Foreword

If the experience of the last few decades of Software Engineering is anything to go by, we seem to always be catching up with the innovations taking place in a hyperactive market, increasing the chances of developing systems that don't fully take account of the needs of users, don't meet legal obligations and end up compromising reliability, security and maintainability.

Fortunately, the software industry is good at learning as we go. Each innovation is typically followed by excitement in the market, some unfortunate system failures due to oversight in the requirements and, most importantly, a realisation that our Software Engineering methods need to adapt to meet the new demands. Examples of this learning process include an era where formal methods of software engineering were developed in the 90's to meet the need to ensure reliability in safety critical applications, and more recently, the drive to adopt agile development methods to increase productivity and reduce risks.

The most recent efforts to improve the software engineering process, which is the subject of this edited collection, is to utilise crowdsourcing and methods from machine learning. Although crowdsourcing, which was advocated by Jeff Howe as a way of achieving "wisdom of the crowd" as far back as 2006, its proposed use for developing software has been more recent and is growing rapidly. The essence of crowdsourcing for software development is to utilise a community of external stakeholders, including potential users, analysts and programmers, to participate in the development of an application on the premise that all stakeholders will eventually gain mutual benefits.

This broad view of crowdsourcing and use of machine learning for software engineering raises many questions, such as:

- How does one use crowdsourcing effectively in the different phases of software development, from requirements elicitation to testing and then maintenance and deployment?
- We know from recent history, that new software engineering methodologies are not universally applicable, so are there specific types of applications where use of crowdsourcing and use of machine learning is best?

This edited collection brings together several studies addressing such questions. The chapters include systematic reviews of the field, case studies showing the use of machine learning and crowdsourcing in domains such as construction and aerospace, and key perspectives from the IT industry.

The chapters in this book will provide valuable insight for both academics pursuing research in this field and software development companies, who are seeking to improve their processes by using crowdsourcing or AI methods for software engineering.

Sunil Vadera
University of Salford, UK
July 2019