Talking Back – human input and explanations to interactive AI systems

Alan Dix[1,2], Tommaso Turchi[3], Ben Wilson[2], Anna Monreale[3] and Matt Roach[2]

[1]*Cardiff Metropolitan University, Wales, UK*

[2]*Computational Foundry, Swansea University, Wales, UK*

[3]*Department of Computer Science, University of Pisa, Pisa, Italy*

Abstract
While XAI focuses on providing AI explanations to humans, can the reverse – humans explaining their judgments to AI – foster richer, synergistic human-AI systems? This paper explores various forms of human inputs to AI and examines how human explanations can guide machine learning models toward automated judgments and explanations that align more closely with human concepts.

Keywords
human-AI interaction, explainable AI, synergistic human-AI systems, user interface, artificial intelligence, design, adaptive interfaces, user experience

1. Introduction

It is now well accepted that it is important for AI systems to offer explanations to users [1]; indeed this is embedded in the EU GDPR legislation [2]. This paper poses the converse question, "what if users offered explanations of their decisions to an AI system".

One of the reasons for transparency and XAI (explainable AI) is to surface potential bias as first noted by one of the authors in the early 1990s [3]. This is equally important for humans, especially when their input is being used to train an AI system, which might otherwise embody the trainer's prejudices.

However, human explanations offer other advantages, allowing machine learning (ML) that is more aligned with human reasoning, and can itself be explained in more human-like ways.

In the rest of this paper, we will first look more broadly at the ways humans currently engage with AI systems in terms of the kinds of inputs they provide (Section 2). We then look at ways in which human explanations might be provided at the user interface level (Section 3). This will create a series of examples of different kinds of user explanations, which raises algorithmic (Section 4), architectural (Section 5), and even social challenges (Section 6).

Once one thinks in terms of human explanations, it is possible to see elements of this in previous work, not least classic knowledge elicitation techniques [4]. However, it appears to be a largely unexplored area, and we hope to encourage others to explore it with us.

2. Human inputs to AI

With the exception of chatbots, humans and AI typically take very different kinds of roles during an interaction. Figure 1 show some common ways in which they communicate. During learning a human expert may provide labeled training examples, which are used to create a model such as a neural network, decision tree or rule set. This model is then used in practice when the user makes some sort of query, perhaps presenting an unseen data item to classify, whereupon the AI system provides a
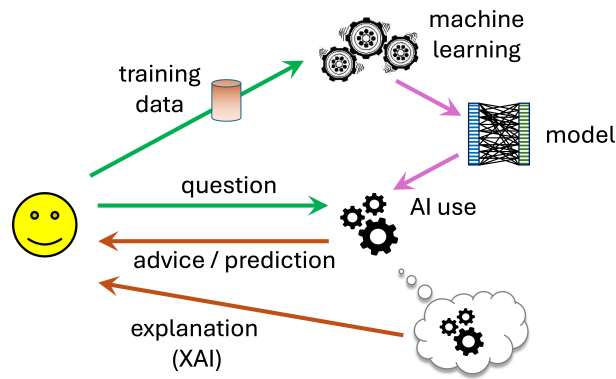
Figure 1: Common communications between human user and AI.

predicted classification or other form of advice or answer. Often also an XAI system will create some form of explanation, that to some extent 'explains' the AI system's response. Most commonly this is a *local explanation* that explains the classification or decision about a particular data item and those close to it, rather than a *global explanation* of the model as a whole.

If we focus on the human inputs, we can identify a number of common ways that knowledge is passed to the AI:

example data – This may be part of a batch exercise during training, or may be more interactive. For example, in Query-by-Browsing (QbB; see Fig. 2), the user looks through a database listing selecting some records of interest and rejecting others; after a while the system infers an SQL query [5]. However, if this query is not correct, the user can simply provide more positive or negative examples of data records, leading to a revised system SQL query.

labeling – Training data is often labeled, but users may also be explicitly asked to label data items for which the algorithm has low confidence, or to help verify learnt rules. In particular during interactive learning [6, 7, 8], the expert presents training data, but the system may challenge these with counterfactuals, leading to a form of 'conversation' where each utterance is a labeled data item.

relevance/quality feedback – Often users are asked to give some form of explicit star rating or thumbs up, which is used to incrementally improve relevance models. This can be thought of as a form of post-hoc labeling.

implicit sensing – The training data and/or labeling may be captured implicitly from one or more humans, for example using physical sensors for activity recognition, click-through data for search optimisation, or purchase/viewing history for recommender systems. This form of data gathering can be made more effective through careful user interface design, such as epistemic interaction [9].

As well as these common forms of human information provision, there are some less common ones:

confirming/rejecting system rules – In some variants of Query-by-Browsing when the system generates a rule, the user can reject it and then the system would search for an alternative rule.

editing/amending system rules – When the model used by the AI is a transparent set of rules, it may be possible for the user to tinker with them. One variant of QbB used a genetic algorithm to generate conjunctive rules displayed in a query-by-example-style tableau [10]. In this case the user could accept a rule, but also edit some of the parts of it, for example, changing the threshold in a 'greater than' clause. Similarly, the paper that first introduced the term 'interactive machine learning', described a system where the user and machine learning system worked collaboratively to create a decision tree [6].
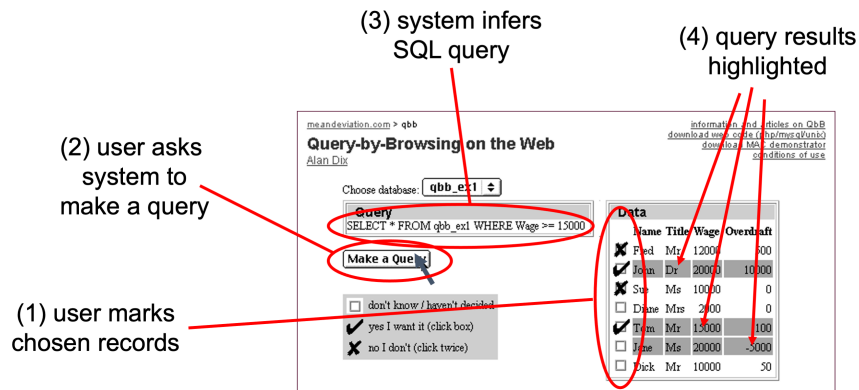
Figure 2: Query-by-Browsing – how it works [5].

confirmation of explanations – When the model is more obscure, local or global explanations may be given. In these cases the user may accept, reject or rate an explanation. This does not mean not so much that the explanation is not correct in the sense of matching the model's behaviour (one presumes that the XAI engine ensures this), but more whether it is comprehensible or useful. In the case of an expert this might be saying "I agree with the model's output, but this is not the right reason for it", for the novice end user more "yes that was helpful". This can be used as a form of relevance/quality feedback for the XAI engine.

editing/amending explanations – FRANK [11] uses a 'skeptical learning' approach [12], which allows the user to reject or modify an explanation. Similarly [13] ask a human annotator to critique a LIME-style explanation and identify "the components that have been wrongly identified by the explanation as relevant".

"*Explanations can be extracted using feature-based methods. At this point, the annotator supplies a label for the instance and, optionally, corrective feedback on the explanation. This means that the user can, for instance, indicate what input variables the machine is wrongly relying on for making its prediction.*" [12, p.17]

Note how the ability to edit rules and explanations can be seen as a form of *equal opportunity* [14], the design principle that suggests that when possible rather than having system output and user input separately, each partner in the dialogue should have access. For example, the system might be able to add defaults/suggestions in user inputs fields. Similarly, where possible, the user can be allowed to edit output values, with this updating appropriate internal data, or at very least be able to reuse the output values.

However, one can apply this principle a little more deeply, looking at the kinds of ways the user and AI system present information and take input from each other and ask if there are parallels the other way round. Figure 3 shows two ways in which this might extend the idea of who does what:

AI as critic – If the user can approve or edit system results and explanations, could the AI system also be able to take on the role of critic? In fact, existing systems can be seen in this light. Skeptical learning does ask the user to think again if the user's classifications appear to contradict previous training examples [11], this can be seen as an AI parallel of the relevance feedback or confirmation of system outputs. At a richer level, DebunkBot, a chatbot that provides evidence to challenge misinformation, has been found to reduce belief in conspiracy theories [15].

user explanations – If systems are asked to produce explanations of their reasoning, what if users could do the same? For example, when an expert labels a candidate's CV as being a good fit for an academic job, they could say it is because, "has_PhD is Yes and number_of_publications is large". Such information could be used both to guide ML systems to create models that correspond
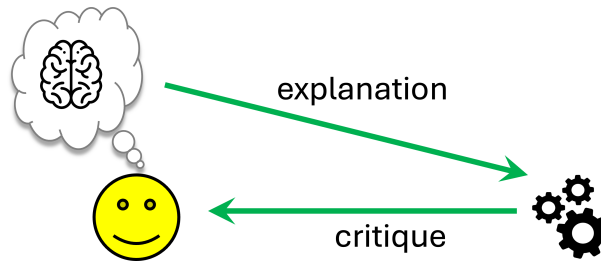
Figure 3: Some less common communication types between human user and AI.

more closely to the human's decision making and also to provide a vocabulary whereby the system can create better explanations of its own operation.

Both of these open up interesting research questions, but in this paper we will focus on the latter.

3. Human explanations

We commonly offer explanations of our own views or decisions when interacting with other humans. Just as with XAI, this may include global explanations, offering some sort of overarching rule, or more often a local explanation, justifying a specific decision. Explanations of this sort will, by their nature, be given by someone with a level of domain expertise; however the degree of expertise may vary. In some cases the explanation may therefore be treated as ground truth, whereas in others more advisory.

In many cases these explanations may be incomplete, and typically only operate for 'examples like this one'. For example, during a job selection process, one might be asked, "why this candidate"; a possible answer could be "because they have a PhD and high publication rate". However, on a different occasion the answer might be "because of extensive industrial experience and large grant income". We understand that these are local criteria, although one could also be asked to justify the apparent contradiction, or think about difficult cases where the two would lead to different outcomes.

In this section we will look at how such human explanations might be included in different kinds of user interactions with AI.

3.1. Feature constraints in QbB

In Query-by-Browsing [5] the user's interaction is to simply add or remove examples, as shown in Figure 2. The user is saying "this is a wanted/unwanted row", but not why.

One could easily add the ability to have a degree of global 'explanation'. Figure 4 shows how this might be done; as well as selecting the wanted/unwanted rows, the user also selects one or more columns. This is saying, "these are the rows I want and these are the columns I'm using to make my judgment". Arguably this is not so much an explanation as simply specifying to the algorithm which input features to use.

It would also be possible for the user to specify columns for a specific row (see Fig. 5, above), which would be mean, "I chose this row because of this column value". This feels more like a true explanation.

The user could be more precise specifying what it is about the feature that made it useful to make the discrimination (see Fig. 5, below). This could be simply giving the general direction, 'salary is large', or an exact rule such as, 'salary greater than 10,000'.

These user explanations can be regarded as constraints on the algorithm, saying that when the system makes a decision about this case, the specified features or rules should be part of the process that gave rise to it. This should mean that the system's decision model is more closely aligned with that of the user.

It is easy to only provide specified features in the training data for a machine learning algorithm – that is, making use of the global 'explanation'. However, these local explanations may require more radical modifications to ML algorithms.

Figure 4: QbB – global feature selection.

Figure 5: QbB – local feature explanations: (above) select feature(s); (below) specify local rule.

An ML research challenge is to tune the ML so that the user-provided criteria are guaranteed (or encouraged to be) part of the ML model. For decision trees or rules determined by a genetic algorithm, this may be a fairly straightforward change to the fitness function For DNNs this might be encouraged by adding the criteria explicitly into pinch-point layers [16], or creating clouds of generated data around the key examples classified by the criteria.

3.2. Counterfactuals in skeptical learning

The Frank system employs techniques described as 'skeptical learning' [11]. The user provides training data, but the system challenges the user when new examples appear to contradict previous ones, effectively saying "you've just said X is class C, but you said Y, which is very similar, is is class D". This forces the user to think again about the example. As a result of this the user might retract, relabel or revise the new example or a previous one.

Could the user offer explanations, potentially local explanations (like local XAI) as part of this interaction?

1. User submits new example X – which the system determines potentially contradictory
2. System says "are you sure?"
3. User says "yes"
4. System says "but what about Y" and presents counterfactual that is very close to X, but with different class/diagnosis
5. User says "but in X property A is much larger than in Y"
6. Maybe further rounds of interaction where the system finds another example Z where A is larger, eliciting further local explanations about a property B, etc ...

Assuming the user does not retract X as part of this process, the system now needs to relearn including X, but also has properties A, B, ... that can and should be part of the local decision around X.

Note that, while the underlying ML algorithms are different, these explanations are like those suggested for Query-by-Browsing. However, there the explanations were about a single item, whereas here the user explanation was in response to a counterfactual and is effectively making a distinction X is different from Y because of A.

This raises further algorithmic challenges. It would be possible to treat the user explanation as two single item explanations: "X because of A" and "Y because of not A", but this overlooks key aspects of a differential explanation, not least that shared features between X and Y would not be mentioned.

3.3. Feeding-back importance measures

While Query-by-Browsing demonstrates the effectiveness of binary feature selection in database query contexts, other domains and situations may benefit from a more fine-grained approach to feature importance. Modern explainable AI techniques, particularly SHAP (SHapley Additive exPlanations) [17], provide visualizations that show the relative contribution of different features to a model's prediction. This same paradigm could be reversed, allowing humans to indicate the relative importance of features in their own decision-making process.

For example, when classifying a medical diagnosis, rather than indicating a binary set of relevant symptoms, an expert could specify that elevated white blood cell count had a strong positive influence (0.8 on a normalized scale), while fever had a moderate positive influence (0.4), and patient history had a weak positive influence (0.2). This richer feedback more accurately captures the nuanced way humans weigh evidence in their decision-making process.

Such continuous importance measures could be captured through various interface mechanisms:

- Direct manipulation of SHAP-style bar charts, where users can adjust the length and direction of bars representing feature importance;
- Interactive slider controls for each relevant feature, allowing precise specification of importance values;
- Visual annotation tools for marking regions of importance in image data, with intensity of markup indicating degree of importance;
- Relative ranking interfaces with automatic normalization to maintain consistent scales.

This approach provides a natural vocabulary for dialogue between human and AI systems, as both can express their reasoning in terms of feature importance distributions that can be directly compared and analyzed. It is particularly valuable in domains where feature interactions are complex and the relative importance of different factors varies significantly between cases.

The challenge then becomes how to incorporate these weighted importance measures into the learning process — both for improving the model's predictions and for generating explanations that better align with human reasoning patterns. This alignment is particularly crucial in cases where the model and human may arrive at the same conclusion but through different reasoning paths, as indicated by divergent feature importance distributions. By enabling this richer form of feedback, we can work towards AI systems that not only learn from human decisions, but also from the reasoning process behind those decisions.

3.4. User generated intermediate features

A user's explanation may include high-level terms or features that are not part of the input feature set, but derived from it. For example, a doctor might say that an important aspect of their diagnosis is an "irregular heart rhythm", and label some data with this additional high-level feature, which is mid-way between the input data and final diagnostic classification. In some cases, the user may be able to identify which parts of the raw data contributed to this (e.g. an ECG trace).

Note we are referring here to derived features that the expert uses when seeing the same data as the AI. This is different from times when the training expert introduces new vocabulary at the data capture stage, for example as part of a physical examination of a patient a doctor might say, "the patient has a gray pallor". In the latter case the new vocabulary is an additional input space feature.

3.5. Free text – Chatbots and LLMs

While structured feedback mechanisms such as feature importance measures provide valuable input to AI systems, the emergence of Large Language Models (LLMs) opens up possibilities for more natural, free-text explanations from users. Modern LLMs often employ Reinforcement Learning from Human Feedback (RLHF) techniques [18], where they present multiple alternative responses and ask users to select the most appropriate one. While traditional RLHF typically only captures the binary or scalar preference between responses, this interaction pattern can be extended to capture not just which response users prefer, but also *why* they prefer it.

For example, when an LLM presents two alternative explanations for a scientific concept, a teacher might select one and explain: "This explanation uses more appropriate vocabulary for my middle school students" or "This version better connects to previous concepts we've covered in class." Such free-text rationales provide rich contextual information that goes beyond simple preference selection. In particular, they may include the user explaining their critique of the system's explanation as well as the system decision.

This approach represents a significant departure from the methods discussed previously. Rather than working through structured interfaces, users can provide feedback in natural language, introducing new concepts and criteria not previously encoded in the system. The explanations can capture complex, context-dependent reasoning that might be difficult to express through feature importance or other structured mechanisms. Consider a medical professional explaining their preference for one diagnostic explanation over another: their rationale might encompass subtle interactions between patient history, current symptoms, and treatment context that would be challenging to capture through more structured feedback methods.

The primary challenge lies in effectively incorporating these free-text explanations into the system's memory and learning process. Future research could explore various approaches to extract structured knowledge from these explanations, identifying recurring themes, preferences, and reasoning patterns. This structured knowledge could then inform the system's response generation more systematically, potentially creating a feedback loop where human explanations continuously refine the model's understanding of different contexts and user needs.

This capability is particularly valuable in educational and advisory contexts, where the same information might need to be presented differently depending on the audience or situation. By accumulating user explanations about why certain responses work better in specific contexts, the system can build a more nuanced understanding of how to adapt its communication style and content appropriately.

4. Algorithmic implications

In the previous section we saw types of explanations which the user might present to the system:

hard feature constraints – that say that a feature must be included in the way that a data item is classified. This may be precise $age >= 18$, imprecise "*age* is large" or unspecified "*age* must be used"

feature importance measures – saying that certain features have greater importance than others or may have a rank order of importance.

intermediate features – such as "unusual behaviour" that are used in the decision, but do not have an *a* priori definition or derivation from the features.

Figure 6: Constraining a decision tree to include explanation criteria.

free text – which may introduce new terms, concepts or even rules, but embedded in unstructured text so that even identifying them is an issue.

Each of these create new algorithmic challenges for machine learning techniques. In this section, we will look at each and suggest how they might be incorporated into specific ML approaches, but it is meant to be a starting point, not a solution.

As noted earlier, human explanations may be based on more or less expertise, and thus need to be weighed more or less strongly. In addition, some explanations concern a single data items whereas others may be counterfactual explaining different classification of two apparently similar examples. Finally, the way these are incorporated into a model might be my (re)starting machine learning from scratch, whereas others may use incremental learning, updating the existing model. We have not dealt with these issues explicitly in the potential approaches below, but they each add to the algorithmic challenges.

4.1. Incorporating feature constraints into decision trees

The examples in both Section 3.2 and 3.1 gave rise to data examples with a list of features with some sort of value constraint on each feature. these constraints may be exact (e.g. `loan<10,000`, or `has_PhD=true`), partial (e.g. `nos_phds` is large, or simply knowing that some feature should be included. In the latter cases, the exact values for the feature will have to be learnt. More abstractly, we have a set of examples, and for each example X a set of local criteria A, B, etc., for which we would like to ensure that A, B, ... are part of the decision that gives rise to X.

In the case of a conjunctive rule, it must be of the form: $A \wedge B \wedge ...$

If we are generating a decision tree it must be constrained to include A and B above the node where X is classified (see Fig. 6).

If the decision tree or rule set is generated by a GA or similar algorithms this can be added to the fitness function.

For top-down algorithms such as ID3 / C4.5 [19, 20] things are more complicated. The criteria A, B, .. can simply be given additional weight alongside entropy, in a similar way to how one might use criteria from previous queries. Alternatively one could start by forcing criterion A as the top node and B as second node of whichever branch X is classified. The disadvantage of this is that this is effectively using them as global criteria, and hence hard to extend this to multiple examples X, Y, Z, with associated criteria.

Another option is to build a tree as normal and then force the key decisions back into the tree as modifications bottom-up from the leaf node where X is classified, pruning or rebuilding sub-trees. One version of this would be to look for the closest ancestor node where there are both A and not-A examples, insert a new A criterion node just above and then rebuild the tree below.

4.2. Incorporating feature importance measures into learning

When users provide SHAP-style importance values for features influencing their decisions, these values represent a form of soft constraint on how the model should weigh different inputs. Unlike the hard constraints of decision trees where features must be present in specific positions, these importance measures suggest relative weights that should influence the learning process.

For neural networks, one approach is to modify the backpropagation algorithm itself to incorporate feature importance. The user-provided importance values can be propagated through the network alongside the regular activations, effectively modifying how strongly different parts of the network adapt to training examples. This is achieved by adjusting the learning rate during weight updates based on the propagated importance values, while ensuring that the standard learning behavior is preserved when no importance values are specified.

For gradient boosted trees, the splitting criteria could be modified to favor splits on features that users have identified as important. This is similar to how decision tree algorithms typically use information gain or Gini impurity, but with an additional weighting factor based on user-provided importance values. Features marked as highly important by users would receive a boost in their splitting score, making them more likely to appear earlier in the tree structure.

A key challenge in implementing these approaches is handling cases where different examples have contradictory importance values for the same features. One potential solution is to cluster examples based on their importance distributions, allowing the model to learn multiple valid approaches to the same problem. This would acknowledge that different experts might have different but equally valid reasoning patterns for reaching the same conclusions. Additionally, factoring in the level of expertise could help weigh each case, enabling the model to prioritize insights from more experienced or relevant sources.

4.3. Intermediate level features and user generated vocabulary

One way to deal with new intermediate level features algorithmically is to separately build a classifier for each new feature and then add these to the input features. However, sometimes this additional information can be embedded more uniformly within the training process. Figure 7 shows one way to achieve this in a deep neural network. An arbitrary node is chosen in a pinch-point layer, where one expects higher level concepts to be encoded. When there is no coding for this intermediate feature backprop works as usual. However, when the value is known the node is 'clamped' to the known value. The feedforward stage of backprop works as normal except the activation value for the node is stored and instead the known value is fed forward. The feedback stage again works as usual, except that at the clamped node the difference between the known value and calculated activation is used as a delta value just as if it were on the output layer and then feeds backwards as normal. The updates up and downstream of the clamped node may also be weighted more heavily.

4.4. Free text explanations

If free text user explanations emerge as part of interactions with LLMs, it may be possible to partially parse the explanation. It might include existing feature names which can then be used to generate feature constraints or importance, and new terms might be introducing *intermediate features*. The explanations could include more complex logical arguments requiring deeper analysis, for example such as "although I wouldn't normally look at foot reflexes when dealing with kidney problems, because of the patients unusual medical history ...". This parsing may itself use LLMs or other NLP techniques.
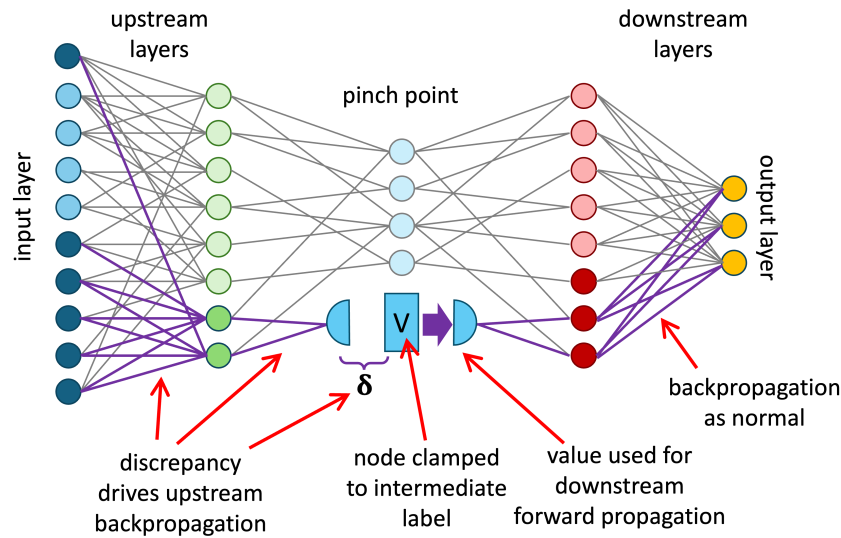
Figure 7: Clamping a deep neural net at feature-space level when semantic values known.

5. Architectural implications

Supporting human explanations as part of AI system development requires rethinking traditional machine learning architectures. While traditional systems follow a linear pipeline from training to prediction, incorporating human explanations creates a cyclic flow where user feedback can influence both immediate responses and long-term model behavior.

Some existing frameworks have begun to tackle the challenge of integrating human feedback into AI systems. For example, HumanLayer[1] provides an API for collecting human input at critical points in AI workflows. However, supporting rich explanatory feedback requires more comprehensive architectural changes.

The interactive nature of this architecture can be illustrated through the following pattern (where `clf` represents a generic classifier):

```python
# Simple one-way explanation
model_explanation = clf.explain()

# Interactive explanation with human feedback
with clf.explain() as model_explanation:
    # Initial model explanation available
    print(model_explanation)

    # Human provides explanation and gets updated model
    model_explanation = model_explanation.provide(human_explanation)
```

This interaction creates two feedback loops, as shown in Figure 8:

- A short-term loop where human critiques and explanations can trigger incremental updates to improve immediate system responses
- A long-term loop where validated explanations become part of the training data, influencing future model behavior

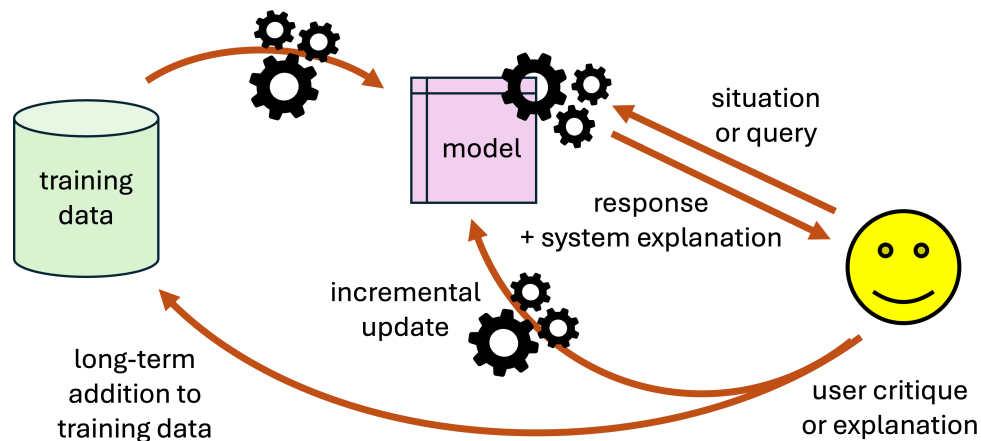Supporting these loops requires several key architectural capabilities:

---

[1]https://humanlayer.dev

Figure 8: Architecture showing the dual feedback loops created by human explanations: immediate response updates and long-term training improvements.

1. Interactive sessions that maintain context across multiple explanation exchanges
2. Version control for both models and their associated explanations
3. Mechanisms to incorporate human feedback into both immediate responses and training processes
4. Metrics to track how effectively the system learns from human explanations

More complex forms of explanation, such as free-text rationales or intermediate features, would require additional processing components while maintaining these fundamental feedback loops. Integration with existing ML infrastructure platforms like MLFlow[2] would need to support both the immediate interactive nature of explanations and their long-term incorporation into model development through appropriate versioning and artifact storage.

6. Social implications

There are social implications of this sort of development that need to be considered, too. For human feedback to be effectively incorporated (whether in training or in active learning) the context of that feedback will need to be taken into account. Is it local feedback that applies in the local context? Or is it global feedback that applies universally? Is it expert feedback or novice feedback? The former likely to include a judgement on the accuracy and appropriateness of any AI explanation, the latter likely to cover only accessibility and user satisfaction in a broad sense. Who is able to determine this context? How is reliability or improvement generally evaluated?

More concretely, the staff at a given hospital – though individually diverse – will have a greater or lesser degree of common purpose according to their local health community and their own organisation's role within it. Similarly, an educational establishment will be attuned to the needs of their own students. In each of these example contexts, there are organic, social processes by which the sharing of developed resources beyond the original organisation can take account of the suitability of one context for the development of resources for use in another. So there are levels of meta information that may or may not become embedded in the technology – dependent on not just 'what' but 'why' explanations. Grasping the meaning, significance and value of that meta information will be a new area for individuals and organisations. How will that be navigated? Who will control and curate the information?

Previous works [21, 22] have drawn attention to the distinct interests and approaches of what they call upstream and downstream parts of the development and adoption process for technology. Is it likely that commercial, upstream stakeholders will prioritise the community interests of expert and lay humans who provide inputs for improved AI decision-making? If a user or group of users provides a huge amount of feedback that improves the AI learning and tailors an AI to their school, hospital,

---

community etc, then the value of that trained AI may be great for similar groups. But if the process is owned and controlled by tech companies, then the ability to negotiate that benefit across communities of interest is lessened and might be eliminated altogether.

AI that can engage in meaningful dialogue on a specific problem may well become more persuasive. This is already a feature of LLMs where the form of the response can often appear more compelling than its content for the unwary. Bringing users closer into active dialogue may raise as many questions as it addresses.

7. Discussion and Future Work

This paper has introduced the concept of human explanations as input to AI systems, moving beyond the traditional one-way flow of AI explanations to humans. We have explored several mechanisms for capturing these explanations, from feature constraints to free-text rationales, and proposed ways to incorporate them into machine learning systems.

Our immediate plans focus on extending Query-by-Browsing to implement human explanations as described in Section 3.1. This will allow users to not just select examples, but also explain their selections through feature constraints and importance measures.

This work opens up several promising research directions:

- Developing algorithms that can effectively incorporate human explanations while maintaining model performance and generalization capabilities, particularly handling cases where explanations may conflict or require careful balance with patterns in the training data
- Creating frameworks for translating different forms of human explanation – from feature constraints to free text – into actionable guidance for machine learning systems
- Investigating how human explanations might help identify and mitigate algorithmic bias, potentially leading to more transparent and trustworthy AI systems
- Extending these approaches beyond QbB to different types of machine learning models and application domains, exploring how various AI architectures might benefit from human explanatory input

The incorporation of human explanations into AI systems represents a significant shift in how we think about human-AI interaction. Rather than treating human input merely as training data, we propose considering it as a rich source of reasoning that can help shape how AI systems learn and explain their decisions. We hope this work inspires others to explore this promising direction for creating more effective and truly synergistic human-AI partnerships.

For ongoing updates on this work see https://alandix.com/academic/papers/AXAI2025-talking-back/.

Acknowledgments

References

[1] Q. V. Liao, D. Gruen, S. Miller, Questioning the ai: Informing design practices for explainable ai user experiences, in: Proceedings of the 2020 CHI Conference on Human Factors in Computing

Systems, CHI '20, Association for Computing Machinery, New York, NY, USA, 2020, p. 1–15. URL: https://doi.org/10.1145/3313831.3376590. doi:10.1145/3313831.3376590.

[2] Council of the European Union, Position of the council on general data protection regulation, Technical Report 8 April 2016, Council of the European Union, 2016. URL: http://www.europarl.europa.eu/sed/doc/news/document/CONS_CONS(2016)05418(REV1)_EN.docx.

[3] A. Dix, Human issues in the use of pattern recognition techniques, in: R. Beale, J. Finlay (Eds.), Neural Networks and Pattern Recognition in Human Computer Interaction, Ellis Horwood, xxxx, 1992, pp. 429–451. URL: https://alandix.com/academic/papers/neuro92/.

[4] N. Shadbolt, P. Smart, Knowledge elicitation, in: J. Wilson, S. Sharples (Eds.), Evaluation of human work, 4th Edition, CRC press Boca Raton, 2015, pp. 163–200.

[5] A. Dix, A. Patrick, Query by browsing, in: Interfaces to Database Systems (IDS94) Proceedings of the Second International Workshop on Interfaces to Database Systems, Lancaster University, 13–15 July 1994, Springer, 1994, pp. 236–248.

[6] M. Ware, E. Frank, G. Holmes, M. Hall, I. H. Witten, Interactive machine learning: letting users build classifiers, International Journal of Human-Computer Studies 55 (2001) 281–292.

[7] J. A. Fails, D. R. Olsen Jr, Interactive machine learning, in: Proceedings of the 8th international conference on Intelligent user interfaces, 2003, pp. 39–45.

[8] J. J. Dudley, P. O. Kristensson, A review of user interface design for interactive machine learning, ACM Transactions on Interactive Intelligent Systems (TiiS) 8 (2018) 1–37.

[9] A. Dix, B. Wilson, M. Roach, T. Turchi, A. Malizia, Epistemic interaction–tuning interfaces to provide information for ai support, in: SYNERGY 2024 – Designing and Building Hybrid Human-AI Systems 2024. Proceedings of the 1st International Workshop on Designing and Building Hybrid Human-AI Systems co-located with 17th International Conference on Advanced Visual Interfaces (AVI 2024), 2024. CEUR Vol. 3701.

[10] A. Dix, Interactive querying-locating and discovering information, in: Second Workshop on Information Retrieval and Human Computer Interaction, Glasgow, 11th September 1998, 1998. https://www.alandix.com/academic/papers/IQ98/.

[11] F. Mazzoni, R. Guidotti, A. Malizia, A frank system for co-evolutionary hybrid decision-making, in: International Symposium on Intelligent Data Analysis, Springer, 2024, pp. 236–248.

[12] A. Monreale, S. Teso, Cognition-aware explanations for hml, TANGO Deliverable D2.1, 2024.

[13] S. Teso, K. Kersting, Explanatory interactive machine learning, in: Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, 2019, pp. 239–245.

[14] C. Runciman, H. Thimbleby, Equal opportunity interactive systems, Int. J. Man-Mach. Stud. 25 (1986) 439–451. URL: https://doi.org/10.1016/S0020-7373(86)80070-0. doi:10.1016/S0020-7373(86)80070-0.

[15] T. H. Costello, G. Pennycook, D. G. Rand, Durably reducing conspiracy beliefs through dialogues with ai, Science 385 (2024) eadq1814.

[16] A. Wüst, W. Stammer, Q. Delfosse, D. S. Dhami, K. Kersting, Pix2Code: Learning to Compose Neural Visual Concepts as Programs, in: Proceedings of the Fortieth Conference on Uncertainty in Artificial Intelligence, PMLR, 2024, pp. 3829–3852.

[17] S. M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, Curran Associates Inc., Red Hook, NY, USA, 2017, p. 4768–4777.

[18] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, R. Lowe, Training language models to follow instructions with human feedback, 2022. URL: https://arxiv.org/abs/2203.02155. arXiv:2203.02155.

[19] J. R. Quinlan, Induction of decision trees, Machine learning 1 (1986) 81–106.

[20] J. R. Quinlan, C4. 5: programs for machine learning, Elsevier, 2014.

[21] T. Greenhalgh, J. Wherton, C. Papoutsi, J. Lynch, G. Hughes, C. A'Court, S. Hinder, N. Fahy, R. Procter, S. Shaw, Beyond Adoption: A New Framework for Theorizing and Evaluating Nonadoption, Abandonment, and Challenges to the Scale-Up, Spread, and Sustainability of Health and Care

Technologies, Journal of Medical Internet Research 19 (2017) e367. doi:`10.2196/jmir.8775`.

[22] P. Lehoux, F. A. Miller, G. Daudelin, J.-L. Denis, Providing Value to New Health Technology: The Early Contribution of Entrepreneurs, Investors, and Regulatory Agencies, International journal of health policy and management 6 (2017) 509–518. doi:`10.15171/ijhpm.2017.11`.