Cost Estimation Guidance for AI Software Development Projects

Arlene F. Minkiewicz

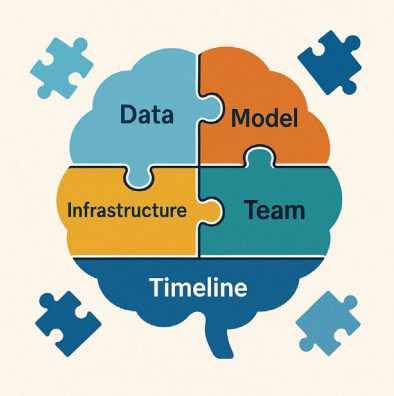
ICEAA Workshop, May 2025

Agenda

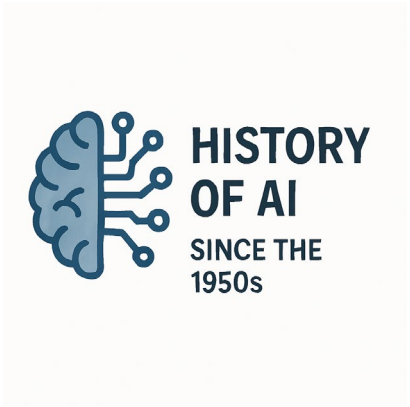
- Introduction
- Artificial Intelligence – history in software development
- Traditional Cost Estimation – challenges for AI projects
- Key Factors Influencing Cost and Effort in AI Projects
- Real World Examples of Cost Related Challenges
- Proposed Framework for Cost Estimation in AI Projects
- Future work
- Wrap Up

Introduction

- Powered by increasing technology and processing power, the use of Artificial Intelligence (AI) in Software Development projects is increasing
- Traditional Software Cost Estimation methods fall short with AI Software development projects in several areas
- This paper proposes a multi-phased framework intended to address these short falls
- Key factors impacting cost and effort will be discussed
- Emphasis is placed on the iterative and experimental nature of AI development



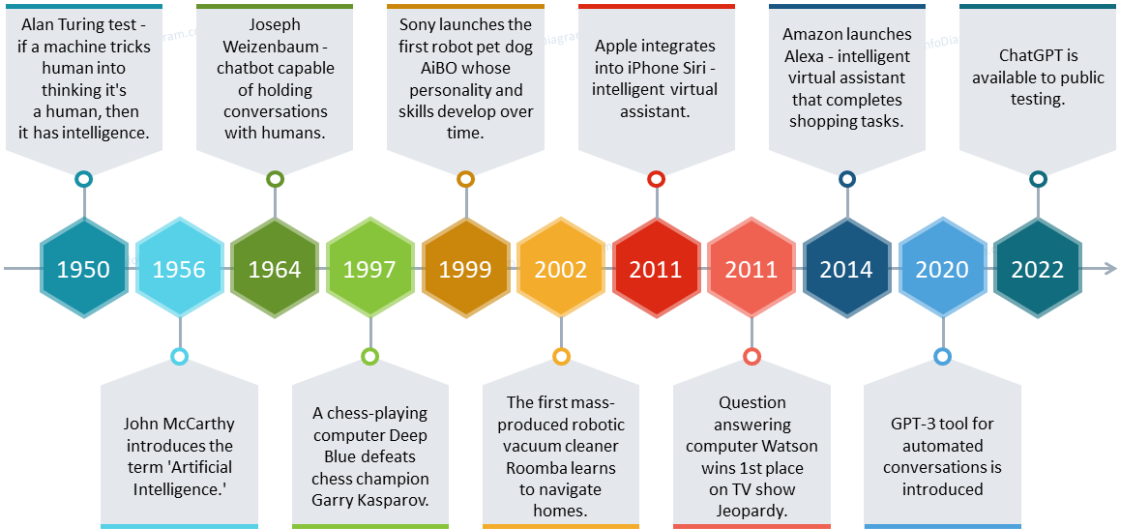
Artificial Intelligence in SW Development Projects



- In the 1950's – AI centered around rule-based systems and symbolic reasoning
- Early efforts were stymied by limited computational power and lack of real-world data
- In 80's and 90's - expert systems emerged as new applications of AI
  - Demonstrate potential of AI
  - Simulation of human decision-making skills through codification of specialized knowledge
  - Limited in adaptability and failed to provide generalized intelligence
- In the 90's and 2000's – computing power and large amounts of data – ML techniques flourished
- ML and ML empowered deep learning seem to be significant game changers for AI

Artificial Intelligence - the Journey

Artificial Intelligence Development History Timeline



Get these slides & icons at www.infoDiagram.com

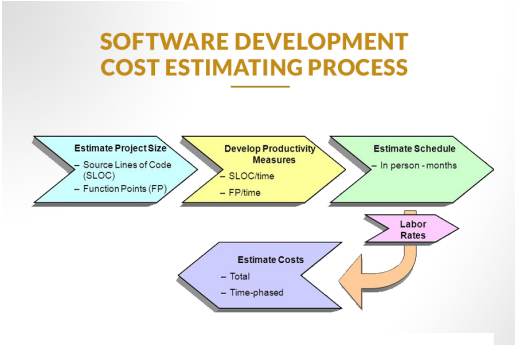
The Estimation Challenge

Traditional Software Cost Estimation

- Cost Estimation is an important aspect of program management for SW development projects
  - Good estimation is critical for project success
- Software has become ubiquitous in almost every important system
- Consider the F-35 Joint Strike Fighter military aircraft compared to the F-16
  - 177x more computer code
  - 300x software development effort/costs
  - 90% of functionality is delivered by software
- Software cost estimation relies on traditional size and complexity measures

Where traditional software estimation methods fail…

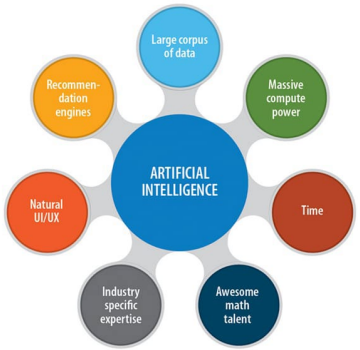
- Cutting edge technology
- Unpredictable nature of data quality and availability
- Unpredictable nature of modeling training and tuning– iterative process
- Tasks such as data acquisition, cleaning, labeling and preprocessing – depend on the nature of the data not the type of project
- High variability from project to project – model development tuning.
- Specialized skill requirements – in addition to SME knowledge, skills such as data science, Machine Learning, Big Data, NLP, etc.



Where traditional software estimation methods fail…

- Ongoing maintenance including monitoring, retraining and model drift
- Lack of relevant historical data
- Iterative development experimentation in AI projects
- Addressing ethical and regulatory requirements
- Complexity of testing and validation
- Computational resource demands
- Traditional size and complexity measures are only part of the equation

Key factors Influencing Cost and Effort in AI Projects



- Data Preparation – acquisition, cleaning and preparation
- Algorithm Selection – choosing the right algorithms
- Model Training – Resource requirements and time considerations
- Deployment and Maintenance – challenges and ongoing costs in AI systems
- Team expertise – role of skills and experience (high priced people)
- Tool Selection and Maturity – effect of tools on development efficiency and cost
- Application Platforms – considerations for platform specific costs
- Integration Challenges – issues related to legacy systems and compatibility

Real World AI Cost Challenges

Real World Example - OpenAI's GPT-5 (Orion) Project

- Project Goal
  - Create a more versatile and efficient AI system by unifying OpenAI technologies to surpass GPT-4 through improvements to contextual understanding and creative generation
- Challenges
  - Availability of high-quality training data. OpenAI has had to employ synthetic data generation and human-created datasets
  - Extremely expensive training runs, with iterations approaching $500 million apiece
  - Disappointing performance improvements
- Lessons Learned
  - Limitations of scaling – improved performance requires more than increasing model size and data
  - Data quality and availability is essential to successful AI advancements
  - Groundbreaking improvements require innovative approaches

Real World Example - IBM Watson Health

- Project Goal
  - Revolutionize healthcare by using AI for analyzing medical data, assisting doctors in making more accurate diagnoses and providing tailored treatment recommendations
- Challenges
  - Project faced significant cost issues due to the complexity integrating AI into clinical environments
  - Need for constant data updates
  - Healthcare data is messy – often inconsistent or incomplete – training is costly and time consuming
  - High operational and maintenance costs
  - Data interoperability and integration technical limitations
- Lessons Learned
  - AI projects in highly regulated fields like healthcare face substantial costs for data handling, compliance and customization to clinical workflows

IBM Watson

Real World Example - Tesla's Autopilot and Full Self-Driving (FSD)

- Project Goal
  - Tesla's FSD software development project aimed to deliver a fully autonomous driving experience powered by advanced ML algorithms
- Challenges
  - Developing and refining autonomous driving technology – substantial costs
  - Heavy investment in massive computational resources to run and train AI models
  - Expensive sensor technology
  - Global test fleets to ensure safety
  - Software needed continuous updates and vast quantities of real-world driving data
  - Regulatory delays and liability issues
- Lessons learned
  - Autonomous AI requires substantial up front investment in hardware and data
  - High costs for maintenance, model updates and regulatory compliance

Real World Example - Apples Project Titan

- Project Goal
  - Create a fully autonomous electric vehicle that would revolutionize the automotive industry offering deep integration with Apple's existing products and services.
- Challenges
  - Creating a vehicle with no steering wheels or pedals required unprecedented technological advancements
  - Lack of coherent and consistent strategy led to repeated redesigns and shifts in focus resulting in resource misallocation and escalating costs
  - Lack of manufacturing partner and automotive expertise resulted in difficulties with battery technology, self-driving systems and overall vehicle design.
- Lessons Learned
  - Clear vision, well-defined strategic vision and consistent leadership are crucial for success of large-scale projects
  - Entering a complex industry requires deep industry knowledge and experience, not just innovation
  - Recognizing when to pivot is vital, though the project was cancelled in 2024, Apple was able to redirect its focus toward generative AI turning to areas with greater synergy to its core competency

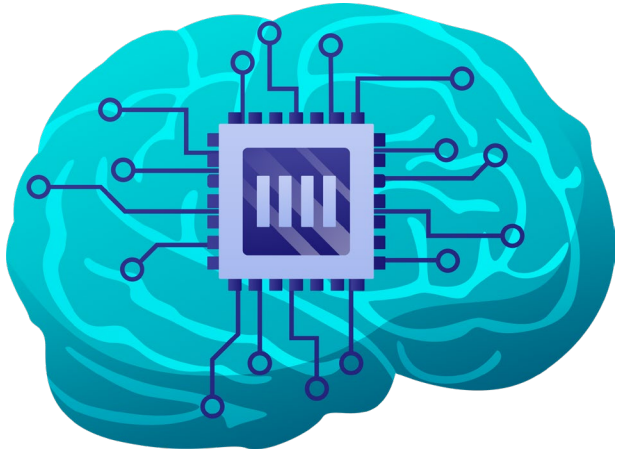
Real World Example - Lockheed Martin's F-35 Lightening II Program

- Project Goal
  - Develop a single aircraft platform adaptable for various missions across multiple branches of the military incorporating cutting-edge stealth , sensor fusion and network centric warfare technologies depending heavily on AI technologies for situational awareness, autonomy, mission adaptability and pilot decision making. Pivotal to project success is the development of the Autonomic Logistics Information System (ALIS)
- Challenges
  - Integration challenges combining diverse data types (radar, EW, IR, comms, etc.) from multiple sources
  - Reliability of AI models – validating AI decisions against traditional military rules of engagement
  - Hardware constraints – onboard AI within limited computational and power budgets
  - Calibrating trust between pilot and AI system was nontrivial
- Lessons Learned
  - Human – Centric AI Design is crucial
  - Iterative development and testing is best strategy
  - AI is a force multiplier – AI does not replace human judgement
  - Model validation and robustness is a long-term investment
  -

Framework for Cost Estimation in AI Projects

Framework for Cost Estimation in AI Projects

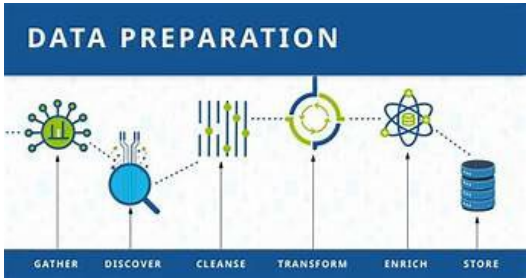
- 1. Project Scope Definition
- 2. Phased Development Approach
  - Phase 1 – Planning
  - Phase 2 – Data Preparation
  - Phase 3 – Algorithm Selection and Development
  - Phase 4 – Model Training and Tuning
  - Phase 5 – Deployment
  - Phase 6 – Maintenance and Monitoring
- 3. Cost Factors and Estimating Techniques
- 4. Dynamic and Iterative Estimation Process
- 5. Team Expertise and Resource Allocation
- 6. Risk Management
- 7. Case Study Review and Validation



Framework for Cost Estimation in AI Projects – Project Scope Definition

- Clear identification of project objectives - what are the project goals and desired outcomes
- Outline the specific features and functionalities required
- Identify clearly (or as clearly as possible) the data requirements
  - Types of data required
  - Volume
  - Quality Requirements
  - Potential Sources (and viability of acquisition of these sources)
  - Unique data considerations

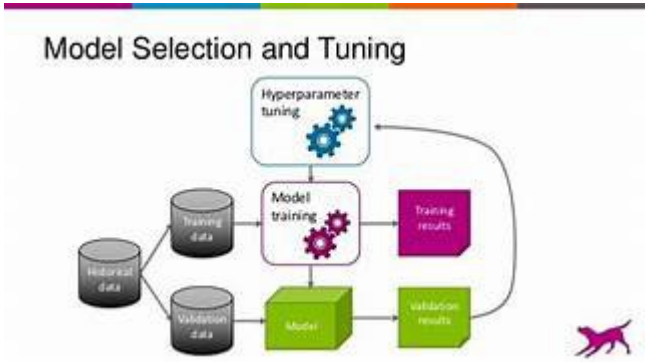
Framework for Cost Estimation in AI Projects – Phased Development Approach



- Phase 1 – Planning
  - Conduct feasibility studies
  - Perform preliminary risk assessments
  - Identify stakeholders
  - Create communication plans and identify owner(s)
  - Develop the high-level project timeline (aware that this will likely change – work that into plan)
- Phase 2 – Data Preparation
  - Estimate cost for data collection, cleaning and preparation – consider types of data, unique data characteristics, potential data sources, obstacles to data collection
  - Consider data acquisition methods – internal or external, data collection method(s), purchased data
  - Consider the associated costs with acquiring data (licensing, storage, etc.)

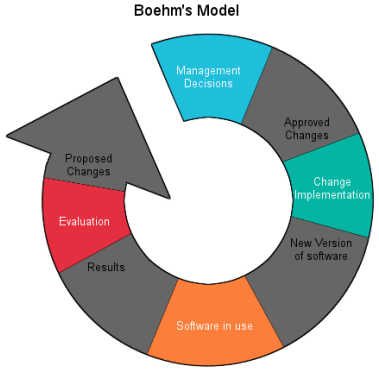
Framework for Cost Estimation in AI Projects – Phased Development Approach

- Phase 3 – Algorithm Selection and Development
  - Assess skills required and evaluate resource pool,
  - Evaluate potential algorithms based on the requirements of the project
  - Estimate time and resources for model development (consult SMEs and others involved in similar projects)
  - Consider the impacts of prototyping and testing
- Phase 4 – Model Training and Tuning
  - Assess skills required and evaluate resource pool
  - Assess model complexity
  - Assess computational resources needed for training - consider things like GPU usage, cloud services, hardware needs, etc.
  - Estimate time for parameter tuning and validation (consult SME's and others involved in similar projects)



Framework for Cost Estimation in AI Projects – Phased Development Approach

- Phase 5 – Deployment
  - Consider the costs related to integrating the AI model into existing systems
  - Consider the on-going infrastructure costs (cloud services, on-premise servers, etc.)
    - Cloud based services such as ASW SageMaker or Google Vertex XI offer scalable solutions but can be pricey
    - Edge computing solutions may be more cost effective but come with initial hardware investments
- Phase 6 – Maintenance and Monitoring
  - Consider ongoing costs for model maintenance, updates, retuning, performance monitoring
    - Data drift, adversarial attacks, evolving regulatory requirements, etc.
  - Identify metrics for success and tracking continuous improvements
  - Consider the on-going infrastructure costs (cloud services, on-premise servers, etc.)



Framework for Cost Estimation in AI Projects – Cost Factors and Estimation Techniques

- Traditional estimation methods are good but need to be tuned for the unique nature of AI development
  - Expert Judgement – experts are still valuable, but their opinions should be metered by experience with AI projects
  - Analogous Estimation – a poor choice if there are no similar projects
  - Parametric Estimation
    - Beware that these models are based on historical data and project parameters
    - Size and Complexity are still important, but estimator should also consider things outside of their current model
      - Data issues (acquisition, cleansing, preprocessing, availability, quality)
      - Iterative nature of data and model tuning should be incorporated
      - Considerations comparable with estimating agile projects should rule
    - Bottoms Up Estimation - break down of costs is still valuable, but awareness of the iterative nature should contribute to estimate uncertainty – especially when organizational history is limited

Framework for Cost Estimation in AI Projects – Dynamic and Iterative Estimation Process

- It is not going to be easy for organizations starting out with AI
- Successful estimators will embrace techniques favored for estimating agile projects
  - Expect things to change
  - Embrace this change
  - Communicate the risks to the stakeholders
  - Flexibility is key to successful projects
  - Probably won't get it right the first time
  - Establish success metrics and track to learn from each project
- Regularly update estimates based on ongoing findings

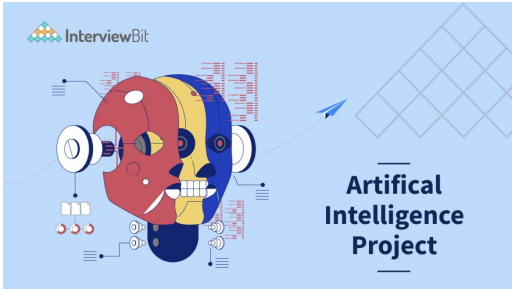
Framework for Cost Estimation in AI Projects – Team Expertise and Resource Allocation

- The resource skill set for AI projects are currently in high demand – incorporate salary differentials
- Assess the skills and experience of team members, factoring in the fact that training and or new hires may be necessary to meet project requirements
- Assess the maturity and capability of tools and platforms beings used
- Consider the maturity and processing power of existing hardware
- Be prepared to invest in resources necessary for project success – human, technology, equipment, etc.

Framework for Cost Estimation in AI Projects – Risk Management and Case Study Review and Validation

- Carefully consider potential risks specific to AI projects (data availability and quality, algorithm bias, iterative nature of AI development)
- Stay away from a Point Estimate - give a range that recognizes the uncertainty in estimation decisions
- Develop mitigation strategies and allocate contingency resources
- Where possible – review similar relevant projects – understand how your estimate aligns with historical information (trust your history)
- Incorporate lessons learned from previous AI projects to validate current estimate and establish credibility

Next Steps



- The framework proposed
  - 50% reliant on Lessons learned from study of AI projects
  - 50% common sense – something all estimators want to think when they estimate
- While AI software development projects aren't new – the speed with which the technology is advancing – makes history not necessarily relevant
- The proposed framework gives a structure for estimators to consider how to estimate software projects and a structure for collecting data going forward
- Next steps are to identify AI projects where quantitative data collection and lessons learned can be collected and used to improve the guidance
- Looking for AI software projects willing to share data (in a perfect world) or lessons learned (in the real world)

Wrap Up

- Technology and data availability creates a situation where Artificial Intelligences applications are practical and valuable
- Organizations intent on developing such apps need to revisit the way they estimate these projects
- Factors such as the cutting nature of the technology, requirements for large amounts of data acquisition and handling, iterative process of data collection, model tuning and retraining are all things that may not be part of the history estimates are based on
- A framework has been presented to help estimators consider these unique factor as appropriate to better aid decision makers in their organizations

arlene.minkiewicz@unisonglobal.com