

# Análise do impacto dos atributos de projetos de software na estimativa de esforço utilizando analogia

Silva, C. I. L. C. \*; Brandão, A. A. F.\* Siqueira, F. L.†;

\*Laboratório de Técnicas Inteligentes - Escola Politécnica - USP

†Laboratório de Tecnologia de Software - Escola Politécnica - USP

E-mail: {igortkd2006, anarosabrandao, levy.siqueira}@gmail.com

**Resumo**— A estimativa de esforço no desenvolvimento de projetos de software é uma métrica primordial para o sucesso do projeto. Os atributos de projetos de software são as características que os diferenciam, por exemplo, a arquitetura utilizada, a metodologia, a atividade operacional do cliente, entre outros. Apesar dos esforços na busca de um método de estimativa que seja adaptável e preciso, ainda não existe uma técnica unânime na literatura. ABE é uma técnica de estimativa híbrida e adaptável, que apresenta bons resultados e baixo nível de complexidade. Apesar disso existem poucos estudos sobre o ABE com foco na utilização de atributos para caracterização dos dados e o seu impacto na estimativa final. Compreender os detalhes da etapa de caracterização do ABE, é necessário para aproveitar melhor o potencial de cada atributo no modelo de estimativa por analogia. Assim a proposta desse trabalho é analisar o impacto dos principais atributos de projetos de software na métrica de estimativa de esforço por analogia.

**Palavras-chave**— analogy based effort; estimativa de esforço, inteligência artificial, projetos de software.

**Classificação**— Mestrado

**Categoria**— Iniciante

## I. INTRODUÇÃO

Segundo Boehm, Abts e Chulani (2000) a estimativa do esforço de um projeto software é uma métrica utilizada em diversas áreas da engenharia de software, tais como, planejamento, gerenciamento, análise de risco, orçamento, entre outros. Trendowicz e Jeffery (2014) afirmam que estimativas pobres são uma das principais causas do aumento de custo da produção de um software e a segunda maior causa de desistência de projetos. Nos últimos anos a acurácia da estimativa de software têm sido um desafio devido às peculiares características desse produto como a abstração, intangibilidade [2].

Encontrar um método de estimativa eficaz, preciso e adaptável tem sido um desafio a diversos profissionais. Além das dificuldades provenientes das características do produto, no momento da estimativa, diversas características do produto e do projeto, se encontram indefinidas ou desconhecidas. É comum no momento da estimativa estaem indefinidas a metodologia, o time e sua maturidade, a arquitetura, entre outros [3].

O dicionário Aurélio *on-line* define analogia como: "Relação de semelhança entre objetos diferentes". Segundo Keung (2009), analogia é uma técnica utilizada até hoje no cotidiano

das pessoas e pode ser definida como: "A relação entre dois problemas, medida em similaridade"[4]. É uma técnica multidisciplinar que utiliza o aprendizado de casos passados, ou análogos, para resolver um problema novo, também chamado de problema alvo [4].

A resolução de problemas por analogia é uma técnica comumente utilizada no cotidiano das pessoas, e por isso atrai maior simpatia dos profissionais da área [5]. É uma técnica de inteligência artificial, que utiliza o conhecimento adquirido previamente, para resolução de novos problemas. A utilização do conhecimento adquirido minimiza a dependência de informações do problema alvo [6].

Segundo Trendowicz e Jeffery (2014) as principais vantagens do ABE, *Analogy Based Effort*, são:

- Não necessita de um especialista em estimativas;
- Flexível;
- Documentação extensa com diversos trabalhos acadêmicos e ferramentas publicadas;
- Reutilizável;
- Validado empiricamente em estudos de contexto acadêmico e comercial;
- Possui bons resultados.

Uma das etapas do ABE é o cálculo da similaridade, que é a distância entre os atributos do projeto alvo e os atributos dos projetos análogos. Baseado nos projetos mais similares é feita a estimativa do projeto alvo [2]. Assim o cálculo da similaridade é uma etapa crucial da técnica, e parte da premissa que quanto maior a similaridade entre dois projetos, maior a similaridade entre suas estimativas.

Apesar de existirem estudos sobre os atributos e seu impacto na estimativa de esforço [3], [7], [8], [9], [10], não foram encontrados trabalhos que aplicassem esse conhecimento no cálculo de similaridade do ABE. Assim o objetivo desse trabalho é analisar a utilização dos atributos de projetos de software, no cálculo da similaridade do ABE, levando em consideração o impacto na estimativa de esforço do modelo.

## II. TRABALHOS CORRELATOS

ABE utiliza as informações do projeto alvo para encontrar os projetos mais similares. A estimativa então é obtida baseada nos projetos similares, cujo esforço de produção já foi

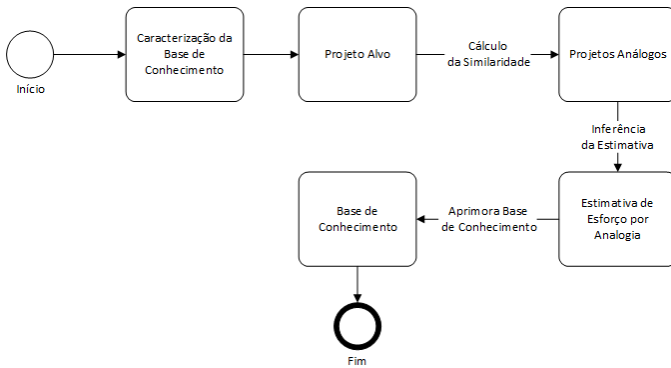


Figura 1. Etapas do ABE

computado [4]. O processo análogo parte da premissa que o cálculo da similaridade sofre menor influência da ausência de informações do que o cálculo da estimativa em si.

Segundo Shepperd e Schofield (1997) ABE é uma técnica que consiste em quatro etapas: Caracterização da base de dados para identificação dos atributos a serem utilizados; Cálculo da similaridade; Solução análoga, utiliza o conhecimento dos casos similares para prover possíveis soluções; Sugestão de uma solução ao caso alvo. Na Figura 1 é possível verificar o fluxograma básico da estimativa por analogia.

Ao utilizar ABE o primeiro passo é a caracterização da base de projetos, que consiste na escolha dos atributos que serão utilizados nas demais etapas do processo. São exemplos de atributos: tamanho do software; arquitetura utilizada; metodologia de desenvolvimento; maturidade do time; entre outros [2], [5]. Os atributos podem ser classificados como atributos de contexto ou atributos de influência.

O ambiente de um projeto de software possui diversas características que se mantêm constantes em determinados projetos. Essas características de um software são chamadas de atributos de contexto. Esses atributos geralmente são qualitativos, portanto dificilmente podem ser expressos em números. Por exemplo, domínio da aplicação, tipo de desenvolvimento, linguagem de programação, metodologia, etc. [3].

Diferentes projetos com fatores de contexto semelhantes podem apresentar métricas de estimativa de esforço bastante diferentes entre si. O desvio de esforço entre esses projetos se deve às mudanças nos demais atributos, aqueles que não são atributos de contexto, e são chamados de atributos de influência. Esses atributos são inúmeros e dificilmente um modelo será capaz de analisar todos, portanto o importante é analisar os fatores de influência mais importantes do projeto a ser estimado [3]. Os atributos de influência geralmente são expressos quantitativamente. São exemplos: métricas de tamanho de software, quantidade de programadores, tempo de execução, métricas de qualidade, análise de risco, entre outros.

Diversos estudos tem sido aplicados para encontrar uma relação entre os atributos e métricas de software como produtividade, esforço, risco, qualidade e outros, vide [3], [7], [8], [9], [10]. De fato a escolha dos atributos é relevante em modelos de estimativa, porém no momento da estimativa, pode

Atributos de Influência	Frequência
<b>Qualificação e experiência do time</b>	<b>64</b>
Experiência com a linguagem de programação	16
Experiência e familiaridade com a aplicação	16
Experiência e habilidade do gestor do projeto	15
<b>Complexidade do software</b>	<b>42</b>
Tamanho e complexidade da base de dados	9
Complexidade da arquitetura	9
Complexidade da interface com outros sistemas	8
<b>Projeto</b>	<b>41</b>
Pressão do calendário	43
Desenvolvimento descentralizado/Multisite	9
<b>Uso de Ferramentas de qualidade/efetividade</b>	<b>41</b>
Ferramentas de CASE	12
Ferramentas de teste	5
<b>Atributos de Contexto</b>	
Linguagem de programação	29
Domínio	14
Tipo de desenvolvimento	11

Figura 2. Fatores mais impactantes na produtividade de software fonte: Trendowicz e Münch (2009)

ser difícil mensurar certos atributos. A utilização do modelo baseado em analogia minimiza essa dificuldade pois utilizará as medidas dos casos análogos.

Segundo [11] a estimativa de esforço pode ser impactada por diversos fatores e suas interações, o autor cita como exemplo o aumento da produtividade quando há decréscimo na qualidade. Apesar da relação citada ser facilmente inferida, outras podem não ocorrer da mesma maneira. Assim utilizar um atributo *a*, que possua interação com um atributo *b*, e não utilizar o atributo *b* ou vice-versa pode enviesar o modelo.

Em seu trabalho Trendowicz e Münch (2009) [3] fizeram uma revisão literária dos atributos mais utilizados em trabalhos científicos e de experiências corporativas sobre estimativas de custo e esforço. Os autores consideraram que atributos relacionados à capacidade e experiência do time são os que mais influenciam medidas de produtividade, que por sua vez estão diretamente relacionadas com a medida de esforço conforme pode ser visto na figura 2 [3]. Trendowicz e Münch (2009) classificaram os atributos em sub grupos e classificaram sua relevância de acordo com sua utilização em estudos anteriores.

A escolha dos atributos, para a caracterização da base de projetos análogos, deve ser baseada nos dados disponíveis na base de conhecimento, e deve considerar a relação entre a influência dos atributos na estimativa de esforço e a dificuldade de mensuração. Deve também incluir ou excluir conjuntos de atributos que possuam interações entre si.

### III. RESULTADOS ESPERADOS

É possível comparar o esforço previsto com o esforço real e obter as métricas de erro relativo. Segundo Bardsiri, Jawawi e Khatibi (2014), as métricas *MRE* (magnitude do erro relativo), *MMRE* (média da magnitude do erro relativo), *MdRE*

(mediana da magnitude do erro relativo), *PRED* (percentual de previsão) e *MSE* (erro quadrado médio) têm sido utilizadas como medida de desempenho do ABE. Ainda segundo Bardsiri, Jawawi e Khatibi (2014), MMRE e PRED foram as métricas mais utilizadas pelos pesquisadores, com percentual de uso de 92% e 88% respectivamente.

Apesar disso não existe um consenso entre os pesquisadores sobre a melhor métrica de avaliação de desempenho, portanto faz-se necessário utilizar métricas que evitem resultados enviesados. Bardsiri, Jawawi e Khatibi (2014) e Kocaguneli *et al* (2012) recomendam a utilização de testes estatísticos como o de *Wilcoxon* para evitar o viés nos resultados. A partir da análise do erro é possível obter eficácia da técnica de acordo com os atributos escolhidos.

Ao variar a escolha dos atributos para caracterização, será possível analisar o comportamento da eficácia da técnica. Essa análise fornecerá dados para validação dos estudos feitos por [3], [7], [8], [9], [10]. A validação ou não desses estudos trará confirmações valiosas para o entendimento das estimativas baseadas em analogia, ou abrirá campo para novas indagações, assim propiciando e fomentando um melhor entendimento do modelo.

#### IV. CONCLUSÃO

O propósito desse trabalho é analisar o impacto dos atributos de projetos em modelos parametrizáveis de estimativa de projetos de software baseados em analogia. Com esse resultado espera-se obter uma visão pragmática da utilização de atributos de projetos em modelos de estimativa baseados em analogia, avaliando a escolha de cada atributo de acordo com sua influência na medida de eficácia do modelo.

O resultado desse trabalho irá propiciar um melhor entendimento das nuances entre os atributos e as estimativas de projetos de software baseadas em analogia. Esse entendimento servirá para melhorar a compreensão dos modelos de estimativa baseados em analogia, principalmente no que diz respeito à escolha dos atributos do modelo.

#### REFERÊNCIAS

- [1] BOEHM, B.; ABTS, C.; CHULANI, S. Software development cost estimation approaches—a survey. *Annals of software engineering*, Springer, v. 10, n. 1-4, p. 177–205, 2000.
- [2] TRENDOWICZ, A.; JEFFERY, R. *Software Project Effort Estimation: Foundations and Best Practice Guidelines for Success*. [S.l.]: Springer, 2014.
- [3] TRENDOWICZ, A.; MÜNCH, J. Factors influencing software development productivity—state-of-the-art and industrial experiences. *Advances in computers*, Elsevier, v. 77, p. 185–241, 2009.
- [4] KEUNG, J. Software development cost estimation using analogy: A review. In: *2009 Australian Software Engineering Conference*. [S.l.: s.n.], 2009. p. 327–336. ISSN 1530-0803.
- [5] SHEPPERD, M.; SCHOFIELD, C. Estimating software project effort using analogies. *IEEE Transactions on software engineering*, IEEE, v. 23, n. 11, p. 736–743, 1997.
- [6] WALKERDEN, F.; JEFFERY, R. An empirical study of analogy-based software effort estimation. *Empirical software engineering*, Springer, v. 4, n. 2, p. 135–158, 1999.
- [7] KITCHENHAM, B. A.; HUGHES, R. T.; LINKMAN, S. G. Modeling software measurement data. *IEEE Transactions on Software Engineering*, IEEE, v. 27, n. 9, p. 788–804, 2001.

- [8] SAUER, C.; GEMINO, A.; REICH, B. H. The impact of size and volatility on it project performance. *Communications of the ACM*, ACM, v. 50, n. 11, p. 79–84, 2007.
- [9] SHIM, J. et al. Coproduction in successful software development projects. *Information and Software Technology*, Elsevier, v. 52, n. 10, p. 1062–1068, 2010.
- [10] TRENDOWICZ, A. et al. Integrating human judgment and data analysis to identify factors influencing software development productivity. *e-Infomatica*, v. 2, n. 1, p. 47–69, 2008.
- [11] PREMRAJ, R. et al. An empirical analysis of software productivity over time. In: *IEEE. Software Metrics, 2005. 11th IEEE International Symposium*. [S.l.], 2005. p. 10–pp.
- [12] BARDSIRI, V. K.; JAWAWI, D. N. A.; KHATIBI, E. Towards improvement of analogy-based software development effort estimation: A review. *International Journal of Software Engineering and Knowledge Engineering*, World Scientific, v. 24, n. 07, p. 1065–1089, 2014.
- [13] KOCAGUNELI, E. et al. Exploiting the essential assumptions of analogy-based effort estimation. *IEEE Transactions on Software Engineering*, IEEE, v. 38, n. 2, p. 425–438, 2012.