Explainable Artificial Intelligence and User Acceptance

Alex Cummaudo
Applied Artificial Intelligence Institute
Deakin University, Australia

March 22, 2018

1 Introduction

What makes outcomes inferred from artificial intelligence (AI) trusting? How do we convincingly communicate results inferred from an AI model to the end-user? Moreover, what types of stakeholders exist, and how would these explanations differ between each? As our AI models improve, we must take a step back and *explain* the outcomes we infer. We must, therefore, begin to embark in a new chapter of AI that makes our models accountable for. Explainable AI (XAI) is this new chapter where we question the outcomes of AI models to reason why the outcome is true, and thus AI for businesses and developers (or any non-AI expert) to accept of such outcomes becomes an integral challenge.

To make an outcome accepting, the explanation must be convincing. How does one make a convincing explanation, and what is the best language to use to explain to the right end user? Studies show explanations are imperative to Decision Support Systems (DSS), and so if an AI is to be used to support decision-making, it must have a thorough but clear explanation to solidify any claim. Therefore, the communication of these explanations are more important than the outcomes themselves.

To make the communication clear, we focus on what makes AI 'intelligent' in the first place. This is approximately defined as a perception of *human*-like intelligence: an AI is only as good as a human trains it to be. Thus, if it is a perception of human intelligence, then what are the attributes of human intelligence? Humans learn from theories that govern specific domains, theories that are gathered and expanded upon over centuries. Yet, these theories are generally based on inductive reasoning on our world: we produce theories through observation and, likewise, so does AI based on what it observes. What XAI aims to solve is the *deductive* reasoning process: given an output, explain the theory.

2 Case Studies

2.1 Executive Management

2.2 Software Engineers