping and Consensus Assessment steps were carried out following Algoss 2 and 3.

The results of the last two steps were structured in two different JSONs and formatted in a prompt explaining the task to be performed. Moreover, the information needed to interpret the two JSONs is provided so that the model can understand what kind of requests the user can make and how it should fulfil them, thus giving the LLM a frame of action. The formatted prompt constitutes the system prompt of the model. After sending this instruction, the framework is ready to use.

The following conversation is extracted from a real interaction with EADS:

**Implemented Evaluative AI Framework**

: How could this instance be classified?

: It could be "Virginica", "Versicolor" or "Setosa".

: I want to know about "Setosa"

: Certainly! Here's a breakdown of the features and their impact on the classification of the instance under the class "Setosa":

1. **sepal length (cm)**:
   - Considered as a CON by 0 explainers.
   - Considered NEUTRAL by 3 explainer.
   - Considered as a PRO by 0 explainers.
   - Consensus: 100%
2. **sepal width (cm)**:
   - Considered as a CON by 0 explainers.
   - Considered NEUTRAL by 3 explainers.
   - Considered as a PRO by 0 explainers.
   - Consensus: 100%
3. **petal length (cm)**:

[1] https://eli5.readthedocs.io/en/latest/index.html

- Considered as a CON by 2 explainers.
- Considered NEUTRAL by 1 explainer.
- Considered as a PRO by 0 explainers.
- Consensus: 76.59%

4. **petal width (cm)**:
   - Considered as a CON by 2 explainers.
   - Considered NEUTRAL by 1 explainer.
   - Considered as a PRO by 0 explainers.
   - Consensus: 76.59%

For the Setosa class, the features *"sepal length (cm)"* and *"sepal width (cm)"* are considered neutral by all explainers with a full consensus. However, the features *"petal length (cm)"* and *"petal width (cm)"* are mostly viewed as cons, with a slightly lesser consensus of 77%.

: And what about "Versicolor"?

: Here's the explanation for "Versicolor":

1. **sepal length (cm)**:
   - Considered as a CON by 0 explainers.
   - Considered NEUTRAL by 3 explainer.
   - Considered as a PRO by 0 explainers.
   - Consensus: 100%

. . .

### 3.4. User Evaluation

For the user study we recruited 50 users using the Prolific[2] platform. The participants have been remunerated in accordance with the prevailing average compensation rate on Prolific. To take part in the user study, the participants had to be fluent in English and had to have a bachelor's degree or higher. Since EADS is meant to be used by domain experts and not only and exclusively by LLM and XAI experts, we decided NOT to filter participants based on their technical/professional profile. They took on average 8.12 minutes to finish the survey. A sample of the user study is provided[3].

The respondents where provided each one with a data instance of the *20 newsgroups* dataset, a commonly used benchmark dataset in XAI (e.g. in [40, 20]). It is a collection of approximately 20,000 newsgroup documents, partitioned across 20 different topics including politics, sports, and technology. In the user study, we considered a subset of 4 classes: *atheism, christian, space, and graphics.* The document provided to the respondents belonged to the class *atheism*. Please notice that the classes *atheism* and *christian* are challenging to distinguish, because the two topics are often mixed in the documents and because they contain features that not generalise outside the validation set [40].

In addition to the document, the participants where given 10 interactions with EADS , useful for making an informed decision about the provided instance. After reading the document and EADS responses to the 10 questions, the user was asked to classify the document into one of 4 topics. After making this decision, the true label of the document was shown and then asked to answer a questionnaire of 4 questions. The questions have been chosen according with recent literature in XAI evaluation [41] to test the following 5 properties of the system:

**Q1:** *Transparency* to explain how the AI system works.
**Q2:** *Trust* to increase users confidence in the AI system.
**Q3:** *Satisfaction* to increase the ease of use or enjoyment of users when interacting with an AI system.
**Q4:** *Effectiveness* to help users make good decisions.
**Q5:** *Debugging* to enable users to identify defects in the AI system.

Does EADS helps the user in the decision making process? The first four properties are tested through direct questions. The users indicate a value on a Likert scale from 1 to 4. Q1 aimed to assess the level of *Transparency* of EADS so we asked to indicate how easy was to understand EADS explanations. Value 1 stood for "very easy" and value 4 for "very difficult". For Q1, the 58% of participants indicated a value of 1-2, 32% indicated value 3 and only 10% indicated value 4. Answers to Q1 indicate how easy it was for most evaluators to understand the explanations provided verbatim by EADS . A good proportion of evaluators encountered some difficulty in the enjoyment of the explanation, indicating how in certain

[2]https://app.prolific.com/
[3]https://forms.gle/HCeA4MEWEpdcR6MM8

scenarios/tasks or in the presence of particular data instances, an explanation in text format is not always the best choice but other ways of presenting the explanation might be more suitable (e.g., images, video, multimedia).

Q2 aimed to understand how EADS helps *Trust* to increase confidence in the AI system. To users was asked if having the opinion of three different explainers increase the reliability of EADS . In this case value 1 stood for "not at all" and value 4 stood for "extremely". For Q2, the 72% of partecipants indicated a value of 3-4 and just the 28% indicated a value of 1-2. Responses to Q2 demonstrate the potential of the Consensus System, of its ability to donate robustness to the explanation provided by EADS . In the critical scenarios for which EADS was designed, the Consensus System emerges as a pivotal component. Its ability to enhance the reliability of the entire AI system and bolster the confidence of decision makers is paramount, fostering greater support throughout the cognitive process.

For Q3 instead, we asked users how satisfied they were using EADS in order to evaluate the *Satistaction* to increase the ease of use or enjoyment of user interacting with an AI system. Users could indicate value 1 for "very dissatisfied" and value 4 for "very satisfied". The 66% of partecipants indicated for Q3 a value of 3-4, demonstrating a positive overall experience with EADS .

To assess EADS *Effectiveness* to help users make good decision, Q4 asked if they believe EADS has the capability to assist decision-makers in making informed decision across different scenarios. Users could express 1 for "not at all" and 4 for "extremely". For Q4, the 68% of participants indicated a value of 3-4. Responses to Q4 support the initial thesis that having a system that gives you the opportunity to explore pros, cons, and neutral aspects of each possibility supports the cognitive process more than having a single opinion (decision) taken by the ML model and provided along with its explanation.

Does EADS helps the correctness of the classification? Property Q5 is measured through the classification of the documents. Indeed, 88% of the respondents classified the instance under the *Atheism* class, while 12% categorized it as *Christian*. None selected the *Space* and *Graphics* classes. This outcome shows that, given sufficient supporting information, users are able to correctly classify the instance. Is noteworthy that cases classified as *Christian* are not necessarily wrong, since the discussion pertains to atheism but compared with the Christian stance rather than religious positions in general (see, e.g., the provided survey sample). In such instances, a classifier, even with the correct prediction, would have chosen the *Atheism* class, leaving the end user with no other option. Instead, EADS improves the decision-making process by enabling users to consider various alternatives, and intercept misclassifications by the ML algorithm. Moreover, it is important to note that none of the users, after interacting with EADS , selected the *Space* or *Graphics* classes, which would have undoubtedly been errors. A XAI approach, on the other hand, rely on the class predicted by a classification algorithm. To test this behaviour, we implemented a SOTA classifier for 20Newsgroup [42] on the 4 classes used in the user study. The total number of instances in the four classes is 3,756, which we divided into 2,256 for training and 1,500 for testing. In the test set we have 319 instances of the class *atheism*, of which 291 (91.2%) are correctly classified, 14 (4.4%) are classified as *christian* and 14 (4.4%) in the other two classes. Therefore, EADS not only involves the user in the decision making process, but it also helps in debugging misclassifications of the classifier or of the XAI algorithm in the predicion of the correct class.

Plots/graphs of the discussed results are in the additional material.

### 3.5. Hallucination Check

Using LLMs require a particular attention to results accuracy, since a well-known problem is that of hallucinations [43]. A possible solution to mitigate this problem is refine prompt using prompt engineering [44]. Beyond user evaluation, we also checked hallucinations, testing fifty different input instances with our prompt on both Open Source models [4] [5] and Closed Source models[6][7]. In none of the

[4]https://llama.meta.com/llama3/

[5]https://mistral.ai/technology/models

[6]https://platform.openai.com/docs/models/gpt-4-turbo-and-gpt-4

[7]https://cohere.com/blog/command-r-plus-microsoft-azure

tests performed we find any inaccuracies or inconsistencies that would allow us to observe hallucinations.

As black box AI-based tools become widespread, it is crucial to emphasize the integral involvement of users in the decision-making process to ensure trustworthiness and transparency. In this paper, we propose EADS a novel framework that connects Evaluative AI with conversational explanations realised via LLMs. To realize EADS, we (i) formalize our approach and link it to the SOTA in XAI and LLM; (ii) propose a framework that supports the user with pros and cons of different possible decisions while preserving current XAI approaches, and (iii) integrate an LLM in the framework to transform the role of the decision maker from passive component of the system to a part of the system that directly interacts with it and forms her decisions based on the rich information provided by the decision system.

The user study has demonstrated that the majority of users appreciate the approach employed by EADS, which involves the user in the decision process and employes LLMs to enrich the explainer's output, enhances trust, satisfaction, and effectiveness in decision-making processes supported by ML algorithms (Q2, Q3, and Q4). Furthermore, the results of question Q5 in the user study indicate that EADS, by involving the user in the decision-making process and shifting the decision locus from the machine to the user, enable the improvement of algorithm performance and facilitate more accurate decision-making. Regarding the transparency of the output (Q1), ambivalent results show that, although almost none of the users found the system very challenging to interact with, a minority of users encountered some difficulties in comprehension. This suggests that natural language interaction may not always be the optimal interface, prompting us to consider integrating additional tools such as graphs/plots, images, and reports in future developments.

Our research delineates the limitations into two main categories: general limitations inherited from the underlying technologies and specific limitations of our proposed approach.

### General Limitations

**Hallucination in Language Models:** LLMs are known to produce hallucinated content, a challenge well-documented across numerous studies. Although many methodologies have been proposed to mitigate this issue, a definitive solution that ensures reliable detection in all cases remains elusive. Although we have tested our framework using both open source and closed source LLMs, without encountering hallucinations, our study acknowledges this limitation but does not address it directly, given the extensive ongoing research in this area.

**Intended Audience of the System:** The foundational technology of our system, Evaluative AI, is primarily designed for decision-makers. Unlike traditional XAI approaches where explanations can be tailored to different audiences, Evaluative AI targets authoritative figures within their domains, limiting its broader applicability.

### Specific Limitations

**Scalability with Increased Complexity:** The output of our framework can become overwhelming as the number of classes and features increases. This scalability issue arises from the system's design to provide consensus values for each feature and class, which can be complex to manage and interpret.

**Data Modality Constraints:** Our framework is currently implemented for text and tabular data. Its effectiveness may not extend seamlessly to other modalities, such as images, audio, or video, thus limiting its applicability in different domains.

## References


[1] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, D. Pedreschi, A survey of methods for explaining black box models, ACM computing surveys (CSUR) 51 (2018) 1–42.

[2] D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf, G.-Z. Yang, Xaiexplainable artificial intelligence, Science robotics 4 (2019) eaay7120.

[3] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al., Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai, Information fusion 58 (2020) 82–115.

[4] A. Holzinger, A. Saranti, C. Molnar, P. Biecek, W. Samek, Explainable ai methods-a brief overview, in: International workshop on extending explainable AI beyond deep models and classifiers, Springer, 2022, pp. 13–38.

[5] L. Malandri, F. Mercorio, M. Mezzanzanica, A. Seveso, Model-contrastive explanations through symbolic reasoning, Decision Support Systems 176 (2024) 114040.

[6] N. Burkart, M. F. Huber, A survey on the explainability of supervised machine learning, Journal of Artificial Intelligence Research 70 (2021) 245–317.

[7] T. Miller, Explainable ai is dead, long live explainable ai! hypothesis-driven decision support using evaluative ai, in: Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency, 2023, pp. 333–342.

[8] P. Madumal, T. Miller, L. Sonenberg, F. Vetere, A grounded interaction protocol for explainable artificial intelligence, arXiv preprint arXiv:1903.02409 (2019).

[9] L. Malandri, F. Mercorio, M. Mezzanzanica, N. Nobani, Convxai: a system for multimodal interaction with any black-box explainer, Cognitive Computation (2022) 1–32.

[10] E. Cambria, L. Malandri, F. Mercorio, M. Mezzanzanica, N. Nobani, A survey on xai and natural language explanations, Information Processing & Management 60 (2023) 103111.

[11] G. Klein, J. K. Phillips, E. L. Rall, D. A. Peluso, A data–frame theory of sensemaking, in: Expertise out of context, Psychology Press, 2007, pp. 118–160.

[12] R. R. Hoffman, T. Miller, W. J. Clancey, Psychology and ai at a crossroads: How might complex systems explain themselves?, The American Journal of Psychology 135 (2022) 365–378.

[13] C. S. Peirce, Writings of Charles S. Peirce: a chronological edition, volume 8: 1890–1892, volume 8, Indiana University Press, 2009.

[14] S. N. Payrovnaziri, Z. Chen, P. Rengifo-Moreno, T. Miller, J. Bian, J. H. Chen, X. Liu, Z. He, Explainable artificial intelligence models using real-world electronic health record data: a systematic scoping review, JAMIA (2020).

[15] X. Gao, R. Gong, Y. Zhao, S. Wang, T. Shu, S.-C. Zhu, Joint mind modeling for explanation generation in complex human-robot collaborative tasks, IEEE RO-MAN (2020).

[16] K. Sokol, P. A. Flach, Conversational explanations of machine learning predictions through class-contrastive counterfactual statements., in: IJCAI, 2018, pp. 5785–5786.

[17] A. P. Chaves, M. A. Gerosa, How should my chatbot interact? a survey on social characteristics in human–chatbot interaction design, International Journal of Human–Computer Interaction 37 (2021) 729–758.

[18] M. De Gennaro, E. G. Krumhuber, G. Lucas, Effectiveness of an empathic chatbot in combating adverse effects of social exclusion on mood, Frontiers in psychology 10 (2020) 3061.

[19] L. A. Hendricks, R. Hu, T. Darrell, Z. Akata, Generating counterfactual explanations with natural language, in: ICML Workshop on Human Interpretability in Machine Learning, 2018, pp. 95–98.

[20] L. Malandri, F. Mercorio, M. Mezzanzanica, N. Nobani, A. Seveso, Contrxt: Generating contrastive explanations from any text classifier, Information Fusion 81 (2022) 103–115.

[21] L. Malandri, F. Mercorio, M. Mezzanzanica, N. Nobani, A. Seveso, et al., The good, the bad, and the explainer: A tool for contrastive explanations of text classifiers., in: IJCAI, 2022, pp. 5936–5939.

[22] I. Donadello, M. Dragoni, Bridging signals to natural language explanations with explanation graphs, Proceedings of the 2nd Italian Workshop on Explainable Artificial Intelligence (2021).

[23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[24] U. Ehsan, P. Tambwekar, L. Chan, B. Harrison, M. O. Riedl, Automated rationale generation: a technique for explainable ai and its effects on human perceptions, in: ACM IUI, 2019, pp. 263–274.

[25] F. Costa, S. Ouyang, P. Dolog, A. Lawlor, Automatic generation of natural language explanations, in: Proceedings of the 23rd international conference on intelligent user interfaces companion, 2018, pp. 1–2.

[26] E. Kokalj, B. Škrlj, N. Lavrač, S. Pollak, M. Robnik-Šikonja, Bert meets shapley: Extending shap explanations to transformer-based classifiers, in: Proceedings of the EACL hackashop on news media content analysis and automated report generation, 2021, pp. 16–21.

[27] Q. Dong, L. Li, D. Dai, C. Zheng, Z. Wu, B. Chang, X. Sun, J. Xu, Z. Sui, A survey for in-context learning, arXiv preprint arXiv:2301.00234 (2022).

[28] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, et al., A survey of large language models, arXiv (2023).

[29] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, OpenAI blog 1 (2019) 9.

[30] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, Advances in neural information processing systems 33 (2020) 1877–1901.

[31] D. Slack, S. Krishna, H. Lakkaraju, S. Singh, Explaining machine learning models with interactive natural language conversations using talktomodel, Nature Machine Intelligence 5 (2023) 873–883.

[32] J. Chun, K. Elkins, explainable ai with gpt4 for story analysis and generation: A novel framework for diachronic sentiment analysis, International Journal of Digital Humanities 5 (2023) 507–532.

[33] T. Ali, P. Kostakos, Huntgpt: Integrating machine learning-based anomaly detection and explainable ai with large language models (llms), arXiv preprint arXiv:2309.16021 (2023).

[34] A. Goldstein, A. Kapelner, J. Bleich, E. Pitkin, Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation, journal of Computational and Graphical Statistics 24 (2015) 44–65.

[35] W. J. Tastle, M. J. Wierman, Consensus and dissention: A measure of ordinal dispersion, IJAR 45 (2007) 531–545.

[36] T. Le, T. Miller, R. Singh, L. Sonenberg, Towards the new xai: A hypothesis-driven approach to decision support using evidence, arXiv preprint arXiv:2402.01292 (2024).

[37] N. Carlini, F. Tramer, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. B. Brown, D. Song, U. Erlingsson, et al., Extracting training data from large language models., USENIX Security Symposium 6 (2021).

[38] A. Fan, M. Lewis, Y. Dauphin, Hierarchical neural story generation, arXiv preprint arXiv:1805.04833 (2018).

[39] S. M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, Advances in neural information processing systems 30 (2017).

[40] M. T. Ribeiro, S. Singh, C. Guestrin, " why should i trust you?" explaining the predictions of any classifier, ACM SIGKDD (2016).

[41] R. Confalonieri, J. M. Alonso-Moral, An operational framework for guiding human evaluation in explainable and trustworthy ai, IEEE Intelligent Systems (2023).

[42] Y. Lin, Y. Meng, X. Sun, Q. Han, K. Kuang, J. Li, F. Wu, Bertgcn: Transductive text classification by combining gnn and bert, in: ACL-IJCNLP, 2021, pp. 1456–1462.

[43] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, P. Fung, Survey of hallucination in natural language generation, ACM Computing Surveys 55 (2023) 1–38.

[44] S. Tonmoy, S. Zaman, V. Jain, A. Rani, V. Rawte, A. Chadha, A. Das, A comprehensive survey of hallucination mitigation techniques in large language models, arXiv preprint arXiv:2401.01313 (2024).