A Collaborative Platform for Identifying Context-Specific Values

Demonstration Track

Enrico Liscio
E.Liscio@tudelft.nl
Delft University of Technology, the Netherlands

Michiel van der Meer
m.t.van.der.meer@liacs.leidenuniv.nl
Leiden University, the Netherlands

Catholijn M. Jonker
C.M.Jonker@tudelft.nl
Delft University of Technology, the Netherlands

Pradeep K. Murukannaiah
P.K.Murukannaiah@tudelft.nl
Delft University of Technology, the Netherlands

ABSTRACT
Value alignment is a crucial aspect of ethical multiagent systems. An important step toward value alignment is identifying values specific to an application context. However, identifying context-specific values is complex and cognitively demanding. To support this process, we develop a methodology and a collaborative web platform that employs AI techniques. We describe this platform, highlighting its intuitive design and implementation.

KEYWORDS
Values; Ethics; Context; Natural Language Processing

1 INTRODUCTION
Values are abstract motivations that guide our opinions and actions [15]. Engineering value-sensitive agents that learn and align their actions with human values is essential for robust and beneficial artificial intelligence (AI) [3, 10, 11, 14, 17]. Then, an important question is: what values should an agent learn and align with?

Several lists of basic values, that transcend cultures and contexts, have been described in the literature [4, 6, 15]. However, a growing number of researchers emphasize that values must be situated within an application context for concrete analysis, e.g., to reason about conflicting values [1, 12], align values and norms [16], or evaluate value adherence of an agent-based system [18].

We define context-specific values as values "applicable and defined specifically within a context" [9]. The following scenario illustrates why context-specific values are important for an agent. Consider a personal travel agent. Schwartz values [15] of security and hedonism are relevant to the agent’s reasoning but the value of power is arguably not. Further, to ensure security, the agent aims at increasing travel safety. However, travel safety takes different meanings in different contexts: during a pandemic, it is safer to travel by car to avoid larger crowds; otherwise, traveling by public transportation is preferable to reduce the likelihood of accidents.

2 AXIES PLATFORM
Identifying the values relevant to a context is challenging. To simplify this task, Axies employs AI techniques to guide a small group of annotators through an opinion corpus composed of value-laden textual opinions about a context. Axies promotes inductive reasoning by asking annotators to annotate values based on the opinions. A value is described by its name, keywords (words that help binding the value to the context) and defining goal (which describes what holding a value in the context means). The result of Axies is a value list relevant to the authors of the opinions in the examined context.

The Axies methodology is composed of two phases: an individual value annotation phase and a collaborative consolidation phase. Axies has two key features: (1) it requires a small number of researchers to perform the individual value annotation task, and (2) it employs natural language processing (NLP) and active learning techniques to guide annotation.

Axies is a hybrid methodology in that human annotators are supported by AI in the process of value identification. However, Axies annotators (e.g., citizens and policy makers) may not have AI expertise. Further, Axies requires collaboration among the annotators. Thus, a computational platform is necessary to support the annotators in applying Axies without exposing them to the underlying technical mechanisms. To enable these features, we develop an intuitive and reusable web platform1, with an AI back-end, on which human annotators can collaborate. In this paper, we describe the design and implementation of this platform.

2.1 Implementation Details
The platform is implemented in Python on the Flask micro web framework [7]. The back end is also implemented in Python to provide seamless integration with state-of-the-art NLP models. All data is stored in a SQLite database [8]. Further, we developed functionalities to import the opinion corpus in a csv or yaml format. Finally, the responsive web interface is implemented in JavaScript. The interface can be used on small (e.g., smart phone) and large screens, and it utilizes the de facto standards in modern web applications.

1Demonstration: https://youtu.be/s7nJPr2Z80w
The modular setup of the two phases enables easy extension to new annotation tasks. The source code is available on GitHub\(^2\).

### 2.2 User Navigation

Annotators are required to register with a username and a password. Operations can be performed asynchronously. Data is stored to the SQL database upon input, allowing the annotators to leave and return to the platform without losing progress.

A top navigation bar is accessible from any page (as shown in Figure 1), permitting users to switch between the two phases of Axies (Explore and Consolidate) and different contexts (e.g., COVID and ENERGY in the case of the experiments in [9]).

### 2.3 Exploration

During the exploration phase, annotators individually generate a value list based on the opinions in the corpus. However, opinion corpora may be too large to be analysed by an individual. Axies aims at exposing the annotators to a subset of the corpus while increasing the coverage of read opinions. Active learning and NLP techniques support the exploration phase by controlling the order in which annotators are presented with the opinions in the corpus.

The web platform reduces information overload by presenting one opinion at a time for annotation as shown at the top of Figure 1. Annotators are asked to annotate values and keywords based on the shown opinion. The interface allows them to add and delete values and keywords at any moment.

To select an opinion for annotation, first, all opinions are encoded to a vector space through the sentence embedding Sentence-BERT model [13]. Distributed Dictionary Representation [5] allows encoding values to the same embedding space. Then, the Farthest First Traversal [2] algorithm selects the next opinion to be annotated as the farthest in the embedding space from the values already annotated and the opinions already shown to the annotators.

The progress plot (on the right of Figure 1) contains a bar per each opinion shown to an annotator, where the color indicates the actions (or lack thereof) performed upon reading the opinion. This intuitive visualization assists annotators in keeping track of their progress and deciding when saturation is reached. Finally, each value is associated with a button to fetch opinions similar to the value in order to refine individual value concepts.

### 2.4 Consolidation

During the consolidation phase, annotators are invited to combine their individual value lists. While exploration promotes divergent thinking, consolidation promotes convergent thinking. To simplify consolidation, Axies creates the union of all individual value lists and guides the annotators in methodically refining it. To facilitate this process, annotators are sequentially presented with just a pair of values at a time. Axies selects the pair as the most similar values in the vector space, assuming them as the most likely to be merged.

For each value in the pair, the annotators can fetch the opinions that led to the value annotation during exploration. If the annotators deem the two values to be conceptually identical, they may merge them by using the interface offered by the platform. Alternatively, the may edit the values in the pair and the whole value list at any moment. Upon consolidation of the value pair, annotators may fetch the following pair suggested by Axies, or decide to manually fetch the next pair from the value list. As in the exploration phase, a progress plot helps the annotators in tracking their progress. Finally, when consolidation of the list is terminated, annotators are asked to add a defining goal to each value.

### 3 CONCLUSION

We present the Axies platform, which simplifies the complex value identification task as a guided value annotation task. Our platform successfully supported the experiments involving two contexts and two groups of annotators [9] by providing an intuitive design that allows the annotators to visualize all components. The experiments show that Axies yields values that are context-specific, comprehensible to laypeople and consistent across different annotators.

Based on the feedback received by the participants in our experiments, we identify three main directions for future work. First, developing techniques to visualize values by highlighting their similarities and differences can help annotators in generating more comprehensive value lists. Second, during consolidation annotators often examined the proposed value pairs without taking actions, and sometimes resorted to selecting value pairs manually. The consolidation phase would benefit of an improved value pair selection, e.g., by normalizing the impact of keywords when computing value embeddings. Finally, as value lists emerge for multiple contexts, we call for maintaining an open-access repository of values and associated contexts. Such a repository would enable researchers in studying connections among value lists, and designers and developers in choosing values suitable for their applications.
REFERENCES


