Declaration

I, the undersigned, declare that this work has not previously been submitted as an exercise for a degree at this, or any other University, and that unless otherwise stated, is my own work.

Daniel Doherty

August 16, 2024

Permission to Lend and/or Copy

I, the undersigned, agree that Trinity College Library may lend or copy this thesis upon request.

Daniel Doherty

August 16, 2024

AIRO-Bias

Extending the AI Risk Ontology (AIRO) with a documentation framework for bias risk management in AI, aligned with international standards and the EU AI Act

Daniel Doherty

A thesis presented in partial fulfilment of the degree
MSc in Computer Science (Intelligent Systems)

Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

School of Computer Science & Statistics
Trinity College Dublin
Ireland
October 7, 2024

AIRO-Bias

Extending the AI Risk Ontology (AIRO) with a documentation framework for bias risk management in AI, aligned with international standards and the EU AI Act

Daniel Doherty

Abstract

With recent advancements in the capabilities of machine learning models in particular, there has been increasing concern regarding the safety of artificial intelligence (AI) systems. The European Union's AI Act proposes a risk-based approach to managing AI systems, considering harmful impacts which such systems may have on citizens, including to their fundamental rights, from the EU Charter of Fundamental Rights (CFR). The importance of the AI Act has led to the development of tools to help AI providers to comply with the requirements set forth by the AI Act. One such tool is the AI Risk Ontology (AIRO), an ontology which utilises state of the art Semantic Web technologies to represent AI systems, their components, and other risk-related information in a structured manner. In this report, a novel, comprehensive framework for representing documentation relevant for bias risk management is proposed, expanding upon AIRO. This novel framework includes a taxonomy for bias types which is based on a core derived from the ISO/IEC 24027:2020 document, and further expanded upon with terms from the AI Ontology (AIO) and societal bias categories adapted from Article 9 of the General Data Protection Regulation (GDPR) (additionally including gender bias). Novel properties and classes are provided to facilitate mapping from bias categories onto potential EU fundamental rights violations. The framework presented also allows for bias tests to be documented in a machine-readable format, extending the Evaluation and Report Language (EARL). Managing bias risks is also aided through the development of a machine-readable representation of datasheets, utilising the Data Quality Vocabulary (DQV) and Data Catalog Vocabulary (DCAT). The viability of this framework is demonstrated through successful implementation of real-world use cases, and its contribution to AI risk management aligned with aiding in EU AI Act compliance activities is demonstrated through the evaluation of competency questions using SPARQL Protocol and RDF Query Language (SPARQL).

Acknowledgements

First and foremost, I would like to thank Prof. David Lewis for supervising this project. Particularly for such a field as that explored here, AI risk management, where the landscape is constantly evolving, Prof. Lewis' expertise has proven invaluable. Our meetings were always enlightening and tremendously interesting, which provided me with great insight into the rapidly evolving state of the art in this new and exciting field. I cannot emphasise enough how appreciative I am to have had such a helpful and knowledgeable supervisor for this research project. Thank you.

I would also like to extend thanks to the friends which I made along the way. During the teaching terms, I spent a great deal of my time in the Phoenix House lab, with other students in the MSc Computer Science course, and there I forged valuable friendships. The discussions which we had and the friendships we forged have helped me to become a better student. Thank you, everyone.

Last but not least, I would like to thank my parents. None of this would have been possible without the support of my mother and father from the very beginning of this journey. They have recently retired from demanding careers, and I hope that they enjoy their retirement. Their encouragement has helped me tremendously, both during the course of my undergraduate and postgraduate studies. Thank you so much.

Contents

Chapter 1

Introduction

The field of machine learning (ML) has advanced dramatically in recent decades (Jordan & Mitchell 2015, Dengel et al. 2021). The advancements in this field have been attributed to various factors; Hu et al. (2019), for example, refer to novel algorithms, increased computational power and big data as factors in these recent advancements. With such rapid advancement in machine learning and artificial intelligence (AI) more generally, and with increasing economic interest in developing larger and more capable machine learning models and systems, it is often not clear where the ceiling of capability lies, which has become a particular concern in the public consciousness with regard to state of the art large language models (LLMs); papers published by *OpenAI* which introduced the *generative pre-trained transformer* (GPT) series of LLMs, which would lead to the famous *ChatGPT* AI chat bot platform, have indicated predictable performance scaling as the number of model parameters is increased (Radford et al. 2019, Brown et al. 2020). There has also been research which has reinforced such observations, with the observations of Kaplan et al. (2020), for example, suggesting that larger models (with a greater number of parameters) will continue to perform better, providing extra credibility to the scaling laws observed. With such growth trends in contemporary machine learning model capabilities, and when considering their application in AI systems exposed to individuals in various domains, academic researchers and individuals involved in the machine learning and AI communities have raised concerns about the ethical implications of AI development and deployment. In recent years, these concerns has prompted a notable interest in AI risk management, frequently featuring in conferences such as the *ACM Conference on Fairness, Accountability and Transparency (FAccT)*.

In AI risk management, it is important to distinguish between harms which arise from malicious intent and those harms which may manifest distinct from the intent of the system developer. Brundage (2018) discusses the former, proposing a number of recommendations which suggest that risk management activities should be conducted, involving collaboration between policymakers and technical researchers, in order to prevent the malicious use of AI. Concerning the latter, Weidinger et al. (2021) suggest that language models (LMs) should be expected to reinforce unfair discrimination and stereotypes in their training data, even exaggerating biases which exist in the training data (frequently referred to as "bias amplification"). In this case, it may not be the intention of the developer of the machine learning model for these harmful prejudices to be reinforced in their model's outputs, yet this is nonetheless a risk which could effect potentially serious consequences if such a model were to be deployed as a component of an AI system.

With regard to AI harms which are unrelated to intent, a significant concern arises, for example, in the reinforcement of harmful biases (Miron et al. 2021, Araujo et al. 2020), the manifestation of which having the potential to result in serious allocational consequences (discussed by Weidinger et al. (2021)). It is not difficult to imagine how the deployment of biased AI systems — for example, one which results in bias based on one's ethnicity — could negatively influence an individual's fundamental rights. This represents a manifestation of harm the European Union's AI Act is particularly concerned with, at least concerning dataset governance practices (Article 10(2), point (f) of the AI Act).

Despite such concerns with regard to the potential risks and harms which may manifest from AI systems, it is important to note the tremendous benefit which may be derived from the use of AI, through a wide a great variety of means. One striking example, among many, is the use of AI systems in medical research, where among myriad applications of AI / machine learning in this field, machine learning models have been developed for the detection of breast cancer (Rodriguez-Ruiz et al. 2019), saving human lives.

While much work has been done on AI risk management in the academic literature, there is still a significant gap in standardised approaches for documenting bias in AI systems in a manner which is aligned with the objective of aiding compliance practices with the European Union's AI Act and international standards. This dissertation aims to bridge this gap in state of the art research by leveraging Semantic Web technologies to implement a framework for bias documentation which is extensive and grounded in international standards, current practices, as well as state of the art research, while remaining aligned with European Union's AI Act (in particular, Articles 10, 11, 53 and the associated Annexes detailing technical documentation, Annex IV & XI). The proposed design expands upon the AI Risk Ontology (AIRO) (Golpayegani et al. 2022), an open ontology for representing AI systems and their components, as well as risks, consequences and impacts, aligned with the legislative requirements set forth in the EU AI Act and international standards from *International Organization for*

Standardization (ISO). Moreover, this work exploits recursive properties present in AIRO to illustrate how bias risks indicated in upstream models may be indicated to those modelling risks in downstream AI systems; this is particularly relevant given the emerging *foundation Model paradigm* (Schneider et al. 2024, Bommasani et al. 2022), whereby general-purpose models pre-trained on massive, general-purpose data may be further specialised for more specific tasks in downstream systems, for example, with fine-tuning practices.

In the present research, state of the art Semantic Web technologies are leveraged. The Semantic Web was first proposed by Tim Berners-Lee (Hendler et al. 2001), known as the inventor of the World Wide Web (WWW). In the Semantic Web, information is encoded in triples (subject-predicate-object, akin to a transitive verb in natural language) using the *Resource Description Framework* (RDF). This is often utilised in conjunction with a set of (entailment) rules, or queried using SQL-like languages such as SPARQL (recursive acronym, *SPARQL Protocol and RDF Query Language*). "Machine readability" is a term frequently used to describe information which is represented in RDF format, but this is somewhat of a misnomer. Machines cannot truly read or understand RDF triples in the human sense. However, when agreed upon vocabularies and ontologies are used with a set of established entailment rules, machines can process and "reason" with these triples in a manner which emulates human understanding, which has powerful implications. Semantic Web technologies hence provide means of representing information in a structured and machine-readable manner, which can be beneficial for a wide range of tasks, with AI risk management representing one such task.

Regarding legislative requirements which concern risks in AI, the European Union's *AI Act* was unanimously approved by the European Council on 2 February 2024. The AI Act is of extreme importance in the broader ethical AI space, representing a significant attempt to comprehensively regulate AI development and deployment in legislation, and it will likely have a significant impact beyond the EU, given the global influence of the EU market. The AI Act is a risk-based approach to managing AI risks through legislative means, broadly classifying AI systems based upon their risk, with four classes: unacceptable, high, limited, or minimal risk. Legislative requirements differ depending on the risk classification assigned to a particular AI system.

To conclude this section, a document guide is provided to inform further navigation of this report. The remainder of this report document is structured as follows:

1. This chapter is immediately followed by the **Motivation and research question** chapter, outlining the motivation for the presented extensions to AIRO (Golpayegani et al. 2022). The goal of the *Moti-*

vation and research question chapter is to provide required context for this research report in the form of a concrete motivation and research question specification, discussing also assumptions which surround these.

2. The Motivation and research question chapter is followed by two chapters, **Background** and **State of the art** (in that order), collectively constituting the literature review component of this report.

   In the **Background** chapter, important prerequisite knowledge is summarised and discussed, informing the reader of important considerations pertinent to risk management, formal ontology design and furthermore providing details regarding the suite of tools used as well as description of important terms required for the avoidance of ambiguity for the remainder of the report.

   In the **State of the art** chapter, research which immediately informs the requirements of the research presented in the present report is discussed and compared / contrasted, identifying notable gaps which inform the design section of this report.

3. A **Design** chapter follows the literature review. The Design chapter synthesises upon the state of the art, and utilises appropriate methodologies and considerations discussed in the Background and State of the art sections to provide a framework which addresses the motivation and research question.

4. Following the Design chapter is an **Evaluation** chapter, where the framework discussed in the Design chapter is evaluated using real-world case studies, illustrating how this approach may be applied in practice to aid in AI risk management aligned with the AI Act and evaluating how this satisfies the research question.

5. Finally, the findings are discussed with reference to their contribution to the state of the art, presenting conclusions based upon the evaluation section, including implementation challenges, areas for future work, and ethical considerations associated with this research.

Chapter 2

Motivation and research question

In this chapter, a description of the motivation for this research project is provided. This is followed by the specification of a research question and discussion of related assumptions.

2.1 Motivation

As outlined briefly in the Introduction section, the AI Risk Ontology (AIRO) (Golpayegani et al. 2022) is an open ontology, developed using Semantic Web technologies. AIRO allows for AI systems to be represented, providing classes and predicates to describe components of such systems as well as risks, impacts and consequences for this system, for a structured means of representing relevant characteristics for AI risk management. An application of AIRO lies in aiding AI risk management activities required for compliance with the European Union's AI Act, as its design is aligned with the European Union's Artificial Intelligence Act (AI Act) as well International Standards documents provided by the *International Organization for Standardization* (ISO).[1]

The motivation for this project and the associated research, as presented in this report, is to represent bias in a more semantically rich manner within the framework of AIRO, supporting more effective AI risk management to aid in compliance with the EU AI Act. Figure 2.1 displays a use-case from the paper introducing AIRO (Golpayegani et al. 2022), indicating how risks are represented in the AIRO ontology on a particular AI system — in this case, Uber's facial recognition AI system. The motivation becomes clear in consideration of a bias risk source as an instance of `RiskSource`. Specifically, this motivation emerges when considering the lack of a necessary explicit connection between the training data and the bias risk sources and risks, prompting research into legislative requirements and the potential

[1] A more detailed description of AIRO is provided in the State of the art section.

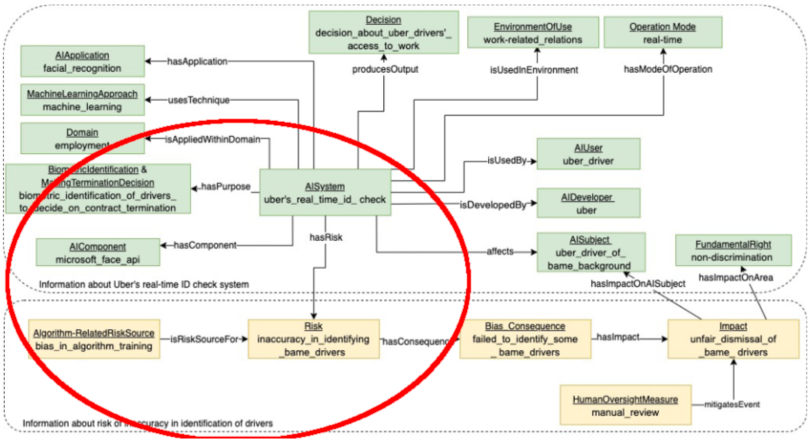


Figure 2.1: Adapted from Figure 2 of *AIRO: An Ontology for Representing AI Risks Based on the Proposed EU AI Act and ISO Risk Management Standards* (Golpayegani et al. 2022), an AIRO use-case for Uber's facial recognition system. This adaptation indicates, in particular, how risks and risk sources are represented on the AI system for this use-case. License: Creative Commons CC BY 4.0, `https://creativecommons.org/licenses/by/4.0/`

for a more taxonomically rich framework for bias as an extension to AIRO. While this representation of risk may be sufficient in many cases, it was hypothesised prior to undertaking this project that representing bias in a more structured manner than this within the framework of AIRO would be beneficial for the AI risk management tasks which AIRO is used for.

Motivation summary

In summary, the motivation of the present research is:

1. Lack of a semantically rich framework for the representation of bias testing and reporting using the AI Risk Ontology (AIRO) (Golpayegani et al. 2022).

    - Classes such as `airo:Control`, `airo:Risk` and `airo:RiskSource` are provided, but there are limited bias categories provided by AIRO for the description of bias risks. There is also no means of mapping between bias risks and EU fundamental rights violations, which may benefit efforts in AI risk management aligned with the EU AI Act.
    - No means of representing bias test results on a machine learning model (potentially containing floating or integer point results), and uncertainty regarding how this relates to the representation of bias on an AI system (`airo:AISystem`).

2. There is currently no dataset documentation approach aligned with satisfying the requirements of the AI Act directly integrated into AIRO, a broader framework for the management of risks in AI system. It was hypothesised that the integration of such a dataset documentation approach into AIRO would further aid in (bias) risk management aligned with the EU AI Act; it may, for example, guide those modelling risks in AI systems in their interpretation of bias test results, helping to gauge the implications of test results on their particular application.

2.2 Research question

In this section, the research question is described, as well as any additional assumptions surrounding this research question.

> **Research question:**
>
> **How can dataset documentation and bias test reporting be implemented in AIRO in a machine-readable manner, to align with the data governance practices outlined in Article 10 of the AI Act and aid in fulfilling the technical documentation requirements outlined in Annex IV and Annex XI of the AI Act,** and otherwise remaining aligned with international standards such as ISO, academic literature and current best practices?

Chapter 3

Background

This chapter constitutes the first component of this report's literature review. While the following section, State of the art, is a discussion and analysis of the important (recent) developments which directly inform the Design chapter, this Background chapter provides a comprehensive review of more foundational concepts and literature which the present research relies upon.

3.1 The Artificial Intelligence Act

The *Artificial Intelligence Act* (AI Act) was proposed in the European Union for the management of risks associated with the use of systems utilising artificial intelligence. In March 2024, the European Parliament adopted the AI Act and the Council later approved the AI Act in May 2024, and it will be fully applicable 24 months after entry into force[1].

The AI Act proposes a risk-oriented approach, classifying AI systems into four broad categories depending on a set of criteria. An AI system may be classified as:

1. One of **unacceptable risk**. Such systems are prohibited according to the AI Act.

2. One of **high risk**. High risk systems are regulated according to the AI Act, and much of the AI Act text addresses risk management for such systems.

3. One of **limited risk**. Such systems have less strict regulations.

4. One of **minimal risk**. Systems deemed of "minimal risk" are unregulated.

[1] Source: `https://www.europarl.europa.eu/topics/en/article/20230601STO93804/eu-ai-act-first-regulation-on-artificial-intelligence`.

3.2 Risk management & risk assessment

The present report is concerned with *AI risk management*. However, it is important that what is precisely meant by "risk management" is clear, and that important background literature is outlined also. Khlaaf (2023) note that the terms "risk management" and "risk assessment" are often conflated, with the distinction being that "risk management" should refer to a *continuous process*, with examples such as ISO/IEC DIS 42001 and ISO 9001:2015 cited, whereas risk assessments are "systematic methodologies that identify, evaluate, and report system risks", noting NIST SP 800-30 as an example (Khlaaf 2023).

However, a risk assessment should ideally be processed within a risk management framework. Whenever one is designing a risk assessment or contributing to an existing risk assessment process, considering its position within a risk management framework provides beneficial insight, serving to assist in guiding the risk assessment based upon the expected procedures at the level of the risk management framework, hence being an important consideration for this research project. Additionally, Article 9 of the AI Act calls for risk management systems to be implemented, reinforcing the importance of their implementation.

In the following section, two relevant risk management frameworks and standards are discussed: ISO 31000 and ISO/IEC 23894, published by the *International Organization for Standardization* (ISO).

3.2.1 ISO 31000 & ISO/IEC 23894

The **ISO 31000** guidelines provide a framework and process for risk management, rather general in nature. Although it is an international standard, it has been criticised. Leitch et al. (2010), for example, criticise ISO 31000 as being unclear, and claim that it would lead to illogical decisions if it were to be followed, being impossible to be in compliance with and not having any mathematical basis. Others have noted that the effectiveness of ISO 31000 as a standard heavily depends upon how it is implemented by organisations (Lalonde & Boiral 2012).

ISO/IEC 23894 expands upon ISO 31000 where appropriate with AI-specific guidance, and it is as such a highly relevant resource in the context of the present research. In ISO/IEC 23894, it is suggested that a record is kept, documenting the results of risk assessments. Three key components of such a risk assessment include risk identification, risk analysis and risk evaluation (for which a documentation on the implementation and results should be provided). The ISO standard also discusses the importance of the following:

1. The presence of a **description and identification** associated with

the system which was analysed;

2. **Intended use** of the system described;
3. **Identity** of those (whether individuals or an organisation) which conducted the risk assessment;
4. **Date of risk assessment** as well as **terms of reference**;
5. **Release status** associated with this particular risk assessment.

3.3 The Semantic Web & Open Semantic Modelling

The various tools applied in the present report were in general built with the *Semantic Web* vision in mind; that is, at a high level, allowing for content on the web to be described using common terms with agreed upon semantics. The concept of a Semantic Web was initially proposed by Hendler et al. (2001), and the tooling which has supported this vision has continued to grow[2]. However, beyond the vision of the semantic web, such tools have a place in *open semantic modelling* more generally, with applications in the semantic description of resources beyond application in the Semantic Web itself.

In this section, some of these open semantic modelling tools are described at an introductory level of detail, so as to comprehensively describe the stack of tools utilised in the present report.

3.3.1 Standards: W3C

A core consideration in the Semantic Web is interoperability, and this is aided through a consistent set of standards. The *World Wide Web Consortium* (W3C) is an example of a standards organisation which publishes different web standards, including those pertaining to open semantic modelling / the semantic web.

3.3.2 Resource Description Framework (RDF)

Resource Description Framework (RDF) allows for *tuples* to be described, indicating semantic relationships between entities[3]. Such tuples share

[2] e.g., *SHACL* was first published in 2015, while *Web Ontology Language (OWL)* was started in 2004.

[3] For a more comprehensive description (including of RDFS), see this W3C document: `https://www.w3.org/TR/rdf11-concepts/`.

notions with those of natural language, where there is a *subject*, *property* and *object* in an active sentence with a transitive verb. An example of this is shown in Listing 1. In this context, "entity" corresponds to natural language nouns, while the property corresponds to a predicate in natural language.

```
@prefix ex: <https://example.org/ex#> .

ex:Daniel ex:likes ex:chocolate .
```

Listing 1: An example of a Resource Description Framework (RDF) triple, written in TURTLE format. The prefix is indicated before the colon (`:`) character, and corresponds to an XML namespace, typically declared at the top of the file.

RDF is typically written either in an XML or TURTLE (TTL) format. For the remainder of this report, TURTLE syntax will be used for the representation of RDF (and extensions thereof), unless specified otherwise.

Each TURTLE statement is terminated using the `.` character. Another important component of TURTLE syntax is in the `,` and `;` characters, which may each be used as shorthands with distinct purposes; the `,` character indicates that the subject and predicate should be kept constant while different objects are listed, while the `;` character indicates that only the subject should be kept constant, while different predicates *and* objects are listed. The example in Listing 2 illustrates this.

3.3.3 RDFS: Classes and constraints

RDFS expands upon RDF in a number of ways, but a key extension is the enhanced representation of *classes*.

In RDF, as previously outlined, it is possible to refer to a particular object as "being" something else, as illustrated in Listing 3 (noting that `a` is equivalent to `rdf:type`).

However, RDFS extends upon this in a number of ways. One of its most significant extensions lies in its introduction of *classes* and *subclasses*, allowing for a particular instance to be considered *of a particular class*. Additionally, RDFS introduces some capability of placing constraints; using RDFS, one can define constraints on the types which may occur in the domain and range of a particular property. This is illustrated in Listing 4.

In consideration of this, one may also describe that a particular property is a *subproperty* of another property, thereby inheriting the same characteristics with respect to constraints as its superproperty.

```
# ; indicates that only ex:Daniel is kept the same,
# while new predicates and objects and used
ex:Daniel a ex:Human ;
    ex:likes ex:chocolate ;
    ex:playsInstrument ex:guitar .
# Equivalent:
ex:Daniel a ex:Human .
ex:Daniel ex:likes ex:chocolate .
ex:Daniel ex:playsInstrument ex:guitar .

# , indicates that only the object changes
ex:Daniel a ex:Human ,
    ex:footballPlayer ,
    ex:golfPlayer .
# Equivalent:
ex:Daniel a ex:Human .
ex:Daniel a ex:footballPlayer .
ex:Daniel a ex:golfPlayer .
```

Listing 2: An example of TURTLE syntax with `.` terminating statements, and shorthands using `;` and `,` characters.

```
@prefix ex: <http://www.example.com> .

ex:Daniel a ex:Person . # ’a’ is an alias for rdf:type
```

Listing 3: An indication of declaring an instance in RDF using `rdf:type`, in TURTLE syntax.

```
ex:Person a rdfs:Class .
ex:Entity a rdfs:Class .

# ex:owns may be used with ex:Person in its domain
# and ex:Entity in its range
ex:owns a rdfs:Property ;
    rdfs:domain ex:Person ;
    rdfs:range ex:Entity .

# Apply to instances.
ex:Daniel a ex:Person .
ex:Football a ex:Entity .

ex:Daniel ex:owns ex:Football . # Valid
```

Listing 4: An example of RDFS domain and range constraints, in TURTLE syntax.

3.3.4 Ontologies, Web Ontology Language (OWL)

In the present report, the *AI Risk Ontology* (AIRO) (Golpayegani et al. 2022) is expanded upon. An important prerequisite for this is understanding the concept of an ontology, and in particular, *Web Ontology Language (OWL)*, an open standard RDF-based language which may be used to describe ontologies. Using OWL, one may "represent the meaning of terms in vocabularies and the relationships between those terms" (McGuinness et al. 2004).

General

An *ontology* may be described as an "explicit specification of a conceptualization" (Gruber 1995). Gruber (1995) specifies that formal ontologies must be *designed*, involving a reasoned design process where intentional design decisions are made. Gruber (1995) details also a number of criteria which should be followed when designing an ontology: (1) **clarity**, (2) **coherence**, (3) **extendibility**, (4) **minimal encoding bias** and (5) **minimal ontological commitment**.

1. **Clarity:** Gruber (1995) uses the term "clarity" to describe ontologies appropriately with clear semantics, i.e. where the intended meaning is clearly conveyed in the terms used.

2. **Coherence:** Gruber (1995) discusses that coherence as a kind of consistency — that inferences should be consistent with the definitions.

3. **Extendibility:** in *extendibility*, Gruber (1995) describes that an ontology should be designed such that it is *monotonically* extendible, i.e. that new, specialised terms may be later added, without revising existing definitions or doing substantial restructuring of the existing ontological structure.

4. **Minimal encoding bias:** Gruber (1995) discusses that there should be minimal reliance on a particular symbol-level encoding, instead defining at the knowledge-level.

5. **Minimal ontological commitment:** Gruber (1995) describes that an ontology "should make as few claims as possible about the world being modeled", allowing for the ontology to be specialised and instantiated as desired by those parties which are committed to it.

Such criteria constitute guidelines for the development of well-formed ontologies.

Beyond those criteria outlined by Gruber (1995), the *Linked Open Terms* (LOT) methodology is an "industrial method for developing ontologies and vocabularies"[4] (Poveda-Villalón et al. 2022). The LOT methodology is segmented into four activities:

1. **Ontology requirements specification**: In the ontology requirements specification task, requirements for the ontology are specified and competency questions are specified.

2. **Ontology implementation**: In consideration of the ontological requirements specified in the previous step, this activity involves using a formal language to build the ontology. This step emphasises ontology reuse, encoding in OWL with relevant metadata and evaluation for errors.

3. **Ontology publication**: In the ontology publication step, human-readable documentation as well as machine readable files are made publicly accessible.

4. **Ontology maintenance**: In the ontology maintenance step, the ontology updated with new requirements or improvements over time, or fix errors.

[4] Linked Open Terms details can be found online at the following link: `https://lot.linkeddata.es` (verified 05.07.2024)

Web Ontology Language (OWL)

OWL allows for complex constraints to be placed on assertions made at the **ABox** level based upon relationships defined at the **TBox** level. The TBox represents statements at the terminology level, whereas ABox represents statements at the assertational level, where ABox statements must conform to those constraints specified at the TBox level.

For example, OWL defines a `disjointWith` predicate, which asserts that any instance of the subject cannot also be an instance of the object. This is illustrated in Listing 5.

```
# TBox
ex:Man a owl:Class .
ex:Woman a owl:Class .
ex:Man owl:disjointWith ex:Woman .

# ABox
ex:Daniel a ex:Man .
# It would violate the disjointWith constraint if this followed:
ex:Daniel a ex:Woman .
```

Listing 5: An example of the OWL `disjointWith` constraint, in TURTLE syntax.

3.3.5 Shapes Constraint Language (SHACL)

Shapes Constraint Language (SHACL) is a W3C Standard, and is a "language for validating RDF graphs against a set of conditions"[5]. Conditions in SHACL are in the form of "shapes", defined using RDF in a *shape graph*. On the other hand, the graph upon which shapes apply such constraints is referred to as the *data graph*.

A highly useful extension to SHACL is **SHACL-SPARQL**, which allows SHACL constraints to be constructed based upon the results of a particular *SPARQL Protocol and RDF Query Language* (SPARQL) query. **SPARQL** uses a syntax similar to SQL, allowing for knowledge bases to be effectively queried. Variables are indicated using the `?` character prefix. As an example, the query in Listing 6 would return a list with the possible subjects that could be replaced with `?name`. The use of SPARQL in conjunction with SHACL allows for powerful and intuitive constraint

[5] Source: `https://www.w3.org/TR/2017/REC-shacl-20170720/`. Verified 17/06/2024.

```
PREFIX ex: <http://www.example.org/>
SELECT ?name
WHERE
{
    ?name ex:likes ex:chocolate .
}
```

Listing 6: An example SPARQL query which would return a list of all possible subjects which satisfy the `?name ex:likes ex:chocolate` pattern.

checking. As an example, Robaldo et al. (2023) illustrate the use of SHACL SPARQL rules for compliance checking, allowing for *if-then* rules to be represented using RDF. An example of such a rule, from Robaldo et al. (2023), is illustrated in Listing 7.

```
:evaluatingProductIsProhibitedUnlessLicence rdf:type
   sh:NodeShape ;
    sh:rule [rdf:type sh:SPARQLRule ; sh:order 1 ;
        sh:prefixes[sh:declare
            [sh:prefix "rdf" ; sh:namespace
               "http://..."],
            [sh:prefix "TBox" ; sh:namespace
               "http://..."]];
        sh:construct """
            CONSTRUCT { $this rdf:type
               TBox:Prohibited . }
            WHERE { $this TBox:has-agent ?x. ?x
               rdf:type TBox:Licensee.
            $this TBox:has-theme ?p. ?p rdf:type
               TBox:Product.
            NOT EXISTS{$this rdf:type
               TBox:ExceptionArt1b}.}"""];
sh:targetClass TBox:Evaluate .
```

Listing 7: An example, from the work of Robaldo et al. (2023), of a SHACL SPARQL rule, in TURTLE syntax.

OWL also allows for the specification of constraints, as does RDFS to a more limited extent. Likewise, SHACL allows for constraints, in the form of shapes, to be evaluated against a data graph. There may be tasks for which either SHACL or OWL could be used. However, there are some important differences. SHACL has been described by some as a description logic (Bogaerts et al. 2022), and Bogaerts et al. (2022) describe two core differences between SHACL and OWL:

1. **First-order interpretation vs first-order theory**: the data graph implicitly represents a first-order *interpretation* in SHACL, whereas in OWL, the data graph represents a first-order theory (an ABox, as previously described).

2. **Differing inference task**: for OWL, the inference task is *inductive*, whereas the inference task is *deductive* for SHACL.

3.3.6 Triplestores

In order to evaluate SPARQL queries, one requires a **triplestore**, a type of *graph database*. A triplestore is akin to an SQL database, but dedicated to the storage, retrieval of RDF triples, most importantly allowing for these triples to be retrieved via semantic queries, e.g., using SPARQL. *GraphDB*[6] is an example of a triplestore.

3.4 The Foundation Model paradigm

Crucial to any consideration of bias documentation is consideration of the existing and emerging paradigms which govern how machine learning models are developed, and how AI systems in the present integrate together, but also the future trajectory in this regard. Schneider et al. (2024) discuss the emerging *Foundation Model* paradigm, referencing the work of Bommasani et al. (2022), who themselves refer to the Foundation Model paradigm as one which is transformational in the machine learning context.

Schneider et al. (2024) discuss an AI development pipeline, reprinted in Figure 3.1. Figure 3.1 describes such a development pipeline in terms of *levels*, *actors* and *outcomes*, with each level corresponding to exactly one actor and with exactly one actor corresponding to exactly one outcome. Each such grouping is chronologically organised, such that the *application* level follows the *system* level, which in turn follows the *model* level, etc. Such a paradigm is a particularly important consideration in the context of the present research endeavour, as it informs considerations regarding the semantic representation of AI models and systems; the *Foundation Model* paradigm in particular suggests that emerging models are not independent of existing models, but rather that downstream models or systems may inherit from pre-trained models which are upstream.

[6] *GraphDB* homepage: `https://graphdb.ontotext.com/`

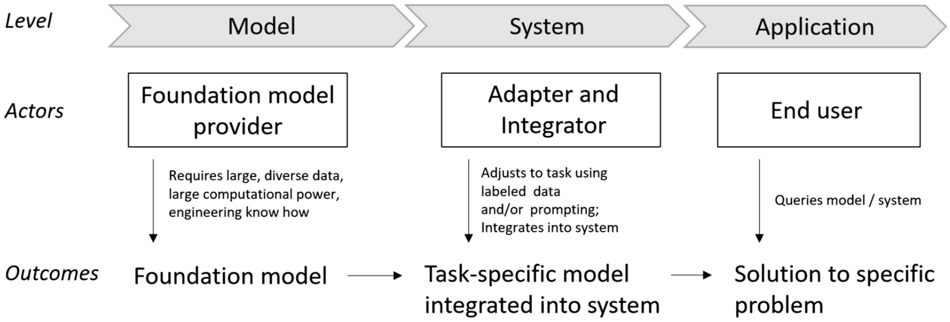

Level
Model
System
Application
Actors
Foundation model provider
Adapter and Integrator
End user
Requires large, diverse data, large computational power, engineering know how
Adjusts to task using labeled data and/or prompting; Integrates into system
Queries model / system
Outcomes
Foundation model
Task-specific model integrated into system
Solution to specific problem

Figure 3.1: Reprint from (Schneider et al. 2024), illustrating an AI development pipeline which utilises a foundation model. Licensed under a Creative Commons Attribution 4.0 International License, which is available here: `https://creativecommons.org/licenses/by/4.0/deed.en` (verified 24.06.2024).

Chapter 4

State of the art

In this section, relevant state-of-the-art research is critically discussed and compared.

4.1 AI Risk Ontology

The AI Risk Ontology (AIRO) (Golpayegani et al. 2022) is an ontology based on the requirements of the EU AI Act and international standards from the *International Organization for Standardization* (ISO). An overview of its structure, from its associated academic publication, is reprinted in Figure 4.1. In the paper introducing AIRO, two real-world use-cases are modelled to demonstrate its usefulness, and it is also demonstrated how information relevant for identifying high-risk AI systems can be obtained using SPARQL, as well as how SHACL constraints can be used to identify high-risk AI systems, based on Annex III of the AI Act.

AIRO hence shows how SHACL and SPARQL can be used to aid in compliance efforts. In another approach presented by Pandit et al. (2019),

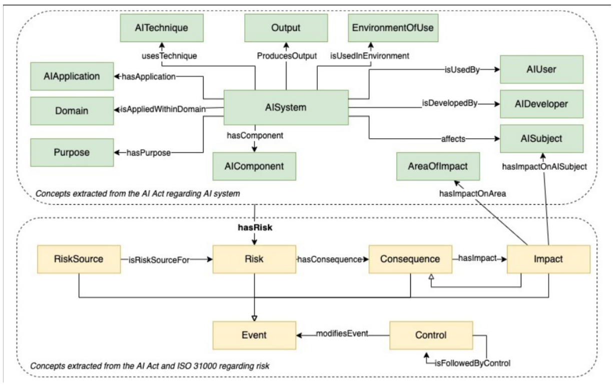


Figure 4.1: AIRO overview (Golpayegani et al. 2022). License: Creative Commons CC BY 4.0 `https://creativecommons.org/licenses/by/4.0/`

SHACL-SPARQL rules are likewise used for GDPR compliance, hence indicating the utility of these tools in aiding in legislative compliance efforts.

4.2 Documentation on the dataset & model

The *Fundamental Rights and Algorithms Impact Assessment* (FRAIA) of the Government of the Netherlands[1] distinguishes between algorithmic and data concerns in the context of risk management. Specifically, in FRAIA, the second stage of its decision making process distinguishes between input (data) and algorithm (throughput).

This is also apparent in the academic literature, where Weidinger et al. (2021), for example, discuss the concept of "bias amplification", where skews in the training data may manifest in an exaggerated ("amplified", or more than would be expected given the training data) manner in the model's outputs. This suggests that considering the data alone may be insufficient, as undetectable biases in the data may become pronounced in the model's outputs, and it may hence be wise to represent bias tests conducted not only on the dataset, but also on the outputs of the model.

4.3 Bias

4.3.1 Definitions of bias in AI

It is important, prior to further investigation of taxonomical means of representing biases, that the "bias" itself is well-defined in the context of this research.

An unambiguous definition of what is meant by "bias" in the context of this research is important, as it is clear that "bias" can take on many meanings. For example, a system can be bias-free in the machine learning sense (i.e., in the sense of balancing bias and variance in a machine learning model), while being biased in a legal sense (Crawford 2017); a machine learning model with technically good performance characteristics may nonetheless output biases which may cause harm in real-world applications. A similar distinction is made in the NIST *Towards a Standard for Identifying and Managing Bias in Artificial Intelligence* technical report, where statistical and legal contexts are distinguished as categories for addressing AI bias (Schwartz et al. 2022).

Shah et al. (2020) discuss that biases (in the sense of statistical trends, without implication of harms) and priors in natural language are not negative

[1] The FRAIA document is provided at the following link: `https://www.government.nl/documents/reports/2021/07/31/impact-assessment-fundamental-rights-and-algorithms`. Verified 15.08.2024.

per se, and Weidinger et al. (2021) defend and expand upon this in a discussion of stereotypes and unfair discrimination, stating that the tracking of such biases and priors is only problematic when training data is harmful, therefore causing the model to mirror the harms of its training dataset.

This consideration of biases as necessarily negative only when harmful mirrors the AI Act in its requirements of high-risk and general purpose AI (GPAI) systems, where there is a particular focus on the avoidance of harm, whether to individuals directly or to their fundamental rights, as set forth in the Charter of Fundamental Rights of the European Union.

4.3.2 Potential harms in AI

As part of the research carried out for this project, the harms which manifest from Large Language Models (LLMs) were investigated. Weidinger et al. (2021) discuss that harms in Language Models may be segmented into:

1. **Allocational harms**,
2. **Representational harms**

"Representational harms" refer to harms (for example, in stereotyping) which are apparent directly in the text output of the Language Model. On the other hand, "allocational harms" refers to downstream harms caused indirectly by the harmful dataset content, having influence on real-world decisions, for example, in the context of a model selecting CVs for interview.

The distinction between representational harms and allocational harms is important in the context of representing biases in AI systems, specifically, a harmful bias may manifest as a representational harm and follow cause allocational harms. In this context, it would likely be best to model representational harms on the machine learning model itself, while tending to represent allocational harms on the AI system. AIRO already supports means of representing allocational harms on the AI systems. Therefore, one may remain aligned with the task outlined in the research question while focusing primarily on representational harms, represented on the machine learning model.

AI Act

As AIRO is aligned with the EU AI Act and ISO standards, considerations of these is crucial in the development of a suitable bias taxonomy.

In the EU AI Act, it is not explicitly stated exactly which bias categories should be considered in risk management efforts, nor are tests and suitable thresholds specified. However, at a higher level, there is a discussion in the AI Act *does* of data governance practices required of high-risk AI systems, as discussed in part (f) of Article 10(2). Specifically, the AI Act specifies that datasets are subject to examination of biases that are:

1. "likely to **affect the health and safety of persons**",
2. "likely to have a **negative impact on fundamental rights**",
3. "likely to lead to **discrimination prohibited under Union law**".

In considering this, it would be difficult to develop a taxonomy for bias based on their likelihood of affecting the health and safety of persons, as this would appear dependent on the system which is affected by the biased model. However, negative impacts on fundamental rights and discrimination were considered further in the approach.

Because the AI Act places such a significant emphasis on negative impacts to EU fundamental rights, it is important that these fundamental rights are represented in a framework which aims to represent bias documentation as an extension to AIRO.

Article 27 of the AI Act is particularly important in this regard, as many high-risk AI systems will be required to perform a fundamental right impact assessment (FRIA), although the format of this assessment is yet to decided upon. In the context of a framework for the documentation of biases in AI models, providing means of indicating potential negative impacts to fundamental rights may serve to aid in the FRIA process.

GDPR

In the endeavour of developing a bias taxonomy appropriate for its intended use, another consideration was the *General Data Protection Regulation* (GDPR). Legislation such as the GDPR are relevant in this case, as EU legislature such as the AI Act does not exist in a vacuum, but rather, in the context of EU legislation more broadly. Article 9 of the GDPR concerns the processing of special categories of personal data, and its first paragraph states the following:

> "Processing of personal data revealing **racial or ethnic origin**, **political opinions**, **religious** or **philosophical beliefs**, or **trade union membership**, and the **processing of genetic data**, **biometric data for the purpose of uniquely identifying a natural person**, data concerning **health** or data concerning a **natural person's sex life** or **sexual orientation** shall be prohibited."

Each of the highlighted terms can notably be interpreted as a potential vector for bias, some of which potentially impacting fundamental rights, as per the EU Charter of Fundamental Rights (CFR).

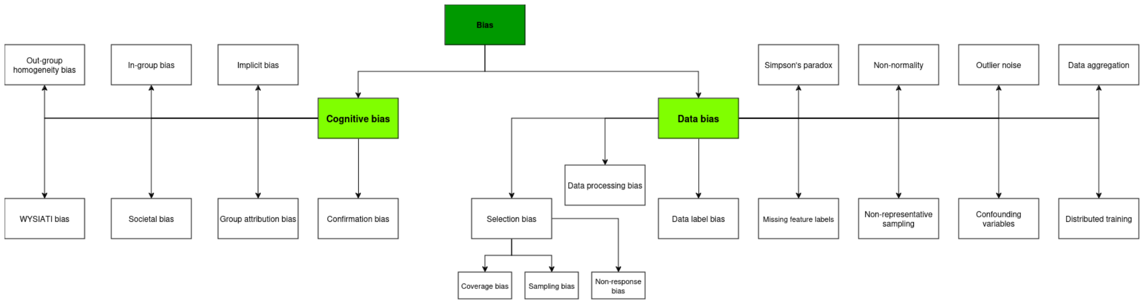


Figure 4.2: Taxonomy of bias sources, extracted from the ISO/IEC 24027:2020 document

ISO standards

With regard to international standards, ISO/IEC 24027:2020, preceding the ISO/IEC 24027:2021 international standard, is a document which provides detail pertaining to bias in AI systems — most importantly in this context, it presents a categorisation of bias sources in AI systems, and it includes also a discussion of bias mitigation. Its discussion of sources for bias in AI systems is structured in such a way that it allows for a taxonomy of bias categories to be derived from sections 7.2 and 7.3, where human cognitive bias and data bias are discussed, respectively. Figure 4.2 indicates a basic taxonomy of bias sources, derived from these sections.

Categorisations for bias

The *National Institute of Standards and Technology*, NIST, is a standards organisation which was established by the U.S. Congress. In the NIST *Towards a Standard for Identifying and Managing Bias in Artificial Intelligence* special publication (Schwartz et al. 2022), categories of AI biases are detailed, distinguishing primarily between three major categories:

1. **Systemic bias**: The systemic bias class results from institutions advantaging certain social groups while disadvantaging others (Schwartz et al. 2022).

2. **Human bias**: In the human bias class, Schwartz et al. (2022) refer to the work of Tversky & Kahneman (1974), describing human bias as reflections of "systematic errors in human thought based on a limited number of heuristic principles and predicting values to simpler judgmental operations". Schwartz et al. (2022) also note that such biases are often implicit, and relate to how information is perceived, whether by an individual or a group.

3. **Statistical/computational bias**: Schwartz et al. (2022) discuss

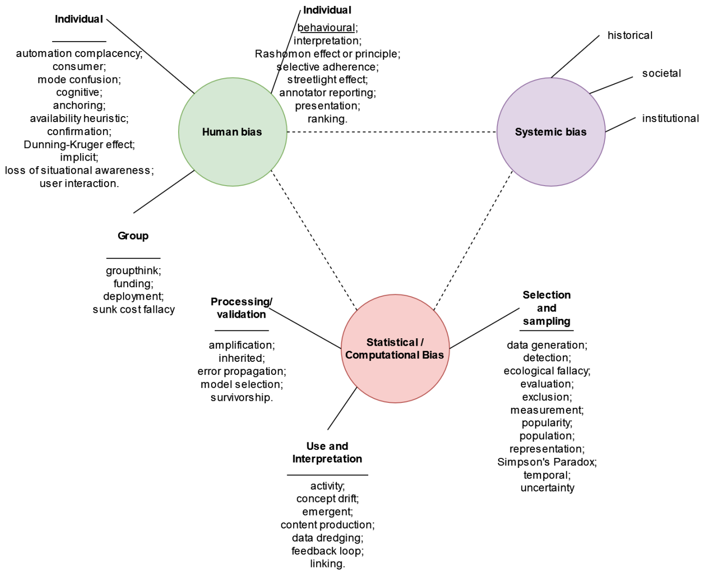


Figure 4.3: Diagram of bias categories, adapted from Figure 2 of (Schwartz et al. 2022).

> that statistical and computational biases result from a sample not being representative of the population.

A graphical representation of the bias categorisations presented by Schwartz et al. (2022) is shown in Figure 4.3, indicating in particular how the broad categorisations discussed above are divided into sub-categories, providing a rich taxonomy for bias. This categorisation notably includes a number of categories not included in ISO/IEC 24027:2020, although there is significant overlap also; for example, there is an overlap in the human bias category proposed by Schwartz et al. (2022) and the human cognitive bias category of ISO/IEC 24027:2020.

Another categorisation for bias in AI may be found in the AI Ontology (AIO) (Joachimiak et al. 2024), an open ontology built using Semantic Web technologies[2]. The AIO categorisation of biases in AI is heavily inspired by the previously discussed NIST categorisation, containing each of its categories, albeit using a different taxonomical representation to that which one would extract given the representation of categories illustrated in Figure 4.3, and an additional bias category "hostile attribution bias" is also defined as follows in AIO:

> "A use an interpretation bias where individuals perceive benign

[2] The AIO bias taxonomy can be found at the following link: `https://bioportal.bioontology.org/ontologies/AIO/?p=classes&conceptid=https%3A%2F%2Fw3id.org%2Faio%2FBias`, verified 10.08.2024.

> or ambiguous behaviours as hostile."[3]

The structure of the AI bias taxonomy in AIO differs quite substantially from that of the NIST categorisation of AI biases (Schwartz et al. 2022), in some respects, while remaining consistent in other respects. For example, while the Statistical / Computational bias category proposed by Schwartz et al. (2022) remains largely unaltered in AIO, there is an apparent collapse in the systemic and human bias categories; among other differences, individual biases are represented as a subclass of Bias rather than Human Bias (as in the bias categorisation of Schwartz et al. (2022)), and this is even moreso the case for the systemic bias category, where this whole class is flattened in the AIO categorisation.

4.4 Measuring and mitigating bias

4.4.1 Fairness

Discrimination

Mehrabi et al. (2021) describe that *discrimination* is a source of unfairness, and consider the term in particular from the perspective of algorithmic unfairness, further describing the term as "a source for unfairness that is due to human prejudice and stereotyping based on the sensitive attributes which may happen intentionally or unintentionally". Mehrabi et al. (2021) contrast the notion of unfairness with bias, which the authors refer to as unfairness that is "due to the data collection, sampling, and measurement". This contrasts categorisations of bias (sources) such as those proposed in the previously discussed NIST technical report on identifying managing bias in AI (Schwartz et al. 2022) as well as the ISO/IEC 24027:2020 document, as indicated in Figures 4.3 and 4.2 respectively; each of these bias categorisations refer to "societal bias", described as follows in ISO/IEC 24027:2020:

> "Societal bias occurs when one or more similar cognitive biases (conscious or unconscious) are being held by many individuals in society. It manifests in ML when models learn or amplify pre-existing, historical patterns of bias in datasets. This societal bias originates from society at large and could be closely related to other cognitive or statistical biases."

Such considerations of societal bias have also been discussed, for example, in the context of language models (LMs), discussed by Weidinger et al. (2021)

[3] Found at the following link: `https://bioportal.bioontology.org/ontologies/AIO/?p=classes&conceptid=https%3A%2F%2Fw3id.org%2Faio%2FHostileAttributionBias`. Verified 10.08.2024.

in the *Ethical and social risks of harm from Language Models* paper. In this paper, Weidinger et al. (2021) discuss that where training data was gathered from contexts where inequality is the status quo, training data may reflect historical patterns of systemic injustice.

However, despite a noted preference for distinct terminology when describing unfairness and bias, Mehrabi et al. (2021) acknowledge that bias, like discrimination, can be a source of unfairness due to human prejudice and stereotyping, but justify their position that these should be represented as separate concepts, as this is more aligned with the algorithmic fairness literature.

Mehrabi et al. (2021) further discuss that this notion of discrimination can be further specified as either **explainable** or **unexplainable**. Referring to the work of Kamiran & Žliobaitė (2013), Mehrabi et al. (2021) describe explainable discrimination as differences in treatment which are justifiable and explained, and hence not illegal. An example is also provided, where Mehrabi et al. (2021) once more reference the work of Kamiran & Žliobaitė (2013), describing an observance of discrimination on the UCI Adult dataset (Asuncion et al. 2007)[4], where the annual income for males is observed to be higher than it is for females. However, in this case, it is noted that this can be attributed to females working fewer hours than males.

Unexplainable discrimination, on the other hand, is described by Mehrabi et al. (2021) as when "discrimination against a group is unjustified and therefore considered illegal", and be further specified as **direct discrimination** or **indirect discrimination**.

To describe direct discrimination, Mehrabi et al. (2021) reference the work of Zhang et al. (2016), describing it as discrimination where an individual's protected attributes are explicitly discriminated against. In contrast, Mehrabi et al. (2021) describe indirect discrimination (with reference to the work of Zhang et al. (2016)) as individuals appearing "to be treated on seemingly neutral and non-protected attributes", but where "protected groups, or individuals, still get to be treated unjustly as a result of implicit effects from their protected attributes".

Fairness & Fairness Metrics

The ISO/IEC 24027:2020 document considers that "fairness" as a concept is related to bias, albeit not the same. In ISO/IEC 24027:2020, it is acknowledged that providing a definition for "fairness" is difficult, and no definition of fairness is hence provided in the document, as it is considered to have contested and varying definitions across cultures, geographies, etc.

However, a number of negative impacts which an AI system can have

[4] The UCI Adult dataset is used as part of a use-case in the evaluation section of this report.

are outlined in ISO/IEC 24027:2020, which are noted as having potential to be perceived as "unfair", namely:

1. **Unfair allocation**.

2. **Unfair quality of service**.

3. **Stereotyping**.

4. **Denigration**.

5. **"Over" or "under" representation**.

In *A Survey on Bias and Unfairness in Machine Learning* (Mehrabi et al. 2021), a number of the most commonly used definitions for fairness are outlined, comprising 10 in total, e.g., equalised odds:

> "A predictor $\hat{Y}$ satisfies equalized odds with respect to protected attribute A and outcome Y, if $\hat{Y}$ and A are independent conditional on Y. $P(\hat{Y} = 1|A = 0, Y = y) = P(\hat{Y} = 1|A = 1, Y = y), y \in \{0, 1\}$" (Hardt et al. 2016)

In the ISO/IEC 24027:2020 document, similar definitions are also represented in its Section 8, "Assessment of bias and fairness in AI systems", where, as in the work of Mehrabi et al. (2021), definitions for equalized odds and equality of opportunity are provided, among others. Mehrabi et al. (2021) also discuss that such definitions of fairness can be further divided into three types:

1. **Individual fairness:** With reference to the work of Dwork et al. (2012), Kusner et al. (2017), Mehrabi et al. (2021) describe the concept of individual fairness as giving "similar predictions to similar individuals".

2. **Group fairness:** With reference to the work of Dwork et al. (2012), Kusner et al. (2017), Mehrabi et al. (2021) describe the concept of group fairness as treating "different groups equally".

3. **Subgroup fairness:** With reference to the work of (Kearns et al. 2018, 2019), Mehrabi et al. (2021) describe subgroup fairness as intending to "obtain the best properties of the group and individual notions of fairness", noting as an example that it "picks a group fairness constraint like equalizing false positive and asks whether this constraint holds over a large collection of subgroups".

A number of these definitions for fairness are represented in *fairness metrics*, evident in their representation in software libraries. In the ISO/IEC 24027:2020 document, it is stated that "One way to uncover evidence of unwanted biases is to assess the system's outputs using one or more fairness metrics". Hence, in this project, fairness metrics are carefully considered.

One such software library, which Mehrabi et al. (2021) discuss as an assessment tool, providing an extensive range of fairness metrics and bias mitigation measures, is *AI Fairness 360* (AIF360) (Bellamy et al. 2018), a toolkit for algorithmic fairness for Python (and R), which is open source. Among the myriad tools provided in the AIF360 toolkit, one is *equalised odds difference*, a measurement of the equalised odds fairness metric, as considered previously. In the AIF360 software package, this is represented as a fairness metric on `equalized_odds_difference` method of the `aif360.metrics.ClassificationMetric` class. AIF360 distinguishes between individual and group fairness[5], but does not consider subgroup fairness.

The *Fairness Metrics Ontology* (Franklin et al. 2022) is an ontology which allows for the representation of fairness metrics and fairness evaluations on machine learning models, containing classes pertaining to fairness, such as "Fairness Notion" (and subclasses thereof), "Fairness Metric" (which measures a Fairness Notion, as well as "Statistical Metric".

4.4.2 Bias mitigation

Castelnovo et al. (2022), in *A Clarification of the Nuances in the Fairness Metrics Landscape*, describe that scientific literature pertaining to fairness in ML has primarily focused on how to accomplish the following two aspects:

(i) **Measurement**: The assessment or measurement of fairness / bias;

(ii) **Mitigation**: The mitigation thereof, in models, if necessary.

In a discussion of methods for fair machine learning Mehrabi et al. (2021) discuss that methods targeting biases generally fall into the following categories:

1. **Pre-processing:** In "pre-processing", Mehrabi et al. (2021) describe the transformation of data in order to remove underlying discrimination, referencing the work of d'Alessandro et al. (2017).

2. **In-processing:** In "in-processing", Mehrabi et al. (2021) describe techniques which "try to modify and change state-of-the-art learning

[5] Details on the types of metrics supported are available on the AIF360 *Getting Started* page, accessible at the following link: `https://aif360.readthedocs.io/en/latest/Getting%20Started.html`. Verified 11.08.2024.

algorithms in order to remove discrimination during the model training process", referencing the work of d'Alessandro et al. (2017).

3. **Post-processing:** In "post-processing", Mehrabi et al. (2021) discuss that it is "performed after training by accessing a holdout set which was not involved during the training of the model" referencing the work of d'Alessandro et al. (2017).

These categorisations correspond with those in the AI Fairness 360 software package[6] (Bellamy et al. 2018), represented in `aif360.algorithms.preprocessing`, `aif360.algorithms.inprocessing` and `aif360.algorithms.postprocessing`, respectively.

Beyond academic literature and state-of-the-art libraries in bias mitigation techniques, it is important in this context that EU legislation, as well as international standards are considered also. In Article 10(2) of the EU AI Act, a data governance and management practice is listed as a requirement for high-risk AI systems, concerning:

> "**10(2), (g)** appropriate measures to detect, prevent and mitigate possible biases identified according to point (f)"

Article 10(2) point (f) concerns examination in view of possible biases, as discussed in Section 4.3.2. This section of the AI Act highlights the requirement for appropriate measures in a risk management process, but does not explicitly specify that technical documentation is required to demonstrate mitigation procedures.

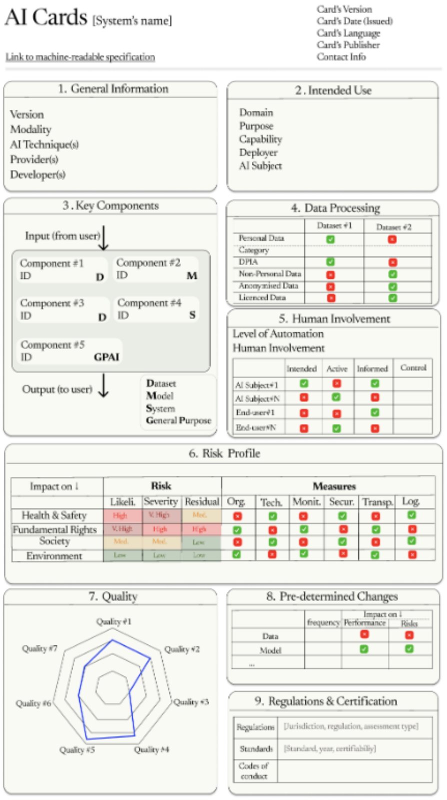
4.5 Documentation in AI

4.5.1 AI Cards

In *AI Cards* (Golpayegani et al. 2024), a means of representing risk documentation required according to the AI Act is proposed, representing relevant information in both machine-readable and human-readable formats. For the machine-readable representation of AI cards, AIRO (Golpayegani et al. 2022) is used. AI Cards is hence a significant consideration in the context of the present research, as it is important that its approach is considered in the framework for representing bias documentation proposed in the present research.

The information elements represented in an AI card are indicated in Figure 4.4, where each pane indicates an information element represented. Based upon the description provided in the AI Cards paper (Golpayegani

[6] Each of the categories and associated algorithms are listed at the following link: `https://aif360.readthedocs.io/en/latest/modules/algorithms.html`. Verified 11.08.2024.



AI Cards [System's name]

Link to machine-readable specification

Card's Version
Card's Date (Issued)
Card's Language
Card's Publisher
Contact Info

1. General Information

Version
Modality
AI Technique(s)
Provider(s)
Developer(s)

2 . Intended Use

Domain
Purpose
Capability
Deployer
AI Subject

3 . Key Components

Input (from user)

Component #1
ID D

Component #2
ID M

Component #3
ID D

Component #4
ID S

Component #5
ID GPAI

Output (to user)

Dataset
Model
System
General **P**urpose

4. Data Processing

	Dataset #1	Dataset #2
Personal Data	✓	✗
Category		
DPIA	✓	✗
Non-Personal Data	✗	✓
Anonymised Data	✗	✓
Licenced Data	✗	✓

5. Human Involvement

Level of Automation
Human Involvement

	Intended	Active	Informed	Control
AI Subject#1	✓	✗	✓	
AI Subject#N	✗	✓	✗	
End-user#1	✗	✗	✓	
End-user#N	✗	✓	✗	

6. Risk Profile

Impact on ↓	**Risk**			**Measures**					
	Likeli.	Severity	Residual	Org.	Tech.	Monit.	Secur.	Transp.	Log.
Health & Safety	High	V. High	Med.	✗	✓	✗	✓	✗	✓
Fundamental Rights	V. High	High	High	✓	✗	✓	✗	✓	✗
Society	Med.	Med.	Low	✗	✓	✗	✓	✗	✓
Environment	Low	Low	Low	✓	✗	✓	✗	✓	✗

7. Quality

8. Pre-determined Changes

		Impact on ↓	
	frequency	Performance	Risks
Data		✗	✗
Model		✓	✓
...			

9. Regulations & Certification

Regulations	[Jurisdiction, regulation, assessment type]
Standards	[Standard, year, certifiabiliy]
Codes of conduct	

Figure 4.4: Visual representation of AI Cards (Golpayegani et al. 2024) from Figure 3 of the AI Cards paper. License: Creative Commons CC BY 4.0 `https://creativecommons.org/licenses/by/4.0/`

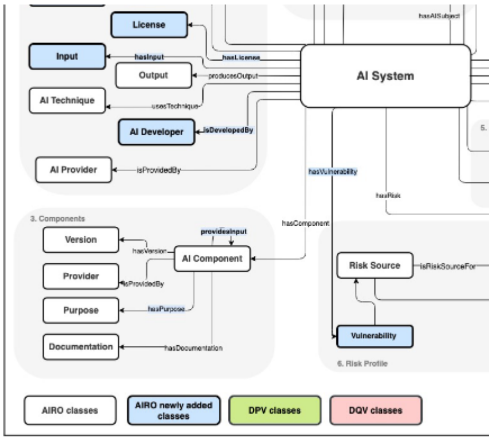


Figure 4.5: Adapted from Figure 4 of the AI Cards paper (Golpayegani et al. 2024). License: Creative Commons CC BY 4.0 `https://creativecommons.org/licenses/by/4.0/`

et al. 2024), it appears that documentation pertaining to bias on machine learning models should be represented in the *Key components* information element of the AI Card.

Figure 4.5 is adapted from a Figure in the AI Cards paper (Golpayegani et al. 2024), and indicates the areas of AIRO upon which it expands, which the design of the present research project is most concerned with. Golpayegani et al. (2024) note that for each key component, “its name, version and link to documentation or ID are presented”, which corresponds to that which is observed in Figure 4.5. Given that the `airo:MLModel` class is represented as a subclass of `airo:AIComponent`, representing any documentation classes as a subclass of `airo:Documentation` and additionally providing values for the `airo:hasVersion`, `airo:isProvidedBy` and `airo:hasPurpose` predicates, documentation related to bias is compatible with AI Cards.

From the Proctify use-case[7], it is clear that datasets are specified as being of the `AIComponent` type, and these are directly linked to the AI system with the `hasComponent` property. However, when seeking to represent a model development pipeline, as might be desirable when considering the emerging foundation model paradigm (Schneider et al. 2024, Bommasani et al. 2022), where modified / fine-tuned variations of general-purpose models are becoming prevalent, representing all datasets and models at the same level (each equally a component of a single AI system), this lineage is not as well represented as it might be if components such as datasets were attached to instances of the model. With this approach, a clearer view of how each model was trained / altered can be obtained through

[7] `https://github.com/DelaramGlp/airo/blob/main/usecase/proctify.ttl`. Verified 12.08.2024.

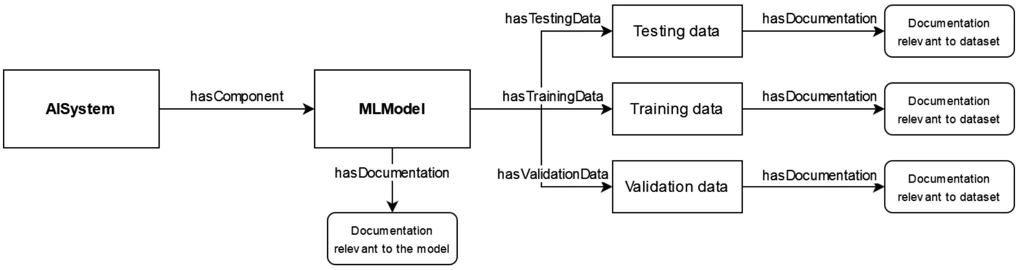


Figure 4.6: Indication of a structure which is compliant with the AIRO constraints, representing the components of components, each with attached documentation.

recursive specification of parent models, with their training datasets and risk documentation attached, implicitly aiding the satisfaction of a technical documentation requirement for high-risk AI systems, outlined in Annex IV of the AI Act:

> "**Annex IV 2(a)** the methods and steps performed for the development of the AI system, including, where relevant, recourse to pre-trained systems or tools provided by third parties and how those were used, integrated or modified by the provider;"

However, despite the representation of the Proctify use-case (Golpayegani et al. 2024), it is not clear that specifying datasets and models as a direct component of the AI system itself is a limitation in AI Cards in the first place; the `airo:hasComponent` property (as of commit `fa9da94`) does not specify `airo:AISystem` in its domain, nor does it specify a domain at all. Therefore, a structure as indicated in Figure 4.6 is legal within the AI Cards framework, given that datasets are likewise specified as instances which are a subclass of `airo:AIComponent`, and each of the `airo:hasTrainingData`, `airo:hasTestingData` and `airo:hasValidationData` are subproperties of `airo:hasComponent`.

4.5.2 Dataset documentation

However, it is clear that for datasets, additional documentation could be provided to aid in downstream risk identification and management. The reason for this is that tests may be carried out with a particular view of an AI model's application, not considering how it may be applied downstream.

In such cases, it would seem sensible to also provide dataset documentation to be used alongside test report documentation for the purpose of downstream risk management tasks.

In Annex IV of the AI Act, the following is specified as one of the requirements in the technical documentation of a AI system providers, before a high-risk AI system can be brought to market:

> "**2(d)** where relevant, the data requirements in terms of datasheets describing the training methodologies and techniques and the training data sets used, including a general description of these data sets, information about their provenance, scope and main characteristics; how the data was obtained and selected; labelling procedures (e.g. for supervised learning), data cleaning methodologies (e.g. outliers detection);"

Moreover, there are requirements of general-purpose AI model providers, specified in Article 53(1) and described in Annex XI of the AI Act, the most important point from the perspective of representing dataset documentation likely being the following excerpt from Annex XI, specifying that the following documentation must be provided:

> "**2(c)** information on the data used for training, testing and validation, where applicable, including the type and provenance of data and curation methodologies (e.g. cleaning, filtering etc.), the number of data points, their scope and main characteristics; how the data was obtained and selected as well as all other measures to detect the unsuitability of data sources and methods to detect identifiable biases, where applicable;"

While technical documentation requirements (specifically, those outlined in Annex IV and Annex XI) are of greatest concern in consideration of the research question, it is also important to consider the data governance practices mandated by the AI Act, in particular with regard to high-risk AI systems.

In particular, Article 10 of the AI Act, as discussed in Section 4.3.2, warrants consideration, as it concerns on data governance and management practices required for high-risk AI systems. Of those practices listed, in consideration of Article 10(2) of the AI Act, the following are directly relevant to dataset risk documentation:

1. **10(2), (b)** "Data collection processes" and the "origin of data". In the case of personal data, the "original purpose of the data collection".

2. **10(2), (c)** "relevant data-preparation processing operations, such as annotation, labelling, cleaning, updating, enrichment and aggregation."

3. **10(2), (d)** "the formulation of assumptions, in particular with respect to the information that the data are supposed to measure and represent"

4. **10(2), (e)** "an assessment of the availability, quantity and suitability of the data sets that are needed"

In the following sections, a number of approaches in the academic literature are considered for representing dataset information relevant to risk management are considered, considering the requirements described Annex IV & Annex XI of the EU AI Act in particular, as well as those data governance and management practices required for high-risk AI system, as discussed in Article 10(2).

Dataset Nutrition Label

The *Dataset Nutrition Label* (Holland et al. 2018) is a diagnostic framework which Holland et al. (2018) claim "lowers the barrier to standardized data analysis by providing a distilled yet comprehensive overview of the dataset 'ingredients' before AI model development."

The Dataset Nutrition Label is divided into seven modules (referred to as "modules" because it is described as a modular architecture), namely **metadata**, **provenance**, **variables**, **statistics**, **pair plots**, **probabilistic model** and **ground truth correlations**.

Using these modules, the Dataset Nutrition Label (Holland et al. 2018) satisfies the requirement from Annex IV of a general description and provenance as well as, to some degree, scope and main characteristics, but does not specify means of describing how the data was obtained and selected or labelling procedures and data cleaning methodologies (although these could be described in the description field of the metadata module, this is not ideal). When considering Annex XI and Article 10(2), this remains an issue with the use of the Dataset Nutrition Label when seeking to align with the requirements of the EU AI Act, as can be seen in particular in Article 10(2) part (b), where emphasis is once more placed on data collection processes and data origin.

Hence, while the Dataset Nutrition Label is elegant and provides important information about datasets in a format which is easy to understand, the fields represented in the Dataset Nutrition Label alone do not appear sufficient for aiding in compliance with the AI Act, as is the case in the context of the present research, considering the research question.

Datasheets for Datasets

Another means of representing risk management information pertaining to datasets in ethical AI research is the approach of *Datasheets for datasets* (Gebru et al. 2021). Gebru et al. (2021) propose that *datasheets* be used to describe each dataset, with each datasheet comprising fields — prompt questions, with fields for responses. Datasheets are divided into **motivation**, **composition**, **collection process**, **preprocessing/cleaning/labeling**, **uses**, **distribution** and **maintenance** sections. Fields may concern, for example, dataset provenance, intended use and source of data, the nature of the data, etc. Returning to the requirements of the EU AI Act, datasheets (as defined by Gebru et al. (2021)) have:

1. Considering the data encoded in the fields collectively, what may be described as a “general description” (req. Annex IV, Annex XI);

2. Information about provenance (Datasheets have fields for the creator of the dataset, maintainer of the dataset and funded the dataset, in particular) (req. Annex IV, Annex XI);

3. Information pertaining to dataset scope and main characteristics, as indicated in the motivation, composition and collection processes sections in particular (req. Annex IV, Annex XI);

4. Information pertaining to how the data was obtained and selected, specifically in the *collection process* section of the datasheet as well as concerning sampling strategies (req. Anniv IV, Annex XI; useful for Article 10(2) part (b));

5. Fields which prompt for details regarding relevant data-preparation processing operations, specifically in the preprocessing/cleaning/labeling section (useful for Article 10(2) part (c));

6. Information about what the data represent from the composition section (useful for Article 10(2), part (d));

7. Information about tasks for which the dataset should not be used and its distribution (useful for Article 10(2), part (e))

Therefore, given the more extensive overlap in concerns, the approach proposed by Gebru et al. (2021) in *Datasheets for datasets* aligns more closely with those requirements set forth in the EU AI Act than the approach proposed by Holland et al. (2018) in *Dataset Nutrition Label.*

4.6 Doc-BiasO: An existing ontology for documenting bias

Doc-BiasO (Russo & Vidal 2024) is an ontology which aims to represent bias documentation across machine learning (ML) pipelines, utilising state of the art Semantic Web technologies, as is the case with the work presented in the present research report. It utilises the aforementioned *Fairness Metrics Ontology* (Franklin et al. 2022) and utilises the AI Ontology (Joachimiak et al. 2024), which is based upon NIST technical report (Schwartz et al. 2022) for its representation of a bias categories.

However, there are some significant differences between the approach proposed in Doc-BiasO (Russo & Vidal 2024) and the approach developed in this research endeavour. Firstly, despite its mention as relevant work, Doc-BiasO does not directly provide means through which it may be integrated with AIRO[8](Golpayegani et al. 2022), a more broad framework for AI risk management, which would allow for bias risks on machine learning models to be represented alongside other AI system risks (on the same knowledge graph), which may be of benefit.

Furthermore, while Doc-BiasO is aligned with the aforementioned NIST technical report (Schwartz et al. 2022) through its adoption of bias terms from the AI Ontology (Joachimiak et al. 2024), this is a national standard (albeit with significant international influence) from the United States rather than an international standard, such as those standards published by ISO; because AIRO relies on international standards from ISO, incorporation of taxonomical structures which are more aligned with those found in ISO standards would provide a more consistent integration between AIRO and the presented bias framework.

The *Fairness Metrics Ontology* (Franklin et al. 2022) is also used in Doc-BiasO, and while elegant, it is very specific to fairness metrics, e.g., defining its own custom class for "Evaluation Metric". In the context of representing fairness metrics in AIRO, which is a broader framework for AI risk management, it may be preferable to utilise a more generic existing Semantic Web reporting vocabulary (such as the *Evaluation and Report Language* (EARL)[9]) to represent tests; this would allow for test reports in other areas, beyond bias, to be represented using the same more generic foundation, which may have benefits, e.g., when queryin using SPARQL.

Finally, while in Doc-BiasO, both DQV and DCAT are utilised to represent information about datasets, it does not align its documentation of dataset properties with the requirements set forth in the EU AI Act, nor does it explicitly consider interactions between the risk management information represented on datasets and bias testing, indicating the utility of a bias test documentation approach when used alongside comprehensive dataset documentation, aligned with the EU AI Act.

[8] In *Doc-BiasO*, it is specified that AIRO is not imported *in its entirety*. However, it appears that while reusing concepts from AIRO, Doc-BiasO does not clearly integrate with AIRO's fundamental structure (i.e., given an AI system modelled using AIRO, it is not clear how one might integrate Doc-BiasO as an extension of this).

[9] The EARL schema can be found at the following link: `https://www.w3.org/TR/EARL10-Schema/`. Verified 13.08.2024.

Chapter 5

Design

5.1 Modularity and the facilitation of machine learning development pipelines

In this section, structural considerations based upon an analysis of AIRO are provided. In particular, the use-cases represented in the paper which introduced AIRO (Golpayegani et al. 2022) are considered, and a number of areas considered are outlined, guiding the overall design.

For the purpose of facilitating modularity and development pipelines for machine learning models, the design presented in this chapter focuses its representation on the machine learning model, represented in AIRO in `airo:MLModel`[1] rather than the AI system (`airo:AISystem`). To facilitate the representation of fine-tuned or otherwise specialised machine learning models (e.g., models derived from foundation models trained on massive and general-purpose data), a model development pipeline is representable using the recursive `airo:hasModel` predicate, already present in AIRO as of the time of writing. This approach, which support the representation of such development pipelines is aligned with the foundation model paradigm, as described by Schneider et al. (2024), Bommasani et al. (2022), as discussed in Section 3.4.

An indication of how this development pipeline is envisioned using terms from AIRO is illustrated in Figure 5.1. In the context of this research, representing features relevant to bias risk management on a particular machine learning model (`airo:MLModel`) would facilitate enhanced transparency when considering bias risks in downstream systems, as in the "AI system" as indicated in Figure 5.1. In prior versions of AIRO, the `airo:hasModel` predicate did not exist, but rather `airo:hasComponent`. To remain con-

[1] As AIRO is still under active development, `airo:MLModel` has since been changed to `airo:AIModel` (changed in Git commit `71463dd` on 24.07.2024). For the remainder of this section, this class remains referred to as `airo:MLModel`.

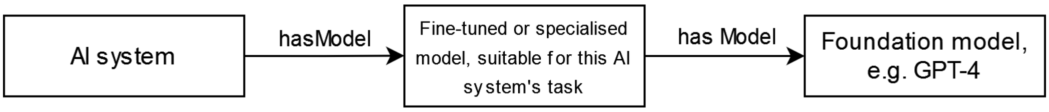

AI system
hasModel
Fine-tuned or specialised model, suitable for this AI system's task
has Model
Foundation model, e.g. GPT-4

Figure 5.1: The development pipeline flow, as it is may be represented using this AIRO.

sistent with AIRO's most recent developments as of the time of writing, the `airo:hasModel` property has been the preferred variant in the project implementation.

Using this approach, whereby documentation directly related to risks of the model itself is represented directly on the model (as opposed to on the downstream AI system), the following were noted as benefits (in contrast with the representation of the bias risk source as in Figure 2.1):

1. **Less error-prone**. In cases such as those described in the beginning of this section, where a foundation model (general-purpose, pre-trained model) is involved, or in simpler cases where a system utilises a common machine learning model for its task, it may be expected that risks (in this case, bias risks) identified for this model (or those upstream models, as it might be) may be relevant when modelling risks in the system.

   If each of the risks which were identified for the upstream system are modelled as risks in the downstream system (represented directly as risk sources (`airo:RiskSource`) on the AI system (`airo:AISystem`)), then there is a significant risk that the relevant risks identified in the upstream system may be misrepresented in the risks represented by the downstream providers, for example, because an individual or organisation wishing to model the risks in their AI system may not be entirely aware of those risks identified in the model, as the representation of the upstream model is not directly represented in the same knowledge graph.

2. **Redundant information**: Even in those cases where relevant risks identified in a particular model are faithfully represented directly on the AI system, this entails a lot of redundant information. A parallel to this is apparent in object-oriented programming, where many subclasses may inherit behaviour from a single superclass.

   Representing information related to bias directly on a model, referencing existing models would be made simple, using Extensible Markup Language (XML) namespaces.

From the perspective of an AI system in AIRO, bias risks can be considered rather as risk sources (`airo:RiskSource`). This is perhaps most apparent on analysis of the AIRO paper (Golpayegani et al. 2022) and its illustration of its capabilities using two use-cases; the first of these is *Uber's real-time ID check system* as previously discussed in the motivation section with reference to Figure 2.1, and the second is the *VioGén Domestic Violence System*. In the former, the `RiskSource` is `bias_in_algorithm_training`, and the `Risk` is `inaccuracy_in_identifying_bame_drivers`; in the latter, the `RiskSource` is `poor_quality_of_input_data`, and the `Risk` is `inaccurate_prediction`. Considering this, the component machine learning model(s) are more likely to manifest as risk sources than risks, as instances of `Risk` in AIRO appear to pertain to risks in the application of the AI system. Hence, this approach which develops a bias framework on the `airo:MLModel` component could be viewed as a contribution which enhances the representation of AIRO risk sources.

Facilitating such machine learning development pipelines may aid in fulfilling the technical documentation requirements outlined in Annex IV and Annex XI of the AI Act, and is hence aligned with the research question, aligning in particular with AI Act Annex IV(2) part (a):

> "'**(a)** the methods and steps performed for the development of the AI system, including, where relevant, resourse to pre-trained systems or tools provided by third parties and how those were used, integrated or modified by the provider;'"

5.2 Considering the algorithm and the dataset

As discussed in the state of the art chapter, in Section 4.2, various sources refer to there being a split in concerns between the dataset and the algorithm, or outputs, with regard to risk management, and these are hence important considerations in any framework for representing bias risk sources for such systems.

However, representing the number of parameters, training accuracy, etc., in a meaningful way such that it would inform risk management by reliable means would likely be difficult and error-prone[2]. Therefore, the consideration of Birhane et al. (2024) is adopted, where "product/model/algorithm audits" are considered to typically evaluate "specific deployed systems", i.e., the algorithmic aspect is accounted for with tests on model outputs, rather than considering the specifics on the model architecture; this accounts for the "algorithmic" component in an analysis of outputs. Such a consideration of the model/algorithm entails that there are no direct mappings represented between that risk information identified in the dataset and how the model should be expected to behave, acknowledging the difficulty of developing

[2] It would be difficult or impossible to infer risks in model outputs based on model metadata, represented in RDF, alone

such mappings for knowledge graphs, instead investigating information which is available before and after the model training process.

To concisely summarise, for the reasons outlined above, documentation pertaining to bias risk management is represented on the machine learning model in RDF at two levels:

1. **Machine learning model.** Representation of bias documentation in RDF on the machine learning model itself (as properties/predicates with a `airo:MLModel` domain);

2. **Dataset.** Documentation may be represented on any dataset associated with the machine learning model, related with properties/predicates as required, defined in AIRO:

   - `airo:hasTrainingData`.
   - `airo:hasTestingData`.
   - `airo:hasValidationData`.

5.3 Bias taxonomy development

When designing a framework for representing bias documentation as an extension to AIRO, an important consideration is which bias categories should be used to represent the outcomes of test results, providing consistent means of referring to a particular bias concept.

5.3.1 One dimensional or taxonomical?

An initial consideration when seeking to represent bias categories would be to consider each category on a singular stratum. In an RDF representation, this might involve representing each bias category as an instance of a parent class, `Bias`. Such a representation is illustrated in Listing 8.

```
:Bias rdf:type owl:Class .

:CognitiveBias rdf:type :Bias .
:SocietalBias rdf:type :Bias .
:EthnicBias rdf:type :Bias .
```

Listing 8: A representation of how a one-dimensional representation might look in RDF, using TURTLE syntax. Each bias category, regardless of its generality or the relationship between it and other bias categories, is represented on a single stratum, each an instance of a `Bias` class.

However, this one-dimensional consideration does not clearly indicate how different bias categories are related, by consequence of its simplicity; some bias categories are more general than others. Defining more general bias concepts at minimum eases comprehension for humans as they attempt to parse the structure, which is important when considering those modelling risks; this approach would place extra burden on those modelling a test result to locate the most specific bias category to describe their test result. For the risk modeller, a one-dimensional list of bias categories would also make it more difficult to identify compounding risks.

Considering bias categories as a simple set of terms also likely harms maintainability, as a taxonomical representation would facilitate a means for quickly locating similar terms and assessing how a prospective new term would fit into the existing structure. Hence, a taxonomical structure to describe bias types is adopted.

5.3.2 Approach

For the task of developing a taxonomy for bias aligned with the research question outlined in Section 2.2. Specifically, the following approach was chosen:

1. **ISO core**: Represent the taxonomy extracted from ISO 24027:2020 in RDF. A graph of the taxonomy derived from ISO/IEC 24027:2020 is illustrated in Figure 4.2. This taxonomy features two major categories derived from the parent `Bias` class, **cognitive bias** and **data bias**, from which more subclasses are defined.

2. **Societal biases**: Societal biases extracted from Article 9 (as discussed in Section 4.3.2) of the GDPR as an extension of the taxonomy extracted in (1), as well as **gender bias**.

3. **Expansion with AIO bias types**: Expand with bias terms from the AI Ontology (Joachimiak et al. 2024), only creating new classes where required, to provide a richer taxonomy while remaining generally aligned with ISO standards through ISO 24027:2020.

The choice to represent the core of the bias taxonomy in a manner generally aligned with ISO standards, specifically deriving a taxonomy from the pre-standard ISO 24027:2020, was made primarily to remain consistent with AIRO, which is itself aligned with the EU AI Act and ISO standards. Moreover, on consideration of the state of the art literature analysed, it was noted that there is significant overlap between this taxonomy and the categorisation of bias types proposed in the previously discussed NIST technical report (Schwartz et al. 2022).

The next extension upon this concerns societal biases specifically, and this was prioritised given the AI Act text, which emphasises negative impacts to fundamental rights. Because of mappings which may be specified between societal bias[3] categories (e.g., gender bias, racial bias) and potential negative impacts to fundamental rights, the societal bias category was prioritised as an initial extension of the initial taxonomy.

The final extension was implemented in order to provide a richer taxonomy, integrating classes from the AI Ontology[4] (Joachimiak et al. 2024) (as in the approach proposed by Russo & Vidal (2024)), and by proxy, bias categories described in the NIST technical report written by Schwartz et al. (2022). This step was carried out because on analysis of state of the art literature, whilst no obvious flaws were noted with the taxonomy based upon ISO/IEC 24027:2020 (including the extension for societal bias categories), this taxonomy alone was rather limited in its quantity of terms. In part, this could be attributed to the difficulty in extracting a branch of this taxonomy from ISO/IEC 24027:2020 related to computational / algorithmic components, a branch which is presented in the aforementioned NIST technical report (Schwartz et al. 2022) and by extension, the AI Ontology (Joachimiak et al. 2024).

In the RDF representation of this taxonomy, the `Bias` class was represented as class in OWL, as in Listing 9. This `Bias` class was then further divided into subclasses, such as `CognitiveBias`, until a full representation of this taxonomy was developed.

```
:Bias rdf:type owl:Class ;
    terms:source "ISO/IEC 24027:2020, 3.3.2"@en ;
    rdfs:comment "systematic difference in treatment
       of certain objects, people, or groups in
       comparison to others."@en ;
    rdfs:label "bias"@en .
```

Listing 9: The RDF representation of the `Bias` class in TURTLE format.

5.4 Mapping bias onto fundamental rights risks

In the EU AI Act, reference to bias mitigation places significant emphasis on the avoidance of harms to individual fundamental rights. However,

[3] Considering, for example, the definition provided in ISO/IEC 24027:2020, indicated in Section 4.4.1.

[4] AIO bias categories can be found at the following link: `https://bioportal.bioontology.org/ontologies/AIO/?p=classes&conceptid=root`. Last verified 24.07.2024.

it is difficult to specify a threshold for a particular bias type, given the results of a particular test, which would indicate a particular violation to an individual's fundamental rights.

Despite this, a novel approach is developed in this project which involves mapping bias categories onto potential fundamental right violations / harms. While this approach does not indicate probabilities of violation, the ability to query for *potential* fundamental rights violations given failed bias tests may prove useful in EU AI Act compliance practices (for example, for fundamental rights impact assessments (FRIA) as detailed in AI Act Article 27), aligned with the research question.

For this purpose, the Charter of Fundamental Rights (CFR) of the European Union was represented in RDF format as instances of a `EUFundamentalRight` class, with the description annotation for each fundamental right corresponding to its textual description in the Charter. A novel predicate, `potentiallyViolates`, was created, allowing for societal bias categories represented in the taxonomy for bias to be mapped onto instances of the `EUFundamentalRight` class to indicate that this bias category *may* indicate harm to a fundamental right. This is illustrated in Listing 10. As discussed in the State of the art chapter, the representation of these EU fundamental rights, as well as the mappings between societal bias categories and potential violations of these fundamental rights, aligns with the data governance and

```
:EUFundamentalRight rdf:type owl:Class .
:RightNonDiscrimination rdf:type :EUFundamentalRight
   ;
    terms:source
       "http://data.europa.eu/eli/treaty/char_2012/art_21/oj"@en
       ;
    rdfs:comment "1. Any discrimination based on any
       ground such as sex, race, colour, ethnic or
       social origin, genetic features, language,
       religion or belief, political or any other
       opinion, membership of a national minority,
       property, birth, disability, age or sexual
       orientation shall be prohibited.\n2. Within
       the scope of application of the Treaties and
       without prejudice to any of their specific
       provisions, any discrimination on grounds of
       nationality shall be prohibited."@en ;
    rdfs:label "Non-discrimination@en .
```

Listing 10: The RDF representation of the `EUFundamentalRight` class and an indication of an instance of this type, `RightNonDiscrimination`.

management practices required for high-risk AI systems, specifically AI Act Article 10(2) part (f). The integration of this into the design hence aligns with the research question.

5.5 Documenting test results on bias

To document test results on bias, the Evaluation and Report Language (EARL)[5] is adopted, hence aligning with the "ontology reuse" sub-activity of *Ontology implementation* stage of the Linked Open Terms (LOT) methodology (Poveda-Villalón et al. 2022). EARL is then extended with additional classes and properties relevant to the bias test documentation task. The reason that it was chosen to extend EARL as opposed to adopting specialised approaches, such as that proposed by Franklin et al. (2022), is for harmony in the AIRO ontology; using EARL, a more generic vocabulary, would facilitate other areas of concern in the risk management process to likewise use EARL to represent test results. This approach prioritises a shared common vocabulary, providing a small number of extensions as relevant to specific area of testing, where necessary. This is also discussed in Section 4.6. In this context, a number of extensions are developed for EARL for the representation of bias tests.

EARL was extended for a number of purposes, with new types represented on the `BiasAssertion` class, a subclass of `earl:Assertion`.

1. **Threshold representation:** Representing the threshold for a pass or fail outcome in this particular test.

2. **Description of groups:** A human-readable description of the groups being tested in the bias test.

3. **Fairness metric:** Represent the fairness metric used for the test.

4. **Mitigation measure:** Represent any mitigation measures applied after completing the test.

5. **Result URI:** A URI indicating where the results can be found on the web.

5.5.1 Test thresholds

When representing thresholds for bias tests, a number of new classes and properties were created. An initial consideration involved only **binary thresholds** (i.e., where a value below or above a particular value indicates

[5] The EARL schema can be found at the following link: `https://www.w3.org/TR/EARL10-Schema/`. Verified 15.08.2024.

passing/failing), but in the implementation of use-cases, it was found that this was initially insufficient for those fairness metrics which were represented, and another threshold type, a **range threshold** was hence considered in addition. Range thresholds, as defined here, specify a range of values which may be considered to "acceptable", with values which do not fall within this range being considered "unacceptable". The RDF structure of this in TURTLE format is indicated in Listing 11. The representation of threshold values as floating point numbers was inspired by their representation in the work of Franklin et al. (2022), but threshold values may also constitute integer values, so integer values may also be used to indicate thresholds.

In Listing 11, the `testThreshold` property connects a `BiasAssertion` to a `Threshold`. Regarding these thresholds, classes were designed (`Threshold` and its subclasses `ThresholdRange` and `ThresholdBinary`), with the properties (`thresholdBegin`, `thresholdEnd` and `binaryThreshold`) on those classes indicating the values associated with that threshold type.

Additionally, OWL cardinality constraints were created on properties of the `Threshold` class and its subclasses to indicate that subclasses of `Threshold` must have exactly one of each required property. The RDF representation of such cardinality constraints is indicated in Listing 12. Moreover, each bias assertion (instance of the `BiasAssertion` class) must indicate exactly one threshold.

5.5.2 Description of groups

Although the bias taxonomy developed describes categories of bias, it does not indicate those groups affected by a particular bias category. Such information is, of course, highly significant for this application, despite not being representable using EARL's existing properties and classes. Hence, a `groupDescription` property was also created, with `BiasAssertion` in its domain. As with thresholds, an OWL property constraint was created for the `groupDescription` predicate also, requiring that each `BiasAssertion` has exactly one `groupDescription`. The RDF representation of the `groupDescription` property is illustrated in Listing 13.

5.5.3 Fairness metrics

A set of fairness metrics were selected and connected to each `BiasAssertion` instance via an associated predicate. Therefore, any test which endeavours to uncover the presence of harmful biases which uses a single fairness metric can be represented as a single `BiasAssertion`.

In this case, the fairness metrics used were adapted into machine-readable representations from the IBM AI Fairness 360 (Bellamy et al. 2018) technical documentation. This approach was taken because of the very broad range

```
:Threshold rdf:type owl:Class .

:testThreshold rdf:type owl:ObjectProperty ;
    rdfs:domain :BiasAssertion ;
    rdfs:range :Threshold .

:ThresholdRange rdf:type owl:Class ;
    rdfs:subClassOf :Threshold .

:ThresholdBinary rdf:type owl:Class ;
    rdfs:subClassOf :Threshold .

:thresholdBegin rdf:type owl:DatatypeProperty ;
    rdfs:domain :ThresholdRange ;
    rdfs:range [
        rdf:type rdfs:Datatype ;
        owl:unionOf ( xsd:float xsd:integer )
    ] ;
    rdfs:comment "Indicates a float or integer
       beginning value for a range threshold"@en ;
    rdfs:label "threshold begin"@en .

:thresholdEnd rdf:type owl:DatatypeProperty ;
    rdfs:domain :ThresholdRange ;
    rdfs:range [
        rdf:type rdfs:Datatype ;
        owl:unionOf ( xsd:float xsd:integer )
    ] ;
    rdfs:comment "Indicates a float or integer
       ending value for a range threshold"@en ;
    rdfs:label "threshold end"@en .

:binaryThreshold rdf:type owl:DatatypeProperty ;
    rdfs:domain :ThresholdBinary ;
    rdfs:range [
        rdf:type rdfs:Datatype ;
        owl:unionOf ( xsd:float xsd:integer )
    ] ;
    rdfs:comment "Indicates a float or inteer value
       for the binary threshold"@en ;
    rdfs:label "binary threshold"@en .
```

Listing 11: Classes and properties/predicates related to test thresholds.

```
:ThresholdRange rdfs:subClassOf [
    rdf:type owl:Restriction ;
    owl:onProperty :thresholdBegin ;
    owl:cardinality "1"^^xsd:nonNegativeInteger
] ,
[
    rdf:type owl:Restriction ;
    owl:onProperty :thresholdEnd ;
    owl:cardinality "1"^^xsd:nonNegativeInteger
] .

:ThresholdBinary rdfs:subClassOf [
    rdf:type owl:Restriction ;
    owl:onProperty :binaryThreshold ;
    owl:cardinality "1"^^xsd:nonNegativeInteger
]
```

Listing 12: RDF representation of OWL cardinality constraints defined on classes and properties related to test thresholds.

```
:groupDescription rdf:type owl:DatatypeProperty ;
    rdfs:domain :BiasAssertion ;
    rdfs:range xsd:string .

:BiasAssertion rdfs:subClassOf [
    rdf:type owl:Restriction ;
    owl:onProperty :groupDescription ;
    owl:cardinality "1"^^xsd:nonNegativeInteger
] .
```

Listing 13: The RDF representation of the `groupDescription` property in TURTLE format, and its OWL cardinality restriction.

of fairness metrics implemented for the AI Fairness 360 library, quite often (although not always) featuring fairness metrics which correspond to that discussed literature analysed; for example, metrics such as *equalised odds difference* in AIF360 (represented in the `equalized_odds_difference()` method) measures the extent to which the "equalized odds" metric (as discussed in ISO/IEC 24027:2020) holds[6].

For the purpose of representing these fairness metrics in RDF, a new class, `FairnessMetric` was defined. Then, each fairness metric adapted from the AI Fairness 360 technical documentation was represented as an instance of this `FairnessMetric` class. An example of this in RDF is illustrated in Listing 14. The approach to represent fairness metrics as instances of a single class rather than in a richer taxonomical structure contrasts the approach taken to represent bias categories, but this approach was chosen for fairness metrics because fairness metrics do not typically differ in their generality, unlike bias categories, which are more conceptual; however, further classification into taxonomical structure based upon fairness categories, such as those outlined in Section 4.4.1, represents a potential area for future work.

From Listing 14, it is also clear that a `FairnessMetric` may be associated with a `BiasAssertion` using the `hasFairnessMetric` predicate. This design approach represents a single result on each `BiasAssertion`, and only a single `FairnessMetric` should be associated with any particular `BiasAssertion` instance. Hence, a cardinality constraint was imposed on the `hasFairnessMetric` property to ensure this structure; this can also be observed in Listing 14.

5.5.4 Bias mitigation measures

Representing mitigation measures is important in the context of testing for bias, and indicates that an entity is proactive with regard to adjusting its system to be more well-behaved with respect to bias. Such proactivity in screening for harmful biases is emphasised in the AI Act text, where

AIRO provides the `Control` class, which is suitable for representing risk mitigating actions, defined as "A measure that detects, mitigates, or eliminates an event"[7], with the event being a particular bias manifestation in this case.

[6] In ISO/IEC 24027:2020, "equalized odds" is described as implying that "True Positive Rates (TPR) are equal across demographic categories and False Positive Rates (FPR) are equal across demographic categories", while AIF360's "equalised odds difference" considers the "greater of the absolute difference in FPR and TPR for unprivileged and privileged groups" (Definition from `https://aif360.readthedocs.io/en/latest/modules/generated/aif360.metrics.ClassificationMetric.html`. Verified 15.08.2024.)

[7] `https://github.com/DelaramGlp/airo/blob/main/airo.owl`

```
:FairnessMetric rdf:type owl:Class ;
    terms:source "IBM AI Fairness 360 Glossary,
       https://aif360.res.ibm.com/resources#glossary"@en
       ;
    rdfs:comment "A quantification of unwanted bias
       in training data or models"@en ;
    rdfs:label "Fairness metric"@en .

:hasFairnessMetric rdf:type owl:ObjectProperty ;
    rdfs:domain :BiasAssertion ;
    rdfs:range :FairnessMetric ;
    rdfs:comment "Indicates that a test uses a
       particular fairness metric."@en ;
    rdfs:label "has fairness metric"@en .

:BiasAssertion rdfs:subClassOf [
    rdf:type owl:Restriction ;
    owl:onProperty :hasFairnessMetric ;
    owl:cardinality "1"^^xsd:nonNegativeInteger
] .

:ErrorRateRatio rdf:type :FairnessMetric ;
    terms:source "IBM AIF360 Python Docs,
       https://aif360.readthedocs.io/en/stable/modules/generated/aif36
       ;
    rdfs:comment "(False positives + False
       negatives) divided by (Positives +
       Negatives)"@en ;  # (FP+FN)/(P+N)
    rdfs:label "Error rate ratio"@en .
```

Listing 14: RDF representation of the `FairnessMetric` class, an example instance of this class, `ErrorRateRatio`, the `hasFairnessMetric` predicate, an OWL cardinality restriction on the `hasFairnessMetric` property for the `BiasAssertion` class, in TURTLE format.

In this design, it was decided to once again adopt those bias mitigation measures detailed in the AI Fairness 360 (AIF360) technical documentation. This approach was chosen primarily because of the broad range of bias mitigation measures exposed in this Python library, but also due to its natural correspondence with the fairness metrics adopted, both being from the AIF360 library. AIF360 is also discussed as an open-source tool available to audit for bias in the ISO/IEC 24027:2020 document.

Remaining consistent with AIRO, a `BiasMitigationMeasure` class is represented as a subclass of `airo:Control`. The `BiasMitigationMeasure` class definition is illustrated in Listing 15. As with the representation of fairness metrics, bias mitigation measures are represented on a shared stratum, with each being an instance of the `BiasMitigationMeasure` class. For example, Listing 16 shows the definition for the instance representing a disparate impact remover. Similar to test thresholds, group descriptions and fairness metrics, bias mitigation measure instances are connected to a particular bias assertion using a predicate. In this case, a `mitigationApplied` predicate was created for this purpose, with the bias assertion in its domain. The RDF definition of the `mitigationApplied` predicate is shown in Listing 17. As it is possible that multiple bias mitigation measures could be

```
:BiasMitigationMeasure rdf:type owl:Class ;
    rdfs:subClassOf airo:Control ;
    rdfs:comment "A mitigation measure for bias"@en ;
    rdfs:label "Bias mitigation measure"@en .
```

Listing 15: RDF representation of the `BiasMitigationMeasure` class, in TURTLE format.

```
:DisparateImpactRemover rdf:type
   :BiasMitigationMeasure ;
    terms:source "IBM AIF360 Python Docs,
       https://aif360.readthedocs.io/en/stable/modules/algorithms.html
       ;
    rdfs:comment "Disparate impact remover is a
       preprocessing technique that edits feature
       values increase group fairness while
       preserving rank-ordering within groups."@en ;
    rdfs:label "Disparate impact remover"@en .
```

Listing 16: Example RDF representation of the `DisparateImpactRemover` instance of the `BiasMitigationMeasure` class, in TURTLE format.

```
:mitigationApplied rdf:type owl:ObjectProperty ;
    rdfs:domain :BiasAssertion ;
    rdfs:range :BiasMitigationMeasure ;
    rdfs:comment "Indicates a mitigation measure
       applied to a bias assertion"@en ;
    rdfs:label "Mitigation applied"@en .
```

Listing 17: RDF representation of the `mitigationApplied` property, in TURTLE format.

applied to a single bias assertion, no cardinality constraint was applied to this predicate.

5.5.5 Result URI

A field is provided to indicate a URI corresponding to a particular result. This predicate was created because of its potential relevance in result verification; a result URI could conceivably provide proof that those results indicated in the knowledge graph correspond to results which have been verified elsewhere, for example as made public by a government agency or AI safety auditing company. The definition of this property in RDF is illustrated in Listing 18.

```
:testResultURI rdf:type owl:DatatypeProperty ;
    rdfs:domain :BiasAssertion ;
    rdfs:range xsd:anyURI ;
    rdfs:comment "URI pointing to the test result
       (e.g., GitHub link)."@en ;
    rdfs:label "Result URI" .
```

Listing 18: RDF representation of the `testResultURI` property, in TURTLE format.

5.5.6 Grouping bias assertions: `BiasAudit`

Beyond extensions made to EARL, a new class was created, `BiasAudit`, to group multiple bias tests under a shared “audit” umbrella. Despite being referred to as an audit, the grouping of a set of tests, each evaluating a result on a particular fairness metric, is an abstraction which allows for a number of testing conditions to be represented. For example, an individual bias test script may be represented as `BiasAudit` instance; bias tests frequently compute values over a number of metrics before concluding the presence or absence of a bias category, and the `BiasAudit` grouping allows for this. A new property, `interpretedBias` (with `BiasAudit` in its domain) aids in facilitating this case, allowing the tester to indicate their interpretation of the results from a suite of fairness metrics. The `BiasAudit` class definition, the `hasAssertion` property (connecting a bias audit with its constituent assertions) as well as the `interpretedBias` property defintions are illustrated in Listing 19.

```
:BiasAudit rdf:type owl:Class ;
    rdfs:subClassOf airo:Documentation ;
    rdfs:comment "An audit which focuses on
       measuring bias in the AI system"@en ;
    rdfs:label "Bias audit"@en .

:BiasAudit rdfs:subClassof [
    rdf;type owl:Restriction ;
    owl:onProperty :auditDate ;
    owl:cardinality "1"^^xsd:integer
] .

:hasAssertion rdf:type owl:ObjectProperty ;
    rdfs:domain :BiasAudit ;
    rdfs:range earl:Assertion ;
    rdfs:comment "Connects a bias audit with an
       assertion"@en ;

:interpretedBias rdf:type owl:ObjectProperty ;
    rdfs:domain :BiasAudit ;
    rdfs:range :Bias ;
    rdfs:comment "Indicates a bias type interpreted
       from a bias audit"@en ;
    rdfs:label "Interpreted bias"@en .
```

Listing 19: RDF representation of the `BiasAudit` class, the `hasAssertion` property, the `interpretedBias` property, as well as an OWL cardinality restriction concerned with the presence of an audit date.

5.6 Dataset descriptions

In Section 4.5.2, management and data governance practices for datasets, as specified by the EU AI Act for high-risk AI systems, are listed, and alignment with these data governance practices is a task outlined in the research question. As discussed before, *Datasheets for datasets* (Gebru et al. 2021) represent a common means of describing datasets in the ethical AI space, and such documents may provide information which is of relevance in AI risk management, including but not limited to risks related to harmful bias manifestations. Datasheets additionally serve to satisfy the requirements outlined in Section 4.5.2, which further lends to their importance in the context of the bias risk management framework extension to AIRO proposed in the present report.

A majority of the questions constituting a datasheet, as proposed by Gebru et al. (2021), ask for a boolean-type answer (yes / no), but it is often the case that more information (a description) is required, typically given that the answer was "yes". A simple representation would associate a string-type (`xsd:string`) with each field in a datasheet. However, this sacrifices machine-readability, not exploiting the full capabilities of knowledge graphs in the semantic representation of datasheet fields.

Hence, a majority of fields were divided into two more atomic components:

1. A **presence** component: a boolean-type field (`xsd:boolean`), indicating whether the answer is "yes" or "no".

2. A **description / other component** for further elaboration, if required: a string-type field, or other more machine-readable type where appropriate, to describe an answer, typically given that the associated presence component is **true** (indicated "yes").

This is implemented primarily using the Data Quality Vocabulary (DQV)[8] and the Data Catalog Vocabulary (DCAT)[9], where appropriate.

For the majority of fields, DQV is used, with two classes used in particular[10]:

1. `dqv:Dimension`: "A Quality Dimension (dqv:Dimension) is a quality-related characteristic of a dataset relevant to the consumer (e.g., the availability of a dataset)."

2. `dqv:Metric`: "Represents a standard to measure a quality dimension. An observation (instance of dqv:QualityMeasurement) assigns a value in a given unit to a Metric."

[8] The following link provides information about DQV: `https://www.w3.org/TR/vocab-dqv/`. Verified 15.08.2024.

[9] The following link provides information about DCAT: `https://www.w3.org/TR/vocab-dcat-2/`. Verified 15.08.2024.

[10] The source of each of these definitions can be found on the W3C webpage for DQV, at the following link: `https://www.w3.org/TR/vocab-dqv/`. Verified 29.07.2024.

3. `dqv:QualityMeasurement`: “Represents the evaluation of a given dataset (or dataset distribution) against a specific quality metric.”

Listing 20 indicates how the following field in a datasheet is represented in RDF, for a particular use-case[11], as an example:

> “How many instances are there in total (of each type, if appropriate)?”(Gebru et al. 2021)

```
:ImageInstanceCountMeasurementTraining rdf:type dqv:QualityMeasurement ;
    rdfs:comment "Count of image instances in the training dataset"@en ;
    dqv:computedOn :ChexpertDataset ;
    dqv:isMeasurementOf :InstanceCountMetric ;
    dqv:value "223414"^^xsd:nonNegativeInteger .
```

Listing 20: RDF representation of a quality measurement (instance of type `dqv:QualityMeasurement`), indicating the number of image instances in the training dataset, in TURTLE format.

The information represented in Listing 20 is an instance declaration, specifically, declaring an instance of type `dqv:QualityMeasurement`; in this case, a `dqv:QualityMeasurement` must be specified for each (sub)field of the datasheet[12] in order for a particular RDF representation to constitute a valid datasheet. `dqv:Dimension` and `dqv:Metric` classes are pre-defined (in the `datasheets-dcat-dqv` file of the source code). Each `dqv:Metric` is referred to by a particular `dqv:QualityMeasurement`, via the `dqv:isMeasurementOf` property. In the above example, `:InstanceCountMetric` and `:InstanceCountDimension` are defined as in Listing 21[13]. However, ensuring that a datasheet is correctly structured in RDF is difficult, given the number of fields which must be represented in a datasheet. Therefore, SPARQL constraints in SHACL were used to validate that a particular dataset (instance of `dcat:Dataset`) has fields which would constitute it being considered as having an associated datasheet. An example of such a constraint is indicated in Listing 22. Notably, this approach also accounts for dependencies between fields; if a particular field is answered “no”, then in some cases, there may be no need for some of the following fields to be completed.

[11] This example is taken from one of the use-cases discussed in the Evaluation section, for the *CheXpert* datasheet (Garbin et al. 2021).

[12] With a few exceptions, where new properties are defined. One such example is the `maintainer` property, a subproperty of `dcat:contactPoint`.

[13] These are defined in the `datasheets-dcat-dqv.ttl` file.

```
# Number of instances
:InstanceCountDimension rdf:type dqv:Dimension ;
    terms:source "Datasheets for datasets,
       https://arxiv.org/pdf/1803.09010."@en ;
    rdfs:label "Instance count"@en ;
    rdfs:comment "Counts of instances in the
       dataset"@en .

:InstanceCountMetric rdf:type dqv:Metric ;
    rdfs:comment "A metric to measure the instance
       count dimension of a dataset"@en ;
    rdfs:label "Instance count metric"@en ;
    dqv:inDimension :InstanceCountDimension .
```

Listing 21: RDF representation of a dimension and metric associated with the instance counts, `InstanceCountDimension` and `InstanceCountMetric`, in TURTLE format.

5.7 Overview

In Figure 5.2, the contributions proposed in this approach are summarised in a graph format. Figure 5.2 primarily serves to indicate the basic structure of the framework, as well as how it integrates with AIRO, and is not extensive in its description of subclasses and fields (neither subclasses of `:Bias` nor different datasheet fields are individually indicated in Figure 5.2, for example).

```
:DatasheetShape rdf:type sh:NodeShape ;
    sh:targetClass dcat:Dataset ;

    ################################
    # Has purpose
    ################################
    sh:sparql [
        sh:message "Dataset must have a purpose measurement with a string value"@en ;
        sh:prefixes [
            sh:declare [
                sh:prefix "rdf" ;
                sh:namespace "http://www.w3.org/1999/02/22-rdf-syntax-ns#"^^xsd:
            ] ;
            # ... other prefixes, withheld for brevity in report format.
        ] ;
        sh:select """
            SELECT $this
            WHERE {
                FILTER NOT EXISTS {
                    ?measurement dqv:computedOn $this ;
                        rdf:type dqv:QualityMeasurement ;
                        dqv:isMeasurementOf :PurposeMetric ;
                        dqv:value ?value .
                    FILTER(datatype(?value) = xsd:string)
                }
            }
        """
    ] ;
    # ... more constraints
```

Listing 22: A SPARQL constraint in SHACL which results in non-conformance if there is no “Purpose” `dqv:QualityMeasurement` defined on that dataset.

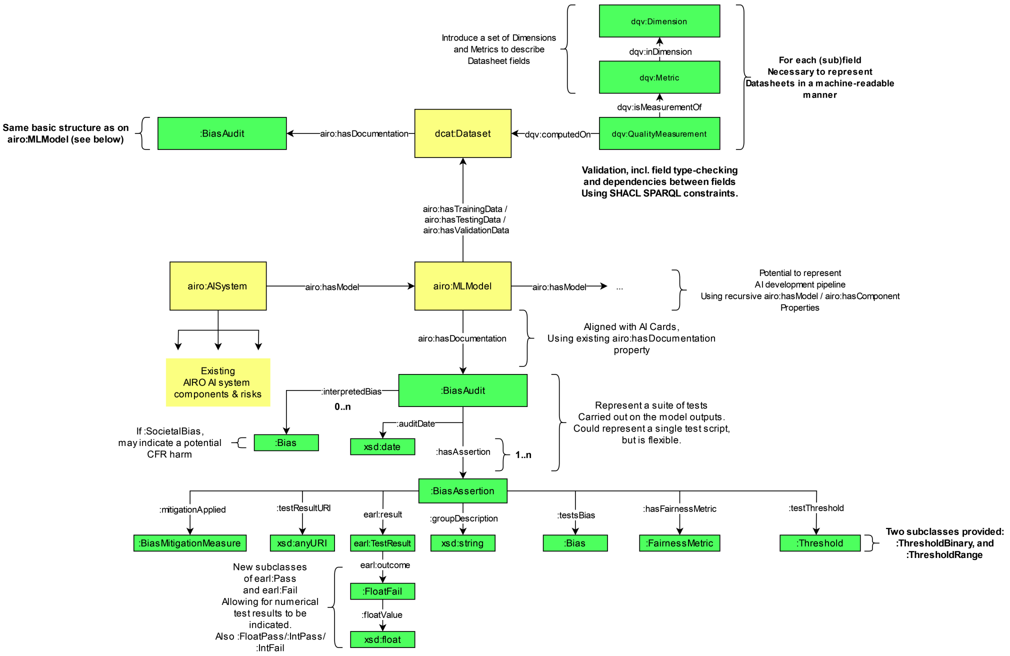


Figure 5.2: An overview of the contributions to AIRO proposed in this design section, intended to indicate basic structure of the framework as well as how it integrates with AIRO. Green colour coding indicates newly developed classes, existing classes which were incorporated or classes adapted from / extended from existing ontologies or vocabularies, such as EARL. Yellow colour coding indicates existing classes in AIRO.

Chapter 6

Evaluation

In this section, the ontological extensions described in the design section are evaluated through a variety of means, including through the implementation of use-cases and the use of tools for tasks such as RDF validation.

6.1 Ontology metrics

To compute metrics for the ontological extensions presented for AIRO, the existing RDF data in TURTLE format was split into files, with the following categories:

1. **Datasheets**: For RDF triples related to dataset documentation segment of the bias documentation framework, including neither use-cases nor any SHACL shapes.

2. **Bias taxonomy**: For RDF triples related to representing bias categories in a taxonomy.

3. **Fairness metrics**: For fairness metrics, including the `hasFairnessMetric` property.

4. **Mitigation measures**: For mitigation measures (instances of the `BiasMitigationMeasure` class), including the `mitigationApplied` property.

5. **Fundamental rights**: For fundamental rights from the EU Charter of Fundamental Rights (CFR), including the `potentiallyViolates` property mappings from bias categories onto potential CFR violations.

6. **Test documentation (misc.)**: Properties and classes required for representing test documentation, but which was not represented in another category, e.g. the `:FloatFail` class, the `groupDescription` property.

7. **Total**: All of the above categories, represented in a single file (in the source code, this file results from merging `main.ttl` and `datasets-dcat-dqv.ttl`).

Using *Protégé* (Musen 2015), each of these files, corresponding to the categories listed above, were loaded. Tables 6.1, 6.2 and 6.3 show the results from the Protégé *Ontology metrics* window. Note that the **Total** counts do not always represent a sum of other category counts, as there is overlap in classes referenced in each category, despite efforts to create independent segments.

	Axiom	**Logical axiom count**	**Declarations axiom count**
Datasheets	810	201	3
Bias taxonomy	427	91	64
Fairness metrics	153	38	2
Mitigation measures	62	17	2
Fundamental rights	244	90	3
Test documentation	110	46	21
Total	**1805**	**483**	**94**

Table 6.1: Ontology metrics from Protégé (Musen 2015). Axiom, logical axiom count and declarations axiom count.

	Class count	**Object property count**	**Data property count**
Datasheets	3	3	0
Bias taxonomy	65	0	0
Fairness metrics	2	1	0
Mitigation measures	4	1	0
Fundamental rights	2	1	0
Test documentation	16	7	8
Total	**86**	**13**	**8**

Table 6.2: Ontology metrics from Protégé (Musen 2015). Class count, object property count and data property count.

	Individual count	**Annotation property count**
Datasheets	199	5
Bias taxonomy	30	4
Fairness metrics	36	3
Mitigation measures	13	3
Fundamental rights	61	4
Test documentation	0	3
Total	**328**	**6**

Table 6.3: Ontology metrics from Protégé (Musen 2015). Individual count and annotation property count.

6.2 Adherence to Linked Open Terms (LOT) methodology

The Linked Open Terms (LOT) methodology (Poveda-Villalón et al. 2022) was used to develop the ontology (intended as ontological extensions to AIRO (Golpayegani et al. 2022)) presented in this report, particularly with reference to the **Ontology requirements specification**, **Ontology implementation** and **Ontology publication** activities.

6.2.1 Ontology requirements specification

Requirements for the ontology, as well as associated competency questions (activities associated with the *Ontology requirements specification* activity of the LOT methodology (Poveda-Villalón et al. 2022)) are outlined in Section 6.6.

6.2.2 Ontology implementation

When developing the ontology based upon the requirements and competency questions specified, the following tasks were carried out, aligned with the LOT methdology (Poveda-Villalón et al. 2022), as follows:

1. **Conceptualisation:** In the design process, diagrams were created a various stages to conceptualise the ontology.

2. **Ontology reuse:** Where possible, existing ontologies and vocabularies were reused. For example, EARL is used in the test reporting section of the framework, and DQV and DCAT are used to represent datasheet fields.

3. **Encoding:** Ontology code was written in the OWL language, including metadata.

4. **Evaluation:** The ontology was frequently evaluated for syntactic, semantic or other errors (see Section 6.3 and Section 6.4), and it was ensured that it conformed to requirements (see Section 6.6).

6.2.3 Ontology publication

The **Ontology publication** activity of the LOT methodology (Poveda-Villalón et al. 2022) is in progress as of the time of writing, with the following sub-activites:

1. **Documentation**: **WIDOCO** (Garijo 2017) was used to generate human-readable ontology documentation, and the associated files are available in the `AIRO-Bias-Doc` folder of the source code repository on GitHub[1].

2. **Publication**: The ontology is available in a public GitHub repository, available at the following link: `https://github.com/drd00/msc-dissertation-files`. However, it is not yet available via a public namespace URI (using w3id or similar).

6.3 RDF validation

The *W3C RDF Validation Service*[2] was used to check for syntax errors. Because the W3C RDF Validation Service accepts RDF/XML formatted documents, and the RDF written was in this case formatted in TURTLE (Terse RDF triple language) formatted `.ttl` files, the *EasyRdf Converter*[3] was used to convert the TURTLE syntax to XML for the purpose of RDF validation using the W3C RDF Validation Service.

The process of converting TURTLE syntax to XML and feeding the output to the W3C RDF Validation Service and subsequently fixing any errors outlined was repeated throughout the design process to prevent the accumulation of syntax errors. This process is aligned with the Linked Open Terms (LOT) methodology (Poveda-Villalón et al. 2022) in the *evaluation* sub-activity of its *Ontology implementation* process; as a part of this *evaluation* sub-activity, the ontology developers "guarantee that the ontology does not have syntactic, modelling or semantic errors" (Poveda-Villalón et al. 2022).

The final state of all `.ttl` files associated with this project have been verified as valid RDF using the W3C RDF Validation Service.

6.4 Reasoners

Reasoners are powerful tools which can be used to aid in ontological development and evaluation, and can be used to uncover properties of an ontology which may not have been apparent to the individual or group that designed the ontology. Among other flaws, reasoners can discover logical

[1] This can be accessed at the following link: `https://github.com/drd00/msc-dissertation-files/tree/master/AIRO-Bias-Doc`. Verified 16.08.2024.

[2] The W3C RDF Validation Service is available at the following link: `https://www.w3.org/RDF/Validator/`. Last verified 26.07.2024.

[3] The EasyRdf Converter is available at the following link: `https://www.easyrdf.org/converter`. Last verified 26.07.2024.

inconsistencies, erroneous entailments, reveal relationships between classes which were not immediately apparent, and much more.

The HermiT reasoner (Glimm et al. 2014) was loaded as a plugin in the *Protégé* software application (Musen 2015)[4] and it was used at various points in the development process. As with RDF validation, this process is aligned with the *evaluation* sub-activity of the *Ontology implementation* process within the LOT methodology (Poveda-Villalón et al. 2022).

As a step in this evaluation, each of the files associated with ontological structure, `main.ttl` and `datasets-dcat-dqv.ttl`, were verified as consistent and do not cause HermiT to produce any reasoning errors in their completed state, as of the time of writing.

6.5 Use cases

In this section, it is discussed how a number of use-cases were represented to demonstrate the effectiveness of the proposed open ontology (AIRO ontological extensions). The implementation of such use-cases in RDF is intended to indicate the capability of the RDF representation of *Datasheets for datasets* (Gebru et al. 2021).

6.5.1 Datasheets

To evaluate the semantic representation of *Datasheets* (Gebru et al. 2021) in RDF, two datasheets, that for the *CheXpert* (Garbin et al. 2021) and that for the *Movie review polarity* dataset (provided by Gebru et al. (2021) as part of the Datasheets for datasets paper) were modelled.

The implementation of the *CheXpert* datasheet (Garbin et al. 2021) as well as the *Movie review polarity* datasheet (Gebru et al. 2021) were successful insofar as in each case, each datasheet field could (with some caveats, as discussed in the following paragraph) be represented in RDF using DQV and DCAT.

Difficulties which were encountered required, in some cases, the use of placeholder values or the attachment of comments (using the `rdfs:comment` predicate). The datasheet fields where this was necessary sometimes differed between the two use-cases represented. The problems encountered for each of the use-cases are as follows:

1. **Ambiguity or lack of a direct response**: A number of fields in each use-case did not directly answer the question asked in the datasheet. An example of this was evident in the modelling of the

[4] HermiT 1.4.3.456 is included in the standard Protégé distribution for Windows, version 5.6.4.

CheXpert datasheet (Garbin et al. 2021) use-case, where the following question is not directly answered:

> "If the dataset is a sample, then what is the larger set? Is the sample representative of the larger set?" (Gebru et al. 2021)

The response to this provided in the *CheXpert datasheet* (Garbin et al. 2021) is as follows:

> "The dataset contains studies from patients that visited the inpatient and outpatient centers of the Stanford Hospital between October 2002 and July 2017 and had a chest X-ray taken." (Garbin et al. 2021)

Such cases can be attributed to deficiencies either in the format of the datasheet questions or responses rather than in the RDF representation of the datasheet itself, where direct responses are always expected in order for a mapping onto a more semantically rich format. Suggestions for future work to prevent such issues discussed in Section 7.3.5.

2. **References to figures, tables, bibliography entries**: A number of fields referenced figures, tables or bibliography entries present in the datasheet PDF. However, this was not foreseen in the mapping of datasheets onto an RDF representation.

   In the *CheXpert* datasheet (Garbin et al. 2021), this was the case for a field asking whether a label or target is associated with each instance, where both tables and figures were referenced in the text response. For the *Movie review polarity* datasheet (Gebru et al. 2021), this was not an issue.

3. **N/A fields, false assumptions:** Fields such as "When will the dataset be distributed?" (Gebru et al. 2021) assume that the dataset has not yet been distributed. In the *CheXpert use-case*, for example, the dataset had already been distributed, and the response was hence "The dataset is currently available" (Garbin et al. 2021) — a string (`xsd:string`) response which does not conform to the expected `xsd:date` type in the SHACL shape used for validation. This is likewise the case for the *Movie review polarity use-case*, although a year was provided in this case: "The dataset was first released in 2002" (Gebru et al. 2021).

   A similar pattern, whereby it was noted that questions may make false assumptions (about previous responses in this case), was also observed in the *Movie review polarity* dataset (Gebru et al. 2021), whereby a number of responses were "N/A"; in particular, this was the case with fields pertaining to consent and its impact on data subjects, after it was indicated that the data was crawled from public sources.

A number of areas of future work which target these issues in representing real-world datasheet use-cases are indicated in Section 7.3.5.

6.5.2 Bias test reports

To evaluate the semantic representation of bias tests in RDF using the bias taxonomy developed from ISO/IEC 24027:2020, GDPR Article 9 (as well as gender bias) and the AI Ontology (Joachimiak et al. 2024) (by proxy, bias terms from the previously discussed NIST technical report (Schwartz et al. 2022)), two real-world use-cases were implemented on the *UCI Adult* dataset (Asuncion et al. 2007) and using metrics from the AI Fairness 360 (AIF360) *Medical Expenditure Tutorial*[5]. Specifically, one use-case which requires predictions (hence represented as a `BiasAudit` on the machine learning model directly) was implemented, and one use-case which does not require predictions (and is hence represented as a `BiasAudit` on the dataset directly) was implemented. The code necessary to perform these tests was written in Python using `scikit-learn` (Pedregosa et al. 2011) and `pandas` (pandas development team 2020, Wes McKinney 2010).

In the AIF360 Medical Expenditure Tutorial, the following fairness metrics are used:

1. **Disparate impact**[6]: without predictions (represented on the dataset use-case),

2. **Average odds difference**[7]: with predictions (represented on the model use-case),

3. **Statistical parity difference**[8]: with or without predictions (represented on both the model and the dataset use-cases),

[5] GitHub link to the AIF360 Medical Expenditure Tutorial: `https://github.com/Trusted-AI/AIF360/blob/main/examples/tutorial_medical_expenditure.ipynb`. Verified 15.08.2024.

[6] `https://aif360.readthedocs.io/en/stable/modules/generated/aif360.metrics.BinaryLabelDatasetMetric.html#aif360.metrics.BinaryLabelDatasetMetric.disparate_impact`. Verified 15.08.2024.

[7] `https://aif360.readthedocs.io/en/stable/modules/generated/aif360.metrics.ClassificationMetric.html#aif360.metrics.ClassificationMetric.average_odds_difference`. Verified 15.08.2024.

[8] `https://aif360.readthedocs.io/en/stable/modules/generated/aif360.metrics.ClassificationMetric.html#aif360.metrics.ClassificationMetric.statistical_parity_difference`. Verified 15.08.2024.

	Value
Disparate impact	0.3597
Statistical parity difference	0.1945
Theil index	0.0838

Table 6.4: Fairness metrics calculated on the *UCI Adult* dataset (Asuncion et al. 2007) (no predictions)

4. **Equal opportunity difference**[9]: with predictions (represented on the model use-case),

5. **Theil index**[10]: without predictions (represented on the dataset use-case).

Use-case 1: Dataset (no predictions)

In Table 6.4, the results from the Python script calculating fairness measures associated with the use-case represented only on a dataset (with no predictions, represented as a `BiasAudit` on the dataset) are shown. Thresholds for each of these metrics were selected manually (no "reasonable" thresholds were suggested in the AIF360 Medical Expenditure Tutorial), with the threshold type aligning with the fairness metric itself, and the test pass/fail status depending on the relationship between the obtained value (from Table 6.4) and the threshold specification. For example, the Theil index will always be positive when calculated over income values, with greater values indicating more significant inequality between groups, and a binary threshold (indicating a value at or above which the test may be considered to have failed) is appropriate, whereas for average odds difference, specifying a range of values considered reasonably "fair" (from a small negative value to a small positive value) is more appropriate. Listing 23 illustrates how the bias audit was indicated on the UCI Adult dataset (Asuncion et al. 2007) for this use-case, indicating also the definition of one of the bias assertions, `DisparateImpactAssertion`. Using the bias test reporting approach proposed in this project, this use-case was successfully represented in RDF format. The implementation of this use-case in RDF, in TURTLE format, is available in the GitHub repository associated with this project[11].

[9] `https://aif360.readthedocs.io/en/stable/modules/generated/aif360.metrics.ClassificationMetric.html#aif360.metrics.ClassificationMetric.equal_opportunity_difference`. Verified 15.08.2024.

[10] `https://aif360.readthedocs.io/en/stable/modules/generated/aif360.metrics.ClassificationMetric.html#aif360.metrics.ClassificationMetric.theil_index`. Verified 15.08.2024.

[11] Link to the file containing both use-cases in RDF format: `https://github.com/drd00/msc-dissertation-files/blob/master/bias-tests/`

```
:AdultDatasetTrainingData rdf:type dcat:Dataset ;
    rdfs:comment "Training data for the Adult dataset"@en ;
    terms:creator "Barry Becker"^^xsd:string .

:AdultDatasetBiasAuditDS rdf:type :BiasAudit ;
    :auditDate "2024-07-04"^^xsd:date .

:AdultDatasetTrainingData airo:hasDocumentation :AdultDatasetBiasAuditDS .

:DisparateImpactAssertion rdf:type :BiasAssertion ;
    :groupDescription "Comparison of positive outcomes rates between females and males for income prediction" ;
    :hasFairnessMetric :DisparateImpact ;
    :testsBias :GenderBias ;
    :testThreshold [
        rdf:type :ThresholdRange ;
        :thresholdBegin "0.8"^^xsd:float ;
        :thresholdEnd "1.2"^^xsd:float
    ] ;
    earl:result [
        rdf:type earl:TestResult ;
        earl:outcome [
            rdf:type :FloatFail ;
            :floatValue "0.3597"^^xsd:float
        ]
    ] ;
    :testResultURI "https://example.org/results/disparate_impact_
        result_for_dataset"^^xsd:anyURI .

# Other bias assertions ...

:AdultDatasetBiasAuditDS :hasAssertion :DisparateImpactAssertion , :StatisticalParityDifferenceAssertion , TheilIndexAssertion .
```

Listing 23: RDF representation of the `BiasAudit` for the UCI Adult dataset (Asuncion et al. 2007) as well as a disparate impact assertion which is a component of this bias audit.

	Value
Statistical parity difference	-0.1104
Equal opportunity difference	0.0910
Average odds difference	0.0786

Table 6.5: Fairness metrics calculated on the *UCI Adult* dataset (Asuncion et al. 2007) (using predictions)

Use-case 2: Model (using predictions)

In Table 6.5, the results from the Python script calculating fairness measures associated with the use-case utilising a logistic regression model (represented as a `BiasAudit` on the model) are shown. As in the previous use-case, which considered a `BiasAudit` directly on the dataset (where the fairness metrics did not require predictions), thresholds were manually selected, and the pass/fail status of each assertion set based on the placement of the value in Table 6.5 relative to these thresholds. Like in the previous section, Using the bias test reporting approach proposed in this project, this use-case was successfully represented in RDF in a TURTLE format. The implementation of this use-case in RDF, in TURTLE format, is available in the GitHub repository associated with this project.

`adult-dataset/use-case-adult.ttl`. Verified 15.08.2024.

```
:AdultIncomeClassifier rdf:type airo:AISystem ;
    airo:hasModel :LogisticRegressionModel .

:AdultDatasetBiasAudit rdf:type :BiasAudit ;
    :auditDate "2024-08-04"^^xsd:date .

:LogisticRegressionModel rdf:type airo:MLModel ;
    airo:hasPurpose [
        rdf:type airo:Purpose ;
        rdfs:comment "Predict whether an
           individual's income exceeds \$50K/year
           based on census data"
    ] ;
    airo:hasDocumentation :AdultDatasetBiasAudit .

:EqualOpportunityAssertion rdf:type :BiasAssertion ;
    :groupDescription "Comparison between male and
       female groups" ;
    :hasFairnessMetric :EqualOpportunityDifference ;
    :testThreshold [
        rdf:type :ThresholdRange ;
        :thresholdBegin "-0.1"^^xsd:float ;
        :thresholdEnd "0.1"^^xsd:float
    ] ;
    :testsBias :GenderBias ;
    earl:result [
        rdf:type earl:TestResult ;
        earl:outcome [
            rdf:type :FloatPass ;
            :floatValue "0.0910"^^xsd:float
        ]
    ] ;
    :testResultURI "https://example.org/results/
        equal_opportunity_for_this_model"^^xsd:anyURI
           .

# Other bias assertions ...

:AdultDatasetBiasAudit :hasAssertion
   :StatisticalParityAssertion ,
   :EqualOpportunityAssertion ,
   :AverageOddsAssertion .

:AdultDatasetBiasAudit :interpretedBias :GenderBias .
```

Listing 24: RDF representation of the `BiasAudit` for the UCI Adult dataset (Asuncion et al. 2007) as well as a equal opportunity difference assertion which is a component of this bias audit. This use-case also illustrates the use of the `interpretedBias` predicate.

6.6 Requirements & Competency questions

For the evaluation of competency questions, the **GraphDB** triplestore was used, loading the triples from both use-cases in Section 6.5.2 as well as all of the extensions to AIRO presented[12].

Firstly, the following competency questions were chosen based on the following requirements:

1. **Indicate where bias audits have been conducted, and if so, which bias category was tested for, and what its result was.**

   The identification of biases, including tests and indication as to whether they failed or passed, is fundamental to this framework for bias, facilitating enhanced risk management.

2. **Indicate potential harms to individual fundamental rights, based on all of the bias documentation indicated in the knowledge graph.**

   This task is fundamental to aiding in EU AI Act compliance practices, as harms to fundamental rights are a particular concern in the AI Act.

3. **Interaction between bias tests and datasheets.**

   This task indicates the usefulness of providing necessary dataset documentation for risk management (in this case, the RDF representation of *Datasheets for datasets* (Gebru et al. 2021)) as a component of a framework alongside bias tests.

Each of the following competency questions were written with consideration of the above requirements.

> **Competency question 1:** Are there any bias audits which have been conducted anywhere in the AI development pipeline?

The SPARQL query written for Competency question 1 is illustrated in Listing 25. Listing 25 is a simple SPARQL `ASK` query, which returns a boolean **true** or **false** value depending on the presence of absence of at least one triple which satisfies the pattern, respectively. This query returns all `BiasAudit` instances in the graph, regardless of where, satisfying the requirement outlined in Competency question 1.

[12] The file which was used to evaluate competency questions is in the project GitHub repository, and is available at the following link: `https://github.com/drd00/msc-dissertation-files/blob/master/sparql-competency-qs/playground.ttl`. Verified 15.08.2024.

```
PREFIX airo: <https://w3id.org/airo#>
PREFIX rdf: <
    http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX : <https://example.org/ex#>

ASK WHERE {
    ?doc rdf:type :BiasAudit .
}
```

Listing 25: SPARQL query used to evaluate Competency question 1.

> **Competency question 2:** Are there any bias audits where a test suite indicated racial bias (not just individual fairness metrics)?

The SPARQL query written for Competency question 2 is illustrated in Listing 26. Listing 26 is another example of a SPARQL `ASK` query, and hence simply returns true or false, depending on whether there exists a bias audit anywhere with `interpretedBias` being `RacialBias`, indicating that a tester deemed the test results to indicate the presence of racial bias. This utilises the bias taxonomy to find a particular bias category of interest, indicating its potential relevance to the downstream risk management task for an AI system.

```
PREFIX airo: <https://w3id.org/airo#>
PREFIX rdf: <
    http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX : <https://example.org/ex#>

ASK WHERE {
    ?doc rdf:type :BiasAudit .
    ?doc :interpetedBias :RacialBias .
}
```

Listing 26: SPARQL query used to evaluate Competency question 2.

> **Competency question 3:** Can you list all biases entailed by failed assertions, at any area of the AI development pipeline?

The SPARQL query written for Competency question 3 is illustrated in Listing 27. The SPARQL query illustrated in Listing 27 is somewhat more

```
PREFIX airo: <https://w3id.org/airo#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX earl: <http://www.w3.org/ns/earl#>
PREFIX : <https://example.org/ex#>

SELECT DISTINCT ?biastype WHERE {
    ?doc rdf:type :BiasAudit .
    ?doc :hasAssertion ?biasassertion .
    ?biasassertion earl:result ?testresult .
    ?testresult earl:outcome ?testoutcome .
    ?testoutcome rdf:type ?type .
    FILTER(?type IN (:FloatFail , :IntFail , earl:Fail))
    ?biasassertion :testsBias ?biastype
}
```

Listing 27: SPARQL query used to evaluate Competency question 3.

complex. This query makes use of the `DISTINCT` keyword (identical to its utility in SQL) to get a unique list of bias types implied by failed bias assertions (i.e., tests which utilise a particular fairness metric). It filters results by the outcome of the test, only returning biases entailed from failed bias assertions (where the test outcome was one of `FloatFail`, `IntFail` or `earl:Fail`). This is aligned with the first requirement outlined in the beginning of this section, where one must have access to bias tests and their results.

> **Competency question 4:** Can you provide a unique list of all potential fundamental rights violations arising from bias in the AI system?

The SPARQL query written for Competency question 4 is illustrated in Listing 28. The SPARQL query illustrated in Listing 28 once more uses the `DISTINCT` keyword, in this case returning a unique lists of potential violations, based upon mappings provided between societal bias categories and EU fundamental rights (indicating potential harms to these fundamental rights entailed from the bias result). In this case, the `UNION` keyword is also used, returning the union of the results from two queries.

The first clause in Listing 28 is rather similar in content to Competency question 3 (as in Listing 27), albeit with additional statements to specify that the bias result must be a societal bias, binding the potential violations entailed from this to the `?potentialViolation` variable. The second clause

```
PREFIX airo: <https://w3id.org/airo#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX : <https://example.org/ex#>
PREFIX dcat: <http://www.w3.org/ns/dcat#>
PREFIX earl: <http://www.w3.org/ns/earl#>

SELECT DISTINCT ?potentialViolation
WHERE {
    {
        ?doc rdf:type :BiasAudit .
        ?doc :hasAssertion ?biasassertion .
        ?biasassertion :testsBias ?bias .
        ?biasassertion earl:result ?testresult .
        ?testresult earl:outcome ?testoutcome .
        ?testoutcome rdf:type ?type .
        FILTER(?type IN (:FloatFail , :IntFail , earl:Fail))
        ?bias rdf:type :SocietalBias .
        ?bias :potentiallyViolates ?potentialViolation .
    }
    UNION {
        ?doc rdf:type :BiasAudit .
        ?doc :interpretedBias ?bias .
        ?bias rdf:type :SocietalBias .
        ?bias :potentiallyViolates ?potentialViolation .
    }
}
```

Listing 28: SPARQL query used to evaluate Competency question 4.

in Listing 28 considers the `interpretedBias` predicate on the `BiasAudit` class.

> **Competency question 5:** If a particular dataset has failed bias assertions, can you return all datasets created by the same entity?

The SPARQL query written for Competency question 5 is illustrated in Listing 29. The SPARQL query illustrated in Listing 29 returns multiple columns, using the SPARQL `GROUP BY` and `GROUP_CONCAT` features. The first column indicates a dataset with a failed bias assertion, while the second column provides a comma-separated list of datasets created by the same author. This indicates how having relevant properties on the dataset, as required in the RDF implementation of *Datasheets for datasets* (Gebru et al. 2021), can be useful in risk management pertaining to harmful biases.

> **Competency question 6:** If a particular dataset has failed bias assertions, can you return whether the dataset creation was funded?

The SPARQL query written for Competency question 6 is illustrated in Listing 30. The SPARQL query illustrated in Listing 30 serves once more to indicate how representing *Datasheets for datasets* (Gebru et al. 2021) on a dataset in RDF format is useful in a bias risk management context. In this example, a unique list of datasets with failed bias assertions are queried for, with the second column indicating whether the creation of this dataset was funded, as indicated by a DQV `QualityMeasurement`, as required in a dataset's datasheet.

```
PREFIX dcat: <http://www.w3.org/ns/dcat#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX airo: <https://w3id.org/airo#>
PREFIX earl: <http://www.w3.org/ns/earl#>
PREFIX dqv: <http://www.w3.org/ns/dqv#>
PREFIX terms: <http://purl.org/dc/terms/>

PREFIX : <https://example.org/ex#>

SELECT ?dataset (GROUP_CONCAT(DISTINCT ?relatedDataset; separator=", ") AS ?relatedDatasets)
WHERE {
    {
        SELECT DISTINCT ?dataset ?creator
        WHERE {
            ?dataset rdf:type dcat:Dataset ;
                airo:hasDocumentation ?doc .
            ?doc rdf:type :BiasAudit ;
                :hasAssertion ?biasassertion .
            ?biasassertion earl:result ?testresult .
            ?testresult earl:outcome ?testoutcome .
            ?testoutcome rdf:type ?type .
            FILTER (?type IN (:FloatFail , :IntFail , earl:Fail))

            ?dataset terms:creator ?creator .
        }
    }

    ?relatedDataset rdf:type dcat:Dataset ;
        terms:creator ?creator .
}
GROUP BY ?dataset
```

Listing 29: SPARQL query used to evaluate Competency question 5.

```
PREFIX dcat: <http://www.w3.org/ns/dcat#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX airo: <https://w3id.org/airo#>
PREFIX earl: <http://www.w3.org/ns/earl#>
PREFIX dqv: <http://www.w3.org/ns/dqv#>
PREFIX terms: <http://purl.org/dc/terms/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema>

PREFIX : <https://example.org/ex#>

SELECT DISTINCT ?dataset ?wasFunded
WHERE {
    {
        ?dataset rdf:type dcat:Dataset .
        ?dataset airo:hasDocumentation ?doc .
        ?doc rdf:type :BiasAudit .
        ?doc :hasAssertion ?biasassertion .
        ?biasassertion earl:result ?testresult .
        ?testresult earl:outcome ?testoutcome .
        ?testoutcome rdf:type ?type .
        FILTER (?type IN (:FloatFail, :IntFail, earl:Fail))

        ?fundingMeasurement dqv:computedOn ?dataset ;
            rdf:type dqv:QualityMeasurement ;
            dqv:isMeasurementOf :FundingPresenceMetric ;
            dqv:value ?wasFunded .
        FILTER(datatype(?wasFunded) = xsd:boolean)
    }
}
```

Listing 30: SPARQL query used to evaluate Competency question 6.

6.7 AI Act coverage

In consideration of the research question, it is important that the presented contributions are contextualised in terms of the requirements of the EU AI Act.

1. The machine-readable representation of *Datasheets for datasets* (Gebru et al. 2021) is aligned with AI Act **Article 10(2), part (b)**, **Article 10(2), part (c)**, **Article 10(2), part (d)** and **Article 10(2), part (e)**, **Article 10(3)**.

   It also provides technical documentation required to fulfill the requirement set forth in **Annex IV(2) part (d)** and **Annex XI(2) part (c)** (for general-purpose models).

2. The consideration of machine learning model development pipelines may aid in the technical documentation requirement specified in **Annex IV(2) part (a)** and to some extent **Annex IV(2) part (c)**.

3. Representing bias test reports is aligned with the technical documentation requirement described in AI Act **Annex IV(2) part (g)**, **Article 10(2) part (f)** and **Article 10(2) part (g)**.

These contributions hence align with the research question, providing means of dataset documentation and bias test reporting which align with the data governance practices outlined in Article 10 of the AI Act and also aiding in the fulfillment of technical documentation requirements outlined in Annex IV and Annex XI of the AI Act.

Chapter 7

Discussion & Conclusions

7.1 Summary and interpretation of findings & limitations

In this report, a novel bias framework intended to be integrated with the AI Risk Ontology (AIRO) (Golpayegani et al. 2022) was proposed. In this section, the contributions as well as associated results and findings from the Evaluation sections are summarised, interpreted and discussed.

A novel bias taxonomy

A bias taxonomy was developed, aligned the ISO/IEC 24027:2020 document, and expanded upon. Extensions included categories from the AI Ontology (Joachimiak et al. 2024) and by proxy a NIST technical report on identifying and managing bias in AI (Schwartz et al. 2022) and GDPR Article 9 (and, additionally, gender bias). The bias taxonomy developed addresses the requirement for an RDF bias taxonomy with its core structure aligned with ISO/IEC 24027:2020, a gap in the state of the art. This approach is aligned with the research question and integrates well with AIRO, given its alignment with ISO standards and the AI Act.

For the use-cases modelled in the Evaluation chapter, the developed taxonomy was shown to be sufficient, notably providing a category corresponding to gender bias, as required in these cases. The extension of the ISO/IEC 24027:2020 derived taxonomy with categories from the AI Ontology provides a significant increase in expressivity while remaining aligned with ISO/IEC 24027:2020 in much of its core structure.

However, despite its success in the use-cases which were modelled, the proposed bias taxonomy may be improved upon further with further extensions. Other use-cases may have required a bias category which was not represented in the proposed bias taxonomy. Notably, societal categories such as bias related to income, social standing, etc., are not yet represented in this taxonomy, and these are particularly relevant given their mappings onto potential fundamental right violations.

Representation of bias tests using (extended) EARL, fairness metrics

A core component of the proposed bias framework facilitates the representation of bias tests in a machine-readable format. While work such as that of Franklin et al. (2022) in the *Fairness Metrics Ontology* have proposed means of representing fairness tests, a novel approach proposed in the present report instead considers further testing applications in AIRO (Golpayegani et al. 2022), utilising and providing expansions, where appropriate, to the Evaluation and Report Language (EARL). EARL is adopted primarily due to its more generic nature in comparison with approaches such as that proposed by Franklin et al. (2022), allowing for bias tests to be represented using only a small set of extensions specific to the area of testing. In the context of an application within AIRO, this is desirable, as providing a common core to represent test documentation can be beneficial, for example, for increased simplicity in SPARQL queries. A significant component of this test documentation approach involved the development of an appropriate set of fairness metrics to describe a particular atomic test's procedure, or "bias assertions". It is proposed that such bias assertions may be grouped and collectively referred to as a "bias audit", allowing for the representation of bias test scripts, for example. It was discussed in the Evaluation chapter how this aligns with the research question.

In the Evaluation chapter, use-cases were successfully modelled in RDF using the proposed structure for representing bias test documentation, notably allowing for bias test scripts to be represented both on the dataset (not requiring predictions) as well as the machine learning model directly (requiring predictions). This indicates the viability of the proposed RDF representation of bias tests on a range of fairness metrics which are included in the AI Fairness 360 (AIF360) software package (Bellamy et al. 2018). While the fairness metrics included in AIF360 are quite extensive, it would be beneficial for a greater number of use-cases to be modelled using the proposed bias test documentation approach, so that any significant deficiencies are highlighted, and the set of fairness metrics may be expanded upon accordingly.

Machine-readable representation of Datasheets for datasets

A significant contribution presented in the present report is the representation of Datasheets for datasets (Gebru et al. 2021) in a machine readable format — a novel contribution to the state of the art. To facilitate this, existing vocabularies, the Data Catalog Vocabulary (DCAT) and the Data

Quality Vocabulary (DQV), were used to represent fields (and subfields, as appropriate for enhanced machine-readability) of such datasheets. It was discussed in the Evaluation chapter how this aligns with the research question.

Because of the number of instances and properties required to constitute such a machine-readable datasheet representation, a shape was written using the Shapes Constraint Language (SHACL) to validate whether a particular dataset has sufficient information associated with it to constitute having an associated datasheet. This involved the use of SPARQL-based constraints for SHACL. This approach also facilitates representing only those fields which are appropriate; if the answer to a previous field is "no", then a following field may not be required, and hence need not be represented at all in the knowledge graph.

The two datasheet use-cases, the CheXpert datasheet (Garbin et al. 2021) and the Movie polarity datasheet (Gebru et al. 2021) were modelled using the proposed approach, but a number of difficulties were encountered, typically pertaining to the format of the answers. Directions of improvement from the current approach are proposed in Section 7.3.

EU Charter of Fundamental Rights (CFR) and mappings from biases

An important consideration in the context of this research is the requirements of the EU AI Act, and a novel contribution is proposed in mapping from harmful biases identified (in particular, societal bias categories) onto fundamental rights, indicating the potential for harm to a particular fundamental right, given the bias tests conducted. It was discussed how this approach aligns with the research question, particularly considering Article 10(2) part (f) of the AI Act.

In the Evaluation chapter, the usefulness of this was illustrated in Listing 28, where a number of potential fundamental rights violations are listed for a particular AI system. Despite the absence of thresholds to indicate a considerable risk of harm to a particular fundamental right, providing means through which all of those fundamental rights which may be harmed given the bias test results could potentially provide value in a fundamental rights impact assessment (FRIA) context.

7.2 Ethical considerations

Although this research project may be considered an endeavour in ethical AI (more specifically, AI risk management), it is nonetheless important that ethical considerations are discussed; indeed, Olteanu et al. (2023) discuss their belief that the research community on ethics in AI should do more to discuss potential impacts of their own research.

A number of ethical considerations arise from this project. With regard to the mappings between bias categories and potential fundamental rights violations, it is important to note that although a list of potential fundamental rights violations may be helpful in highlighting areas of concern for the fundamental rights impact assessment (FRIA) process, the mappings between failed bias tests and potential fundamental rights violations are heuristic and should not be relied upon solely. If the list of potential fundamental rights impacts, as in Listing 28, were seen as absolute (and other fundamental rights considerations hence discarded), this would pose an ethical issue.

Furthermore, particularly if the proposed approach were widely adopted, there would be great importance in the faithful representation of test results. If a test result is not faithfully indicated in its RDF representation, then this could have significant downstream consequences. It is for this reason that the `testResultURI` predicate was created. A negative consequence of facilitating the inheritance of risk information from upstream models is that unfaithful representations of risk profiling in upstream models may have downstream impacts, as the risk profile for systems which utilise this model is analysed with assumptions about the validity of the risk assessments conducted upstream. Extending the proposed approach with any further weaknesses which are identified can help to offset this risk, as those modelling risk can more precisely document the testing conducted.

7.3 Future work

In this section, a number of areas for potential future work are discussed.

7.3.1 Bias taxonomy

Although classes from the AI Ontology (AIO) were added to enhance the bias taxonomy based on the ISO/IEC 24027:2020 document and GDPR Article 9, it is highly probable that this taxonomy can be further expanded, and this represents an area for future work. Considering EU AI Act compliance practices (considering, in particular, Article 10(2)), this is perhaps most important for societal bias categories, as these biases may be directly mapped onto potential harms to an individual's fundamental rights.

7.3.2 Fairness metrics

Fairness metrics were implemented as instances of a single `FairnessMetric` class in this implementation. However, from Section 4.4.1, it is clear that fairness metrics may be arranged in a taxonomical manner, typically categorised as either individual or group fairness metrics. Although the main AI Fairness 360 (AIF360) Python library documentation does not typically indicate whether a particular fairness metric belongs to the individual or group category, the `scikit-learn` (Pedregosa et al. 2011) API classifies fairness metrics into "generic", "group" and "individual" categories[1].

7.3.3 Bias test documentation

Beyond the extension of the proposed bias taxonomy, there are areas of potential future work in the bias test documentation approach, with a notable example being the `groupDescription` predicate, which has a string (`xsd:string`) object. While a human-readable representation is suitable for many use-cases, declaring classes corresponding to different groups being analysed would allow for group-specific queries using SPARQL, which could be beneficial in a risk management context.

7.3.4 AI Cards alignment

A significant effort has been made to remain consistent with AIRO (Golpayegani et al. 2022) and AI Cards (Golpayegani et al. 2024) in the design of the proposed ontological extensions to facilitate the documentation of bias tests and as well as features on datasets relevant to risk management. The `BiasAudit` class is a subclass of `airo:Documentation` and is, considering the discussion in Section 4.5.1, compatible with AI Cards. However, while `dcat:Dataset` is represented in AIRO as a subclass of `airo:AIComponent` (as required to be considered an AI Cards key component), Datasheets (Gebru et al. 2021) fields are not represented as `airo:Documentation` and connected to the dataset using `airo:hasDocumentation`, as required.

Future work might investigate representing datasheet fields as `airo:Documentation`. One approach might involve the creation of a `Datasheet` class, a subclass of `airo:Documentation`, and representing datasheet fields on instances of this `Datasheet` class.

7.3.5 Datasheets for datasets RDF implementation

While the presented implementation of *Datasheets for datasets* (Gebru et al. 2021) structures, using DQV and DCAT, allow for a broad range of tasks, there are a number of areas for future work on this aspect of the project.

Areas for future work in this domain are summarised in the following:

[1] Available at the following URL: `https://aif360.readthedocs.io/en/stable/modules/sklearn.html#module-aif360.sklearn.metrics`. Verified 15.08.2024.

1. **Providing mechanisms for enhanced machine-readability:** While question fields in Datasheets have typically been divided into two or more subfields for enhanced machine-readability, there are a number of fields, particularly those providing descriptions, which are only represented using human-readable `xsd:string` types.

   Expanding upon this by providing options to further represent elements of (description) fields in a more machine-readable manner would represent a valuable area of future work. This would ideally utilise existing open ontologies and vocabularies, where possible.

2. **Providing a standard means of representing "N/A" fields:** One issue which was encountered in modelling datasheet use-cases was the presence of "N/A" fields. This can be problematic, because it results in the SHACL validation to fail if the expected field is not present in the expected format.

   Future work could investigate how such fields, i.e. where the datasheet author explicitly specifies that the field does not apply, could be represented without causing SHACL validation to fail.

3. **Providing means of referencing figures and tables:** A common observation in modelling the datasheets use-cases was the presence of figures or tables. In the current implementation, such references remain in the textual description, but there is no representation of these figures in the knowledge graph itself.

   Future work could identify how such references to figure and tables could be integrated into the current representation, enabling efficient retrieval of the desired figure or table referenced in a datasheet.

4. **An online interface for streamlined RDF generation for datasheets:** Although the SHACL shape to validate the well-formedness of an RDF datasheet representation is helpful in avoiding erroneous RDF datasheet representations, the process of manually representing the DQV measurements necessary to represent a datasheet in RDF can be quite time consuming.

   Moreover, it was clear when modelling the datasheets use-cases that fields are sometimes not filled in a manner which would allow for a satisfactory, consistent mapping onto more machine-readable data types.

   Future work might investigate the development of an application to streamline much of the RDF datasheet modelling process, requiring only that the user manually selects special types, such as those corresponding to specific entities, for machine-readability.

7.4 Conclusions

In this report, a number of novel contributions to the state of the art were outlined. It was discussed how this work differs from existing work and fills a gap in the state of the art. An important objective in facilitating a more taxonomically rich representation of bias in AIRO (Golpayegani et al. 2022), as outlined in the research motivation, was achieved.

The contributions of the proposed components, each constituents of a broader bias risk management framework for AI, were evaluated by various means. Importantly, with respect to answering the research question, it was noted how the contributions presented in this report contribute to requirements for high-risk AI systems (AI Act Annex IV), general-purpose AI systems (AI Act Annex XI) as well as to data governance and management practices for high-risk AI systems (AI Act Article 10).

Other means of evaluation included:

1. The representation of real-world use-cases in a machine-readable format, indicating the viability of the proposed approach;

2. The use of reasoners and validation programs to verify the syntactic and semantic validity of the ontology;

3. Evaluation of the ontology's capabilities using competency questions

Furthermore, the *Linked Open Terms* (LOT) methodology was utilised as a structured approach for ontology development. However, it is clear that there are limitations, as discussed in the Evaluation chapter as well as in Sections 7.1 & 7.3. Future work, providing improvements as necessary, may be integrated into the proposed ontology over time. This aligns with the **Ontology maintenance** activity of the LOT methodology (Poveda-Villalón et al. 2022).

Bibliography

Araujo, T., Helberger, N., Kruikemeier, S. & De Vreese, C. H. (2020), 'In ai we trust? perceptions about automated decision-making by artificial intelligence', *AI & society* **35**(3), 611–623.

Asuncion, A., Newman, D. et al. (2007), 'Uci machine learning repository'.

Bellamy, R. K. E., Dey, K., Hind, M., Hoffman, S. C., Houde, S., Kannan, K., Lohia, P., Martino, J., Mehta, S., Mojsilovic, A., Nagar, S., Ramamurthy, K. N., Richards, J., Saha, D., Sattigeri, P., Singh, M., Varshney, K. R. & Zhang, Y. (2018), 'Ai fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias'.
URL: *https://arxiv.org/abs/1810.01943*

Birhane, A., Steed, R., Ojewale, V., Vecchione, B. & Raji, I. D. (2024), Ai auditing: The broken bus on the road to ai accountability, *in* '2024 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)', IEEE, pp. 612–643.

Bogaerts, B., Jakubowski, M. & den Bussche, J. V. (2022), 'Shacl: A description logic in disguise'.
URL: *https://arxiv.org/abs/2108.06096*

Bommasani, R., Creel, K. A., Kumar, A., Jurafsky, D. & Liang, P. S. (2022), 'Picking on the same person: Does algorithmic monoculture lead to outcome homogenization?', *Advances in Neural Information Processing Systems* **35**, 3663–3678.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A. et al. (2020), 'Language models are few-shot learners', *Advances in neural information processing systems* **33**, 1877–1901.

Brundage, M. (2018), 'The malicious use of artificial intelligence: Forecasting, prevention, and mitigation'.

Castelnovo, A., Crupi, R., Greco, G., Regoli, D., Penco, I. G. & Cosentini, A. C. (2022), 'A clarification of the nuances in the fairness metrics

landscape', *Scientific Reports* **12**(1).
URL: *http://dx.doi.org/10.1038/s41598-022-07939-1*

Crawford, K. (2017), The trouble with bias. NIPS 2017 Keynote.
URL: *https://blog.revolutionanalytics.com/2017/12/the-trouble-with-bias-by-kate-crawford.html*

d'Alessandro, B., O'Neil, C. & LaGatta, T. (2017), 'Conscientious classification: A data scientist's guide to discrimination-aware classification', *Big data* **5**(2), 120–134.

Dengel, A., Etzioni, O., DeCario, N., Hoos, H., Li, F., Tsujii, J. & Traverso, P. (2021), 'Next big challenges in core ai technology', pp. 90–115.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O. & Zemel, R. (2012), Fairness through awareness, *in* 'Proceedings of the 3rd innovations in theoretical computer science conference', pp. 214–226.

Franklin, J. S., Bhanot, K., Ghalwash, M., Bennett, K. P., McCusker, J. & McGuinness, D. L. (2022), An ontology for fairness metrics, *in* 'Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society', pp. 265–275.

Garbin, C., Rajpurkar, P., Irvin, J., Lungren, M. P. & Marques, O. (2021), 'Structured dataset documentation: a datasheet for chexpert'.
URL: *https://arxiv.org/abs/2105.03020*

Garijo, D. (2017), Widoco: a wizard for documenting ontologies, *in* 'International Semantic Web Conference', Springer, Cham, pp. 94–102.
URL: *http://dgarijo.com/papers/widoco-iswc2017.pdf*

Gebru, T., Morgenstern, J., Vecchione, B., Vaughan, J. W., Wallach, H., au2, H. D. I. & Crawford, K. (2021), 'Datasheets for datasets'.
URL: *https://arxiv.org/abs/1803.09010*

Glimm, B., Horrocks, I., Motik, B., Stoilos, G. & Wang, Z. (2014), 'Hermit: an owl 2 reasoner', *Journal of automated reasoning* **53**, 245–269.

Golpayegani, D., Hupont, I., Panigutti, C., Pandit, H. J., Schade, S., O'Sullivan, D. & Lewis, D. (2024), 'Ai cards: Towards an applied framework for machine-readable ai and risk documentation inspired by the eu ai act', *arXiv preprint arXiv:2406.18211* .

Golpayegani, D., Pandit, H. & Lewis, D. (2022), *AIRO: An Ontology for Representing AI Risks Based on the Proposed EU AI Act and ISO Risk Management Standards.*

Gruber, T. R. (1995), 'Toward principles for the design of ontologies used for knowledge sharing?', *International journal of human-computer studies* **43**(5-6), 907–928.

Hardt, M., Price, E. & Srebro, N. (2016), 'Equality of opportunity in supervised learning', *Advances in neural information processing systems* **29**.

Hendler, J., Lassila, O. & Berners-Lee, T. (2001), 'The semantic web', *Scientific American* **284**(5), 34–43.

Holland, S., Hosny, A., Newman, S., Joseph, J. & Chmielinski, K. (2018), 'The dataset nutrition label: A framework to drive higher data quality standards'.
URL: *https://arxiv.org/abs/1805.03677*

Hu, Y., Li, W., Wright, D., Aydin, O., Wilson, D., Maher, O. & Raad, M. (2019), 'Artificial intelligence approaches', *Geographic Information Science & Technology Body of Knowledge* **2019**(Q3).
URL: *http://dx.doi.org/10.22224/gistbok/2019.3.4*

Joachimiak, M. P., Miller, M. A., Caufield, J. H., Ly, R., Harris, N. L., Tritt, A., Mungall, C. J. & Bouchard, K. E. (2024), 'The artificial intelligence ontology: Llm-assisted construction of ai concept hierarchies'.
URL: *https://arxiv.org/abs/2404.03044*

Jordan, M. I. & Mitchell, T. M. (2015), 'Machine learning: Trends, perspectives, and prospects', *Science* **349**(6245), 255–260.
URL: *https://www.science.org/doi/abs/10.1126/science.aaa8415*

Kamiran, F. & Žliobaitė, I. (2013), Explainable and non-explainable discrimination in classification, *in* 'Discrimination and Privacy in the Information Society: Data mining and profiling in large databases', Springer, pp. 155–170.

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J. & Amodei, D. (2020), 'Scaling laws for neural language models', *arXiv preprint arXiv:2001.08361* .

Kearns, M., Neel, S., Roth, A. & Wu, Z. S. (2018), Preventing fairness gerrymandering: Auditing and learning for subgroup fairness, *in* 'International conference on machine learning', PMLR, pp. 2564–2572.

Kearns, M., Neel, S., Roth, A. & Wu, Z. S. (2019), An empirical study of rich subgroup fairness for machine learning, *in* 'Proceedings of the conference on fairness, accountability, and transparency', pp. 100–109.

Khlaaf, H. (2023), 'Toward comprehensive risk assessments and assurance of ai-based systems', *Trail of Bits* .

Kusner, M. J., Loftus, J., Russell, C. & Silva, R. (2017), 'Counterfactual fairness', *Advances in neural information processing systems* **30**.

Lalonde, C. & Boiral, O. (2012), 'Managing risks through iso 31000: A critical analysis', *Risk management* **14**, 272–300.

Leitch, M. et al. (2010), 'Iso 31000: 2009-the new international standard on risk management', *Risk analysis* **30**(6), 887.

McGuinness, D. L., Van Harmelen, F. et al. (2004), 'Owl web ontology language overview', *W3C recommendation* **10**(10), 2004.

Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K. & Galstyan, A. (2021), 'A survey on bias and fairness in machine learning', *ACM computing surveys (CSUR)* **54**(6), 1–35.

Miron, M., Tolan, S., Gómez, E. & Castillo, C. (2021), 'Evaluating causes of algorithmic bias in juvenile criminal recidivism', *Artificial Intelligence and Law* **29**(2), 111–147.

Musen, M. A. (2015), 'The protégé project: a look back and a look forward', *AI matters* **1**(4), 4–12.

Olteanu, A., Ekstrand, M., Castillo, C. & Suh, J. (2023), 'Responsible ai research needs impact statements too'.
URL: *https://arxiv.org/abs/2311.11776*

pandas development team, T. (2020), 'pandas-dev/pandas: Pandas'.
URL: *https://doi.org/10.5281/zenodo.3509134*

Pandit, H. J., O'Sullivan, D. & Lewis, D. (2019), Test-driven approach towards gdpr compliance, *in* 'Semantic Systems. The Power of AI and Knowledge Graphs: 15th International Conference, SEMANTiCS 2019, Karlsruhe, Germany, September 9–12, 2019, Proceedings 15', Springer, pp. 19–33.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. et al. (2011), 'Scikit-learn: Machine learning in python', *Journal of machine learning research* **12**(Oct), 2825–2830.

Poveda-Villalón, M., Fernández-Izquierdo, A., Fernández-López, M. & García-Castro, R. (2022), 'Lot: An industrial oriented ontology engineering framework', *Engineering Applications of Artificial Intelligence* **111**, 104755.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I. et al. (2019), 'Language models are unsupervised multitask learners', *OpenAI blog* **1**(8), 9.

Robaldo, L., Batsakis, S., Calegari, R., Calimeri, F., Fujita, M., Governatori, G., Morelli, M. C., Pacenza, F., Pisano, G., Satoh, K. et al. (2023), 'Compliance checking on first-order knowledge with conflicting and compensatory norms: a comparison among currently available technologies', *Artificial Intelligence and Law* pp. 1–51.

Rodriguez-Ruiz, A., Lång, K., Gubern-Merida, A., Broeders, M., Gennaro, G., Clauser, P., Helbich, T. H., Chevalier, M., Tan, T., Mertelmeier, T., Wallis, M. G., Andersson, I., Zackrisson, S., Mann, R. M. & Sechopoulos, I. (2019), 'Stand-Alone Artificial Intelligence for Breast Cancer Detection in Mammography: Comparison With 101 Radiologists', *JNCI: Journal of the National Cancer Institute* **111**(9), 916–922.
URL: *https://doi.org/10.1093/jnci/djy222*

Russo, M. & Vidal, M.-E. (2024), 'Leveraging ontologies to document bias in data'.
URL: *https://arxiv.org/abs/2407.00509*

Schneider, J., Meske, C. & Kuss, P. (2024), 'Foundation models: A new paradigm for artificial intelligence', *Business & Information Systems Engineering* pp. 1–11.

Schwartz, R., Schwartz, R., Vassilev, A., Greene, K., Perine, L., Burt, A. & Hall, P. (2022), *Towards a standard for identifying and managing bias in artificial intelligence*, Vol. 3, US Department of Commerce, National Institute of Standards and Technology.

Shah, D. S., Schwartz, H. A. & Hovy, D. (2020), Predictive biases in natural language processing models: A conceptual framework and overview, *in* 'Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics', Association for Computational Linguistics.
URL: *http://dx.doi.org/10.18653/v1/2020.acl-main.468*

Tversky, A. & Kahneman, D. (1974), 'Judgment under uncertainty: Heuristics and biases: Biases in judgments reveal some heuristics of thinking under uncertainty.', *science* **185**(4157), 1124–1131.

Weidinger, L., Mellor, J., Rauh, M., Griffin, C., Uesato, J., Huang, P.-S., Cheng, M., Glaese, M., Balle, B., Kasirzadeh, A., Kenton, Z., Brown, S., Hawkins, W., Stepleton, T., Biles, C., Birhane, A., Haas, J., Rimell, L., Hendricks, L. A., Isaac, W., Legassick, S., Irving, G. & Gabriel, I. (2021), 'Ethical and social risks of harm from language models'.

Wes McKinney (2010), Data Structures for Statistical Computing in Python, *in* Stéfan van der Walt & Jarrod Millman, eds, 'Proceedings of the 9th Python in Science Conference', pp. 56 – 61.

Zhang, L., Wu, Y. & Wu, X. (2016), 'A causal framework for discovering and removing direct and indirect discrimination', *arXiv preprint arXiv:1611.07509* .