# Explainable Inference in the FRANK Query Answering System

# Explainable Inference in the FRANK Query Answering System

**Kwabena Nuamah** [1] and **Alan Bundy** [2]

**Abstract.** The demand for insights into how artificial intelligent systems work is rapidly growing. This has arisen as AI systems are being integrated into almost every aspect of our lives from finance to health, security and our social lives. Current techniques for generating explanations focus on explaining opaque algorithms such as neural network models. However, considering the fact that these models do not work in isolation, but are combined, either manually or automatically, with other inference operations, local explanations of individual components are simply not enough to give the user adequate insights into how an intelligent system works. It is not unusual for a system made up of fairly intuitive components to become opaque when it is combined with others to build an intelligent agent.

In this paper we argue that there is the need to combine diverse forms of reasoning in order to generate explanations that span the entire chain of reasoning: not just explanations for the, so called, black-box models. Our hypothesis is that:

> *A hybrid approach using statistical and deductive reasoning makes possible a richer form of explanation not available to purely statistical ML approaches.*

We explore the concepts of 'local' and 'global' explanations and show how to give users a wide range of insights, using what we term an '*explanation blanket*'. We tackle this challenge using the FRANK query answering system and show that its hybrid approach facilitates this kind of reasoning with explanations. It is important to note that the evaluation of user preferences for explanation is outside the scope of this work.

## 1 INTRODUCTION

In this work, we show that FRANK's (Functional Reasoner for Acquiring New Knowledge) compositional architecture and recursive inference algorithm [20, 18, 7] facilitates the generation of explanations of its answers. We formalise the new idea of an *explanation blanket* in §3.2 and show how FRANK's inference makes possible this novel way of providing insights into non-trivial inferences.

As AI becomes an integral part of our lives, the demand to understand the outputs of these systems continues to grow. One common area of application of AI systems is query answering. Many techniques have been explored for QA including search, information retrieval and more recently deep neural networks. However, the current focus in explainable artificial intelligence (XAI) tends to be limited to providing explanations of only the core inference mechanism, e.g., a neural network model. This is only a small piece of the puzzle that needs to be unravelled for users. These methods for explaining answers to users ignore several crucial aspects of the inference process including data pre-processing, knowledge source selection and uncertainty, to name a few.

Consider the question "*What will be the population of Europe in 2022?*". An answer to this question has to be predicted from a regression on past populations of Europe, or a sum of predicted populations of countries in Europe in 2022, or by applying another non-trivial calculation. To automatically infer this answer such that it is explainable to a user, a combination of deductive and inductive reasoning must be applied. Explaining just the core regression model gives only a partial insight into how the answer was arrived at. Other steps in the inference, such as which data points were selected and from which sources, are also vital for understanding the answer. Also, such prediction tasks and the composition of answers from different pieces of information from diverse sources introduce uncertainties into answers which must be made explicit to users.

End-to-end differentiable systems used in Deep Neural Networks (DNNs) tend to be opaque and offer very little insights into their decisions, even for the simplest problems. Further, these systems lack intuitive means to explain their decision given the internal representation of their statistical inference mechanism.

Our work is motivated by the need to improve the intelligibility of the inference process, work around the opaqueness of some statistical methods and reduce the complexity of inference (§3.1). In this paper, we argue that the concept of "explainability" should incorporate: (1) knowledge source selection, (2) feature selection of inputs and their pre-processing, (3) easy-to-explore inference outputs and (4) inference uncertainty. We also argue that a hybrid system, such as FRANK, that combines both statistical and deductive reasoning, lends itself to explainable decisions. We show that explanation blankets, which provide explanations at different levels of detail, provide users with insight into the reasoning behind the decisions. Finally, we evaluate Frank using a test set based on questions about country development indicators from the World Bank.

## 2 BACKGROUND

### 2.1 FRANK

The FRANK query answering (QA) system [20, 18, 7], combines deductive and statistical inference methods to infer new knowledge using information retrieved from knowledge sources on the Internet. It infers both its best estimate of the answer and an estimate of its uncertainty. In the case of numeric answers, for which a probability would not be meaningful, the uncertainty estimate is instead an error bar.

[1] University of Edinburgh, United Kingdom, email: knuamah@ed.ac.uk
[2] University of Edinburgh, United Kingdom, email: a.bundy@ed.ac.uk

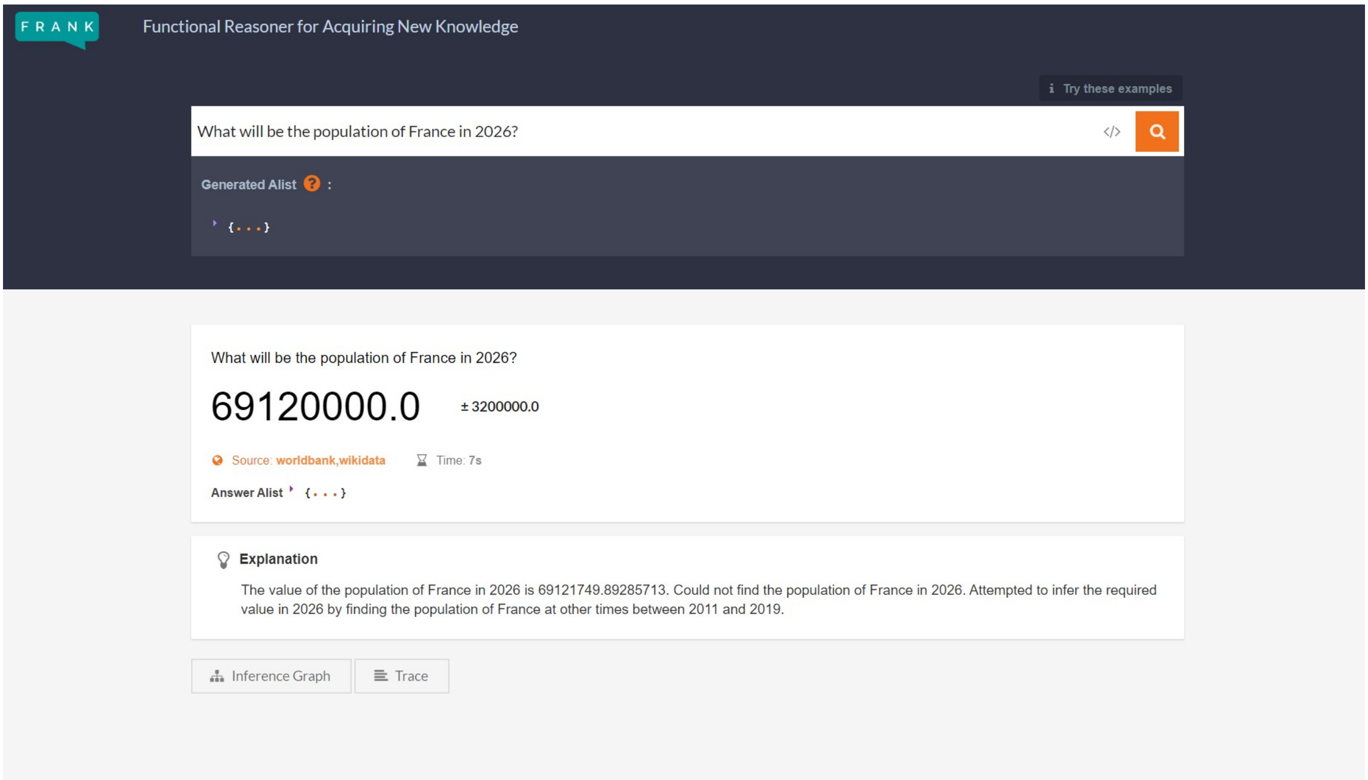


**Figure 1.** FRANK's UI for the query "What will be the population of France in 2026?"

FRANK uses a flexible attribute-value pair data structure known as an association list, abbreviated as *'alist'*. Alists enable the representation of queries, data and intermediate states of the problem uniformly during inference. Attributes include the standard triple found in RDF [11], i.e., subject ($s$), property ($p$), object ($o$), as well as other attributes required by FRANK e.g., time ($t$), inference function ($h$), uncertainty ($u$) and operation variable ($v$). For example, the question: *"What will be the population of Europe in 2022?"* is represented by the alist: $\{\langle \texttt{h,VALUE}\rangle,\langle \texttt{v,?x}\rangle,\langle \texttt{s,Europe}\rangle,\langle \texttt{p,population}\rangle,\langle \texttt{o,?x}\rangle, \langle \texttt{t,2022}\rangle\}$, where $?x$ is both the value of the object $o$ and the value to be returned $v$.

FRANK answers the question by constructing an *inference graph* on the fly which recursively decomposes the query alist using decomposition rules until the variables at the leaves of the inference graph are instantiated. Once a variable is instantiated, the inference function ($h$), usually a reduce operation such as *regression, average, sum, maximum, minimum,* etc., in the parent of the instantiated alist, attempts to aggregate the values returned by its children. This *up-propagation* happens recursively until the operation variable in the root node is instantiated. Thus, an inference graph can represent an intermediate state of inference, where variables are not yet instantiated, or the final inference for an answer, once all the necessary variables are instantiated and propagated to the root alist. Decomposition operations include: (1) temporal, which decomposes by data/time attributes; (2) geospatial, which decomposes by place or location attributes; (3) normalisation, which decomposes nested queries into simple ones and (4) comparison, which decomposes to determine items to be compared.

**Definition 1 (Inference Graph)** *An inference graph is an acyclic graph with two kinds of nodes.*

$z$ **nodes** *are OR nodes, each labelled with an alist representing a sub-goal to be solved. Its children are $h$ nodes representing each of the alternative inference operations that could be used to solve this sub-goal.*

$h$ **nodes** *are AND nodes each labelled with a inference operation for solving their parent's sub-goal. Their children are $z$ nodes representing the sub-goals that this inference operation needs to solve.*

**Leaf nodes** *are $z$ nodes that can be solved by direct look up in a knowledge source.*

*A proof is a sub-graph of an inference graph in which all non-leaf $z$ nodes have only one child, i.e., there is only AND branching.*

*Figure 3 depicts an inference graph.*

Our reference implementation of FRANK has a web-browser user interface (UI) supported by a back-end reasoning service. The UI provides two forms of output: a trace of the inference process and a graphical display of all or part of the inference graph. Figures 1 and 2 show an example of a query answered using FRANK. The purpose of this paper is to describe the different forms of explanations generated from FRANK's reasoning processes and the additions to the UI to assist with a user's insight into answers presented.

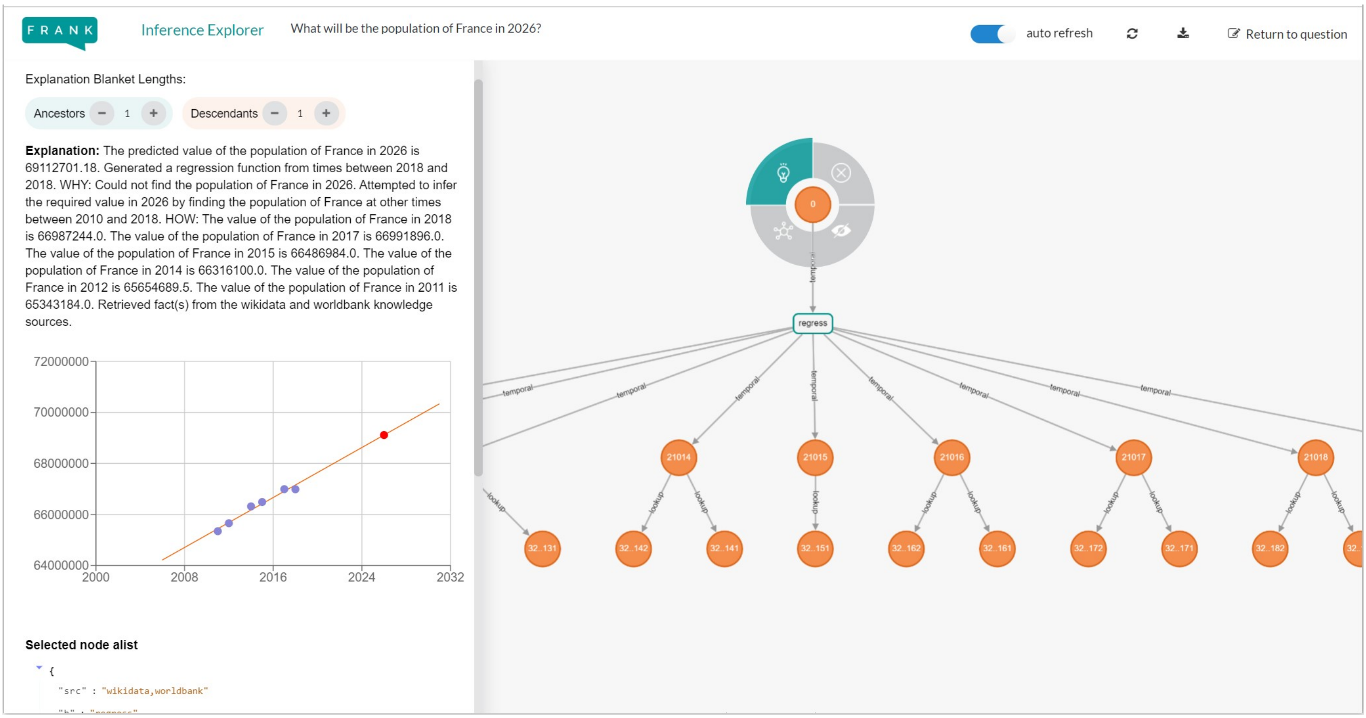


**Figure 2.** FRANK's **Inference Explorer** showing the inference graph for the question "What will be the population of France in 2026?". Node 101 is selected and its explanation and regression graph displayed in the sidebar. Context menu for the nodes in the inference graph allows users to hide or show branches.

## 2.2 Explainable AI (XAI)

A critical aspect of intelligence is the ability to explain one's decision to others. As much as it important to make decisions or predictions in an automated way, it is also necessary to be able to convey the rationale behind the decision to others for several reasons. These include the desire to: (i) verify decisions, (ii) integrate into larger decision processes, (iii) improve the system, (iv) comply with legislation and (v) allow the subjects of the decisions to appeal them. These social aspects of AI systems are necessary in order for intelligent systems to coexist with humans and provide the needed support to them.

Many organisations consider explainable decisions by AI systems as a key factor in the success and wide adoption of AI beyond academia and information technology companies. DARPA, an advocate for XAI [12], has challenge problems in data analytics and autonomy where an AI system learns a model to solve a task, generates an *explainable model*, and presents the explanations to the users through an *explanation interface*.

While the field of XAI is now gaining prominence, there is long history of work in developing explanations for intelligent systems. Several of these studies are captured in surveys of the field such as [3] and [17]. These earlier studies underline the fact that explanations are important for users interacting with complex, intelligent systems. Early work such as [23] and [24] highlight the fact that an ability to explain decisions is vital to the acceptance of intelligent systems. Other studies including [22] and [4] stress the importance of explanations in helping users check the accuracy of predictions, thereby increasing their confidence in the outputs of the machine learning system or providing the basis for appeal against them.

Explanations in an AI system can be local or global [17]. Local explanation provide insights into specific aspects of the AI system for specific cases. For instance, in a question such as "What will be the population of Europe in 2022?", a local explanation can focus on the prediction component. At the moment, most attempts at explanation in AI systems follow this approach. However such localized explanations are not sufficient to give an adequate understanding of the answer returned. A global explanation provides a better mental model of the system to the user [15, 21]. Global explanations give insights into how the whole AI system works. In the above example, such an explanation will include details about the data selected, the reasons for decomposing the problem in a particular way, the reason for selecting the prediction algorithm used, etc. This gives users a better mental model of the system and provides them with sufficient insights that they can use to solve similar problems.

There are diverse viewpoints for explanations in intelligent systems. For example, explanation as the search for answers to why, how and what-if questions using causal chains, goal-plan-actions hierarchies and justifications [9] and explanation as a mechanism for discovery [16].

## 2.3 Compositional AI Architectures

Compositionality refers to a the property that an intelligent system exhibits by generating solutions to inference problems from an automatic composition of modules that individually solve different parts of the problem. This property plays an important role in generating explanations from complex systems since the interactions between the components of the system, if carefully designed, can provide significant insights into the workings of an AI system.

In QA, the nature of the inference pipeline used can be recursive or non-recursive. Recursive inference algorithms dynamically and recursively decompose problems into sub-goals and propagate answers to sub-goals up the inference graph. Non-recursive ones generate an entire program that combines different operations and then executes it in a single shot to solve the problem. Many of the approaches to QA tend to be non-recursive. For instance, Dependency-based compositional semantics (DCS) [14], Neural Module Networks [1, 2] and The Neural Symbolic Machines [13] work on the assumption that the knowledge base (KB) from which they look for facts contains all the knowledge needed to solve the problem. Hence, they fail if all the facts they need are not pre-stored in the KB.

In FRANK we make no assumptions about the existence of facts in KBs. *Alists* are recursively decomposed into sub-goals and then aggregated when instantiated to solve the question. This recursive algorithm also means that FRANK does not commit to a particular strategy to solve a question and so can choose different strategies given, for instance, the data available. A recursive algorithm like FRANK also allows additional information, such as uncertainty from its inference components and data, to be propagated through the inference.

# 3 XAI in FRANK

## 3.1 Motivating factors

The inference graph and trace generated by FRANK naturally provide insight into its reasoning. However, by themselves, they are not very accessible to users since the trace can span several hundreds of lines and the inference graph grows exponentially.

The sections that follow explore the use of English text based on FRANK's inference graph to explain FRANK's reasoning augmented with prediction graphs summarising intermediate data aggregation and predictions. This is a contribution to the burgeoning research area of XAI. The factors driving XAI in FRANK are:

- *Intelligibility*: Just the program traces and inference graphs that FRANK previously displayed are suitable as debugging tools for its developers, but would not be understood by lay users.
- *Inherent Opaqueness*: Whereas deductive reasoning can be explained by a chain of reasoning (i.e., an inference graph), machine learning methods, such as a deep neural network, are inherently opaque.
- *Complexity*: Inference trees can be very large, so user can be overwhelmed by their sheer size.

Our aim is to make every node in FRANK's inference graph explainable based on the alist or decomposition operator that labels it and its neighbours.

## 3.2 Explanation Blanket

FRANK's UI is interactive and so selecting the root node of the inference graph provides an explanation for the answer FRANK returns. However, for deeper inference graphs the explanation will only be a high-level one. It is therefore necessary to generate explanations for both root and non-root nodes if we're to give users insights into FRANK's reasoning. We use the idea of an *explanation blanket* to achieve this.

An explanation blanket is a sub-tree of the inference graph from which a meaningful, non-trivial explanation can be generated for the alist that labels a node. An explanation describes (1) *what* FRANK is doing at a given step of its reasoning, (2) *why* it needs to do this and (3) *how* it is achieving this. We map these to the explanation blanket as follows:

1. *what:* the reduce operation performed by the node $n$ on its children;
2. *why:* the map operation (decomposition) of the parent node;
3. *how:* the manner in which the children are combined. e.g. relevant details of algorithm used and the children selected for reduction.

For example, FRANK generates this explanation for its answer to the query:

> ***What is the country in Africa with the lowest population in 2010?***
> *We found entities that have type 'country' and are located in Africa and calculated the minimum of the population in 2010 of the entities. Inferred country is Seychelles*[3].

### *3.2.1 Definitions*

Once a proof has been found within the inference graph, the values at the leaf nodes are propagated up through the proof to the root node to provide the answer to the original query. Propagation is defined recursively as follows.

**Definition 2 (Propagation of Values)**

**Base Case:** *The values of leaf nodes are provided by look-up in a knowledge source.*

**Step Case:** *Let $x$ be a z-node which is decomposed by the inference operation $\Delta$ labelling its z-node(s) child(ren). Let $\Delta$ have $m$ z-node children and assume recursively that they return the values $v_1, \ldots, v_m$, respectively. Let $h$ be the inference function of $x$. Then the value returned by $x$ is $h(v_1, \ldots, v_m)$.*

Uncertainty estimates are also propagated up the proof, from leaves to root, in a similar way. Uncertainty is not the subject of this paper, but more details can be found in [19].

An explanation blanket is a sub-graph of the inference graph that has the following $\delta$-complete property.

**Definition 3 ($\delta$-complete sub-graph)** *A $\delta$-complete sub-graph is a sub-graph of the inference graph that has a z-node as its root and z-nodes at its leaves.*

Explanation blankets can be centred on either *z-node*s or *h-node*s. To generate an explanation for node $n$, a blanket must be centred on $n$ with its length (height and depth) specified

**Definition 4 (Size of an Explanation Blanket)**

**The height,** *$L_H$, of an explanation blanket is the length of the path from the central node to its highest ancestor.*

**The depth,** *$L_D$, of an explanation blanket is the length of the path from the central node to each of its lowest descendants.*

*Either $L_H$ or $L_D$ can be 0. In fact, an explanation blanket centred on the root node must have $L_H = 0$ and one centred on a leaf must have $L_D = 0$. Note that, in order to satisfy the $\delta$-complete property, a z-centred explanation blanket must have even numbered height and depth and an $h$ centred one must have odd numbered height and depth.*

*Figure 3 depicts both a $z$ and an $h$ explanation blanket.*

[3] FRANK did not receive the territory of Saint Helena, Ascension and Tristan da Cunha as a country from the KBs it searched.

| OPERATIONS | TEMPLATES |
|---|---|
| **DECOMPOSITIONS** | |
| **`temporal`** | Could not find the ⟨`property`⟩ of ⟨`subject`⟩ [in ⟨`time`⟩]. Attempted to infer the required value in ⟨`time`⟩ by finding the ⟨`property`⟩ of ⟨`subject`⟩ at other times between ⟨`time`$_{min}$⟩ `and` ⟨`time`$_{max}$⟩. |
| **`geospatial`** | Could not find the ⟨`property`⟩ of ⟨`subject`⟩ [in ⟨`time`⟩]. Finding the ⟨`property`⟩ [in ⟨`time`⟩] for the constituent parts of ⟨`subject`⟩: ⟨`entities from sub-query`⟩. |
| **`normalize`** | Had to solve the sub-queries before determining the ⟨`property`⟩ [in ⟨`time`⟩]. |
| **`comparison`** | Need to solve the sub-queries to determine the items to compare. |
| **FAILED REDUCTIONS** | |
| **`eq`**, **`gt`**, **`gte`**, **`lt`**, **`lte`** | Could not compare the values since the values of all items being compared are not known. |
| **`comp`** | Failed to solve the sub-problem. |
| **`value`**, **`values`** | Failed to find the ⟨`operation`⟩ of ⟨`property`⟩ [in ⟨`year`⟩]. |
| **`regress`** and others (**`min`**, **`max`**, **`sum`**, **`avg`**, etc.) | Failed to calculate the ⟨`operation`⟩ value of ⟨`property`⟩ [in ⟨`year`⟩]. |
| **SUCCESSFUL REDUCTIONS** | |
| fact retrieval | Retrieved fact(s) from the ⟨`kb`$_1, \ldots, kb_n$⟩ knowledge base(s). |
| **`eq`**, **`gt`**, **`gte`**, **`lt`**, **`lte`** | Inferred value is ⟨`inferred`⟩. Did a comparison to determine if ⟨`child`$_1$⟩ is ⟨`operation`⟩ ⟨`child`$_2$⟩. |
| **`comp`** | Solved the sub-query and found the following ⟨`type`⟩ values: ⟨`child projections`⟩ |
| **`regress`** | Generated a regression function from times between ⟨`time`$_{min}$⟩ and ⟨`time`$_{max}$⟩. |
| others (for alist with aggregate only) | The ⟨`operation`⟩ value of the ⟨`property`⟩ [in ⟨`year`⟩] of ⟨`subject`⟩ is ⟨`aggregate value`⟩. |
| others (for alist with both aggregate and projected variable) | The entity whose ⟨`property`⟩ [in ⟨`year`⟩] has the ⟨`operation`⟩ value of ⟨`aggregate value`⟩ is ⟨`projected entity`⟩. |

**Table 1.** Templates for generating explanations for various decomposition operations and inference operations for reducing alists.

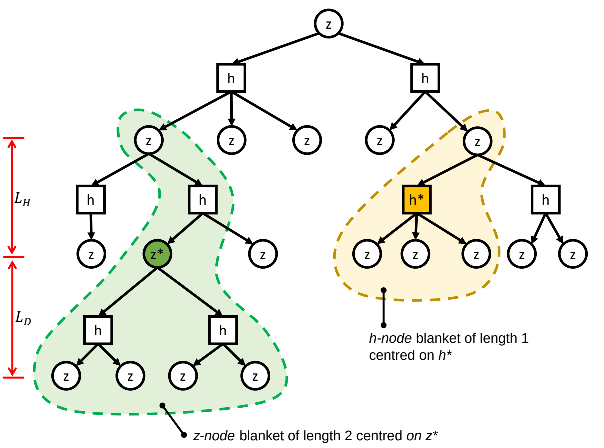


**Figure 3.** A *z-node* blanket of length 2 ($L_H = 2$ and $L_H = 2$) and a *h-node* explanation blanket of length 1.

## 3.3 Adjusting the Richness of Explanations

We can increase the richness of the explanation of an alist or inference operation by increasing the height and/or depth of the explanation blanket of the selected node. By enlarging the explanation blanket, we can generate a more detailed account of the decompositions involved in an inference step, as well as richer account of reductions of descendants that lead to the answer inferred at the node being explained. The default height and depth in our current implementation is 1. The user can change this in the Inference Explorer UI (see 2).

## 3.4 Evaluation

Recall that our hypothesis was:

*A hybrid approach using statistical and deductive reasoning makes possible a richer form of explanation not available to purely statistical ML approaches.*

To evaluate this hypothesis, Frank was run on a test set of queries. The results are shown in Table 2. The columns show: the query asked; the answer returned; the height and depth of the explanation blankets set; and the explanation given for each setting. Heights and depths were set to be equal in these tests, but needn't have been. Note that larger explanation blanket settings generated more detailed explanations for the same query. The test set was chosen to demonstrate: both quantitative and qualitative queries; both direct look-up and inferred answers; the diversity of the knowledge sources consulted; predictions; and nested queries.

Figures 1 and 2 are screenshots of FRANK's GUI when some of these queries are answered. Together they illustrate: the answers generated and its uncertainty; the depiction of explanation blankets and the UI to change their size; the graphical depiction of regression used in prediction; explanation in English; and the knowledge sources consulted.

We claim that this evidence supports our hypothesis. More discussion about this claim can be found in §5

## 3.5 Templates

The English explanations in Table 2 were generated by instantiating templates associated with the inference operations and the explanation blanket currently chosen.

These templates are generic and not specific to any particular domain. They depend only on the inference operations used and not on the domain. They have been successfully tested on a range of different domains. We are continuously introducing new question types and their corresponding templates.

The templates are listed in Table 1, organised by the different inference operations currently available to FRANK.

### *3.5.1 Geospatial decomposition template*

*Geospatial decomposition* turns a query over a composite entity, e.g., a continent, into a set of sub-queries by asking the same query for

| Question | Answer | EB size. | Explanation |
|---|---|---|---|
| Q1. What was the gdp of Ghana in 1998? | 7481000000 | 1 | The value of the gdp of Ghana in 1998 is 7480968858. Retrieved fact(s) from the World Bank and Wikidata knowledge sources. |
| Q2. What will be the population of France in 2026? | 69250000. | 1 | The value of the population of France in 2026 is 69247799. HOW: Generated a regression function from times between 2010 and 2018.The predicted value of the population of France in 2026 is 69247799. Retrieved fact(s) from the World Bank and Wikidata knowledge sources.. WHY: Could not find the population of France in 2026. Attempted to infer the required value in 2026 by finding the population of France at other times between 2010 and 2018 |
| Q3. Country in Africa with the lowest urban population in 2010? | Seychelles | 1 | The entity whose urban population in 2010 has the minimum value of 47880.0 is Seychelles. HOW: Retrieved fact(s) from the World Bank and Wikidata knowledge sources. Had to solve the sub-queries before determining the urban population in 2010. |
| | | 2 | The entity whose urban population in 2010 has the minimum value of 47880.0 is Seychelles. HOW: The country values found for the sub-query include: Botswana, Algeria, Cameroon, Angola, Benin, Burundi, Cape Verde, etc. Retrieved fact(s) from the World Bank and Wikidata knowledge sources. Had to solve the sub-queries before determining the urban population in 2010. |
| Q4. Country in Asia with the highest energy consumption in 2012? | | 2 | The entity whose energy consumption in 2010 has the maximum value of 61093297.293516 is People's Republic of China. HOW: The country values found for the sub-query include: Israel, Laos, Malaysia, Jordan, Bahrain, Japan, etc. Retrieved fact(s) from the World Bank and Wikidata knowledge sources. Had to solve the sub-queries before determining the energy consumption in 2010. |
| Q5. What will be the population in 2025 of the country in Europe with the highest gdp in 2010? | 8176000 | 3 | The value of the population of Germany in 2025 is 85632039.0. HOW: The entity whose gdp in 2010 has the maximum value of 3417094562648.95 is Germany. The entity whose gdp in 2010 has the maximum value of 3417094562648.95 is Germany. The value of the population of Germany in 2025 is 85632039. Generated a regression function from times between 2010 and 2018.The predicted value of the population of Germany in 2025 is 85632039.0. Retrieved fact(s) from the World Bank and Wikidata knowledge sources. Had to solve the sub-queries before determining the population in 2025. |
| Q6. Will the population of Ghana in 2022 be greater than the population of Belgium in 2030? | True | 1 | Inferred value is 'true'. Did a comparison to determine if 31890697 is greater than 12224774. WHY: Need to solve the sub-queries to determine the items to compare. HOW: An input value for operation is 31890697.399999857. An input value for operation is 12224774.890931368. Retrieved fact(s) from the World Bank and Wikidata knowledge sources. |
| | | 4 | Inferred value is 'true'. Did a comparison to determine if 31890697 is greater than 12214670. WHY: Need to solve the sub-queries to determine the items to compare. HOW: The value of the population of Ghana in 2022 is 31890697. Generated a regression function from times between 2010 and 2018. The predicted value of the population of Ghana in 2022 is 31890697. An input value for operation is 12214670. The value of the population of Belgium in 2030 is 12214670. Generated a regression function from times between 2010 and 2018. The predicted value of the population of Belgium in 2030 is 12214670. Retrieved fact(s) from the World Bank and Wikidata knowledge sources. |

**Table 2.** Examples of queries, answers returned and explanations of various blanket lengths. Answers to questions are displayed to 4 significant figures, while the full figures are shown in the explanations. It is also worth noting that these answers are based on real-time data present the in the Wikidata, World Bank and Geonames knowledge bases at the time of they were queried.

each of its constituents, e.g., countries. This template describes these constituents. For example, to predict the African country that will have the highest population in 2021, first Africa is broken into a set of its constituent countries: $\{Algeria, \ldots, Zimbabwe\}$, then the population prediction is asked of each constituent of this set. The country with the maximum population is then returned as the answer.

#### 3.5.2 Normalisation template

This template describes the need to first solve sub-queries and their intermediate results. For queries containing a sub-query where the type of constituents are specified, the query alist is normalised by creating a sub-query (child alist) that has a **`comp`** (set comprehension) operation for aggregation. Entity types are extracted from the description of the filters in the sub-query. For example, to predict which country in Africa will have the largest population in 2021, FRANK first creates a sub-query to find all the countries in Africa, which is returned as a set using **`comp`**. Next, the population in 2021 of each member of this set is predicted by a temporal decomposition and the member with the maximum predication is returned.

#### 3.5.3 Temporal decomposition template

*Temporal decomposition* turns a query about a time for which the answer is not yet known into a set of similar queries for times for which the answer is known. This template describes the new times for which child alists are created and the subsequent regression operation that is performed. For example, to predict a population at some point in the future, FRANK first finds the population for previous census dates. Regression is then used to turn the set of time/answer pairs into a function. This function is then extrapolated (or interpolated) to estimate an answer for the original time.

#### 3.5.4 Comparison decomposition template

FRANK supports queries that compare one value to another. Question 6 in Table 2 is an example of such a query. The comparison decomposition first solves the sub-queries while ensuring the instantiation of the pair of operation variables to be compared before the up-propagation. The template describes the values being compared and the outcome of the comparison.

#### *3.5.5 Aggregation templates*

During the aggregation phase, templates are used to generate additional explanations of the aggregations performed. The **`regress`** (regression) and **`comp`** (set comprehension) operations are explained with the temporal and normalisation decompositions.

At the leaf nodes, we describe the sources from which the variables are instantiated. For example: *"Retrieved facts from the world bank and geonames knowledge sources*.

## 4 ADDITIONAL SOURCES OF INSIGHT

### 4.1 Graphical Explanations

Graphical explanations can be used to support the textual explanations generated for the alists. It helps users to visually understand parts of the inference process especially when the underlying aggregation operations are dense and difficult to understand textually. A typical example of this is regression. For problems that include regression as an inference step, FRANK returns not only the predicted value, but also the prediction function (e.g. the coefficients in the linear equation). Again, using the explanation blanket, we plot (1) a prediction graph using the regression function calculated in the regress node; (2) the underlying data points from the child nodes; and (3) the predicted value from the value propagated to the parent node. Figure 2 shows an example of the resulting graphical plot in the reference implementation of FRANK. Such prediction graphs provide insights into the intermediate inference steps in FRANK as it infers answer.

### 4.2 Interactive Inference Graph

The recursive inference algorithm constructs an inference graph through the sequence of alist decompositions, with $z$ nodes labelled by alists and $h$ nodes labelled by inference operations. In our implementation of FRANK, the entire inference graph is presented to the user in real-time via the Inference Explorer view shown in Figure 2. Each node has a context menu that allows a user to view the details of the node including its alist as well as an explanation for a specified explanation blanket length. The menu also allows a user to choose how much of the inference graph to view by collapsing or expanding sub-graphs. This is in the spirit of the second author's earlier work on *proof planning* and *hiproofs* [6, 8]. Visually, this makes the entire inference transparent and enables users to decide on which parts of the inference to focus.

## 5 DISCUSSION

In this section we show how our claims about our work are confirmed. Examples of questions, answers and explanations generated by FRANK are shown in Table 2.

Firstly, our algorithm for FRANK naturally lends itself to a combination of deductive reasoning for developing the inference graph and statistical/arithmetic operations for combining instantiated alists in the graph.

Although FRANK's integration of explanation into its inference mechanism is novel, our underlying viewpoints of explanations is motivated by earlier work. First is the view in [9] that explanation is the search for answers to why, how and what-if questions using causal chains, goal-plan-actions hierarchies and justifications. We leave the 'what-if' question for future work, and replace it with a 'what' question. In the context of QA, our use of explanation blankets achieves this view of explanations in a manner that scales from very local explanations to global ones as the blanket size is increased.

Richer explanation does not only mean adding more detailed text to support an answer. Rather, it is about providing different viewpoints as well as using different modalities to support an answer. In some cases, an explanation blanket of depth 1 on the root node, similar to the explanation in Figure 1, suffices. But in other cases such as question Q3: "*What will be the population in 2025 of the country in Europe with the highest gdp in 2025*", where the answer is obtained from a deep inference graph, a blanket of depth 1 on the root node is restrictive as it presents only a high-level explanation to a user. By allowing a user to interactively centre explanation blankets on any node in the inference graph, FRANK gives the user diverse viewpoints of explanations. The interactive inference graph makes the entire QA process transparent to the user. Using this to generate explanations based on specified blanket lengths provides further insights to the user.

Furthermore, our work presents insights into FRANK's inference through different modalities. These include textual explanations, as well as the interactive inference graph. Additionally, for a node labelled by a temporal decomposition, a regression graph is presented to the user to support the prediction at that node.

Finally, the prediction examples in Table 2 show that even with opaque statistical methods such as neural networks for regression (e.g. Q6), we are able to give a user meaningful insights into answers and how they are found or inferred. This is only possible because our approach to explanations focuses not only on the key operation such as regression, but on all operations leading to an answer including variable instantiations with facts from KBs. Combining this with regression graphs allows FRANK to give users adequate insights to justify its answers.

## 6 RELATED WORK

FRANK's algorithm facilitates explanations that are better than other data-intensive QA systems such as [10, 5], similar QA systems such as Wolfram|Alpha[4] and, more recently, search engines that attempt to return exact answers instead of results lists (e.g. search via Google Knowledge Graph[5]). FRANK returns not only an answer, but its entire inference graph. Given the full graph of decompositions and aggregations, we are able to elicit simple English explanations of computations that led to the answer. Even in the case where FRANK fails to find an answer, an interrogation of the inference graph can give the user some insights into why FRANK failed. For instance, in Figure 2, the grey coloured nodes show the nodes whose alists FRANK was unable to instantiate.

Our algorithm for FRANK also provides insights into the answers by looking not only at the primary inference algorithm, e.g. regression, but at other inference operations, such as knowledge sources from which variables in alists were instantiated, and the uncertainties of the various facts retrieved and how they propagate to the answer. While this happens in a single hop for those current QA systems that we are aware of, which answer questions at web-scale, FRANK can provide such insights at all levels of decomposition in the inference graph.

[4] https://www.wolframalpha.com
[5] https://developers.google.com/knowledge-graph

FRANK's UI provides a simple query-answer page that is traditionally found for most QA systems. Additionally, FRANK provides an "Inference Explorer" view that exposes FRANK's entire inference graph to the user in an interactive UI. This gives a user access to both the detailed level instantiations of variables and the high level inferences that are applied to give an answer. A user can view the content of alists as well as generate explanations based on their chosen explanation blankets. The examples in Table 2 show how a user can obtain both local and global explanations for a node by changing the explanation blanket lengths.

# 7 CONCLUSION AND FUTURE WORK

This work explored the underlying architecture of a QA system to facilitate or inhibit the generation of intuitive explanations for the answers it infers. Subsequent work will focus on how users engage with the QA system, the kinds of interactions they have with the system and how they use the explanations generated by the system to inform subsequent actions and confidence in its answers. More details about the inference operations will also be added to the template.

We plan to enable users to explore 'what-if' scenarios. For instance, we would allow users to change the value being returned by either a terminal or non-terminal node in the inference graph and then discover what effect this has on both the inference process and the answer it returns. For instance, suppose FRANK has predicted the population of the UK in 2025 on the basis of previous census data, but that the user would now like to explore the possible effect of Brexit, suspecting that this might cause an exodus of non-UK EU citizens. FRANK could be used to both predict the population in the upcoming 2021 census and estimate what proportion of non-UK EU citizens it contains. The user could then create a terminal node for the 2021 results and provide a value for the population which takes into account the anticipated exodus. The 2025 population prediction could then be re-run.

We will also enhance a user's insights into FRANK's reasoning by allowing users to questions the inference itself. By enabling the persistence of not only the answer, but of the entire inference graph, we could allow users to ask follow-up questions about the answer returned by FRANK. This is especially important in cases where a graphical UI is not available, such as a smart speaker and in devices that provide accessibility to the visually impaired.

# ACKNOWLEDGEMENTS

The research reported in this paper was funded by Huawei Technologies Co., Ltd under project HO2017050001B8.

# REFERENCES

[1] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein, 'Learning to compose neural networks for question answering', in *Proceedings of NAACL-HLT*, pp. 1545–1554, (2016).

[2] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein, 'Neural module networks', *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, **2016-Decem**(Figure 1), 39–48, (2016).

[3] Or Biran and Courtenay Cotton, 'Explanation and Justification in Machine Learning: A Survey', *IJCAI Workshop on Explainable AI (XAI)*, (August), 8–14, (2017).

[4] Or Biran and Kathleen R McKeown, 'Human-centric justification of machine learning predictions.', in *IJCAI*, pp. 1461–1467, (2017).

[5] Eric Brill, Jimmy Lin, Michele Banko, Susan Dumais, Andrew Ng, et al., 'Data-intensive question answering', in *TREC*, volume 56, p. 90, (2001).

[6] A. Bundy. A science of reasoning, 1991.

[7] Alan Bundy, Kwabena Nuamah, and Christopher Lucas, 'Automated reasoning in the age of the internet', in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11110 LNAI, pp. 3–18. Springer Verlag, (2018).

[8] E. Denney, J. Power, and K. Tourlas, 'Hiproofs: A hierarchical notion of proof tree', *Electronic Notes in Theoretical Computer Science*, **155**, 341–359, (May 2006).

[9] Arthur C Graesser, William Baggett, and Kent Williams, 'Question-driven explanatory reasoning', *Applied Cognitive Psychology*, **10**(7), 17–31, (1996).

[10] Konrad Höffner and Jens Lehmann, 'Towards question answering on statistical linked data', *Proceedings of the 10th International Conference on Semantic Systems*, 61–64, (2014).

[11] Ora Lassila, Ralph R Swick, et al., 'Resource description framework (rdf) model and syntax specification', (1998).

[12] John Launchbury, 'A darpa perspective on artificial intelligence', *YouTube video,(February 15, 2017). https://www.youtube.com/watch?v=-O01G3tSYpU*, (2017).

[13] Chen Liang, Jonathan Berant, Quoc Le, Kenneth D Forbus, and Ni Lao, 'Neural symbolic machines: Learning semantic parsers on freebase with weak supervision', in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 23–33, (2017).

[14] Percy Liang, Michael I Jordan, and Dan Klein, 'Learning dependency-based compositional semantics', *Computational Linguistics*, **39**(2), 389–446, (2013).

[15] Zachary C Lipton, 'The mythos of model interpretability', *Communications of the ACM*, **61**(10), 36–43, (2018).

[16] Tania Lombrozo, 'The instrumental value of explanations', *Philosophy Compass*, **6**(8), 539–551, (2011).

[17] Shane T. Mueller, Robert R. Hoffman, William Clancey, Abigail Emrey, and Gary Klein, 'Explanation in Human-AI Systems: A Literature Meta-Review, Synopsis of Key Ideas and Publications, and Bibliography for Explainable AI', (February 2019), (2019).

[18] Kwabena Nuamah, 'Functional inferences over heterogeneous data'. Unpublished Ph.D. Dissertation, University of Edinburgh, 2018.

[19] Kwabena Nuamah and Alan Bundy, 'Calculating error bars on inferences from web data', in *Proceedings of SAI Intelligent Systems Conference*, pp. 618–640. Springer, (2018).

[20] Kwabena Nuamah, Alan Bundy, and Christopher Lucas, 'Functional inferences over heterogeneous data', in *International Conference on Web Reasoning and Rule Systems*, pp. 159–166. Springer, (2016).

[21] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin, 'Why should i trust you?: Explaining the predictions of any classifier', in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144. ACM, (2016).

[22] Panagiotis Symeonidis, Alexandros Nanopoulos, and Yannis Manolopoulos, 'Moviexplain: a recommender system with explanations.', *RecSys*, **9**, 317–320, (2009).

[23] Randy L Teach and Edward H Shortliffe, 'An analysis of physician attitudes regarding computer-based clinical consultation systems', *Computers and Biomedical Research*, **14**(6), 542–558, (1981).

[24] L Richard Ye and Paul E Johnson, 'The impact of explanation facilities on user acceptance of expert systems advice', *Mis Quarterly*, 157–172, (1995).