# Classificação de Artigos de Engenharia de Software: uma Replicação Experimental Estendida

**Ramon Gomes Pereira**[1], **Alcemir Rodrigues Santos**[1]

[1]Universidade Estadual do Piauí (UESPI) – Campus Piripiri

ramongomes@prp.uespi.br, alcemir@prp.uespi.br

***Resumo.*** *Busca de informação científica em grandes volumes de dados não é uma tarefa trivial. A classificação automatizada de texto científico é uma forma de contornar o problema. Embora haja evolução dos algoritmos de classificação, pouco é feito sobre a classificação de textos de engenharia de software. As soluções identificadas apresentam resultados não conclusivos, o que abre espaço para estudos de replicação experimental em busca de novas evidências. Este artigo avalia a eficiência dos algoritmos (`Naive Bayes`, `J48` e `SVM`) na classificação de artigos de engenharia de software através de uma replicação estendida de estudo primário. Os resultados desta pesquisa apontaram um melhor desempenho, com ênfase para o `SVM`.*

## 1. Introdução

Há tempo que a recuperação de informação em bases de artigos científicos, outrora manual, depende de ferramentas de busca automatizadas. A busca simplificou, mas a enorme quantidade de artigos resultantes pode dificultar uma tarefa básica de pesquisa, que em muitos casos a primeira delas, é a revisão de literatura [Santos et al. 2015]. Mais que eficiência, a busca por reprodutibilidade da pesquisa deveria ser um objetivo da pesquisa científica em quaisquer áreas, visto que estudos sem essa propriedade podem ser vistas com pouca ou nenhuma significância [Popper 2005]. Particularmente, muito do que foi produzido de ciência em engenharia de *software*, e possivelmente em outras áreas, é impossível de ser reproduzido.

Determinar se um estudo é reprodutível automaticamente ou não, não é uma questão trivial. Existem detalhes que precisam de avaliação de um especialista para a determinação da viabilidade e/ou a disponibilidade de alguns recursos necessários na reprodução do estudo, no entanto de maneira a simplificar tal processo o uso de inteligência artificial pode automatizar parte desta tarefa. Por exemplo, em uma fase anterior é preciso i) identificar a que área do conhecimento o artigo pertence e ii) identificar a presença de estudo empírico a ser replicado. [Woodson et al. 2018] destacam exatamente as dificuldades envolvidas na reprodutibilidade. Os autores buscaram separar trabalhos de engenharia de requisitos e também aqueles que continham alguma espécie de estudo empírico. Apesar de complexa, tal conjuntura pode ser tomada como referência para a construção de uma nova abordagem de melhor eficácia. Neste contexto, construiu-se aqui uma replicação comparativa e estendida do estudo experimental de [Woodson et al. 2018].

Para Wohlin e seus colegas [Wohlin et al. 2012] a replicação experimental implica em repetir a investigação sob semelhantes condições, e na capacidade de um teste ou experimento ser reproduzido ou replicado com precisão. Para o presente trabalho, teve-se

como objetivo a replicação do trabalho de Woodson e seus colegas na classificação de artigos de Engenharia de Software, para isso, utilizou-se aqui o aprendizado indutivo supervisionado [Monard and Baranauskas 2003], onde um conjunto de exemplos de treinamento é apresentado ao algoritmo de aprendizado, contendo os dados classificados e rotulados à priori.

As contribuições deste trabalho, podem ser enumeradas da seguinte maneira. **Confirmação de resultados:** este estudo corrobora os resultados encontrados no estudo original [Woodson et al. 2018]. **Extensão de evidências:** o novo algoritmo analisado (SVM) teve resultados similares aos daqueles do estudo comparado.

Este artigo encontra-se dividido da seguinte forma: A Seção 2, consiste em um resumo da literatura. A Seção 3 apresenta o planejamento e a execução do estudo experimental de replicação. A Seção 4 apresenta os resultados alcançados com o estudo e a Seção 5 discute os mesmos à luz da interpretação dos autores e da comparação com o estudo original. Por sua vez, a Seção 6 trata das ameaças à validade do estudo. Por fim, a Seção 7 conclui este relatório e aponta direções para trabalhos futuros.

## 2. Trabalhos Relacionados

Esta seção apresenta uma lista não-exaustiva de trabalhos que consideramos serem relacionados a este [Lagerkrants and Holmström 2016, Ahmed et al. 2020, Dosciatti et al. 2013, Woodson et al. 2018]. No trabalho de Lagerkrants e Holmström [Lagerkrants and Holmström 2016], foi realizado um experimento para selecionar artigos de interesse de um usuário usando classificação automática. Os autores descobriram que para conjuntos de dados com tamanho maior que 50 artigos não houve aumento significativo na confiança da classificação. Um dos algoritmos sob avaliação neste trabalho foi utilizado naquele estudo, porém com outro propósito.

Ahmed e seus colegas [Ahmed et al. 2020] classificam artigos científicos semiestruturados usando diferentes técnicas de classificação supervisionada. De forma semelhante a este trabalho, Ahmed e seus colegas utilizaram aprendizado de máquina para classificar automaticamente um conjunto de dados textuais, este trabalho distingue-se do deles uma vez que o conjunto de dados utilizado é estruturado e o deles semiestruturados. Classificação automática também é utilizada para a identificação de emoções [Dosciatti et al. 2013]. O trabalho de Dosciatti e seus colegas está relacionado a esta pesquisa, uma vez que usam aprendizado supervisionado na classificação de textos.

O trabalho de Woodson e seus colegas [Woodson et al. 2018], serviu de inspiração para esta pesquisa. O mesmo aborda a classificação automática de artigos de subáreas da engenharia de *software* utilizando os algoritmos de `Naive Bayes` e `J48`, e avalia o desempenho dos mesmos utilizando as métricas de *precisão*, *revocação*, *acurácia*, e $f_1$-*score*. Para esta pesquisa, tomou-se como base os algoritmos de classificação utilizados pela pesquisa original, assim como a base de dados utilizada pelos mesmos.

## 3. Estudo Experimental

Este estudo experimental replica e estende o trabalho primário [Woodson et al. 2018], com a adição de um novo algoritmo de classificação (`SVM`). Avalia-se a eficiência (com relação à *revocação*, *precisão*, $f_1$-*score* e *acurácia*) dos algoritmos de classificação

(`Naive Bayes`, `J48` e `SVM`) no contexto de artigos de engenharia de software. Mais especificamente, busca-se respostas para as seguintes perguntas de pesquisa (QP):

**QP1:** É possível replicar os resultados alcançados no estudo original [Woodson et al. 2018]?

**QP2:** Como o `SVM` se comporta em comparação com os outros algoritmos avaliados?

Para responder **QP1** e **QP2**, planejou-se o estudo tomando como referência os parâmetros de Wohlin e seus colegas [Wohlin et al. 2012]. A Figura 1 apresenta as etapas conduzidas na realização do estudo, a saber: *(i)* coleta dos dados; *(ii)* tratamento dos dados; *(iii)* classificação dos estudos; *(iv)* coleta dos resultados; e por fim a *(v)* análise dos resultados. Cada uma das etapas será detalhada a seguir.

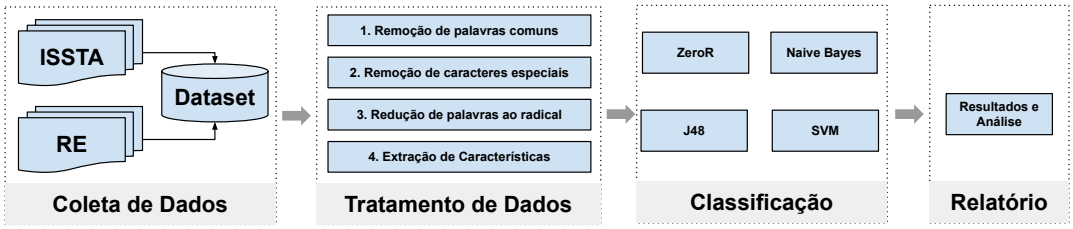


**Figura 1. Etapas executadas para a classificação dos artigos.**

### 3.1. Coleta dos dados

Foram coletados artigos de duas conferências, a Conferência Internacional de Engenharia de Requisitos do IEEE (RE), que apresenta somente artigos relacionados a engenharia de requisitos e o IEEE Simpósio Internacional de Teste e Análise de Software (ISSTA), que disponibiliza conteúdo relacionado a não engenharia de requisitos. Além disso, ambas possuem artigos empíricos, o que justifica a escolha dessas bases de dados para análise. Para representar RE foram coletados artigos dos anos 2000, 2005, e 2015, para ISSTA, os anos de 2000, 2004, e 2015.

Após a coleta de dados foi realizada uma classificação dos artigos de forma manual, para assim serem rotulados como sendo da subárea de Engenharia de Requisitos ou não, bem como se continham estudos empíricos ou não. A Tabela 1 apresenta o resultado dessa classificação.

**Tabela 1. Classificação manual dos artigos.**

| Fórum | Ano | AE | ANE | AER | ANER | TOTAL |
|---|---|---|---|---|---|---|
| RE | 2000 | 1 | 12 | 13 | 0 | 13 |
| | 2005 | 14 | 30 | 44 | 0 | 44 |
| | 2015 | 14 | 20 | 34 | 0 | 34 |
| ISSTA | 2000 | 8 | 14 | 0 | 22 | 22 |
| | 2004 | 16 | 8 | 0 | 24 | 24 |
| | 2015 | 30 | 12 | 0 | 42 | 42 |

**AE:** Artigos Empíricos; **ANE:** Artigos Não-Empíricos; **AER:** Artigos Engenharia de Requisitos; **ANER:** Artigos Não Engenharia de Requisitos.

### 3.2. Tratamento dos dados

Antes de utilizar os algoritmos de classificação para processar os artigos selecionados da base de dados, se faz necessário, como é praxe nos estudos de processamento de linguagem natural, uma etapa de pré-processamento para remover ruído. Nessa etapa, foram utilizados três algoritmos de pré-processamento, *remoção de palavras de parada*, *remoção de caracteres especiais* e *redução ao radical das palavras*.

Com o texto pré-processado é feito a extração de características. Estas características foram utilizadas para ensinar os algoritmos de classificação a classificar estes textos científicos específicos. Para extração de características usamos o método *apply_features* do NLTK para aplicar os recursos ao classificador SVM e os métodos *IDFTransform* e *TFTransform* para os algoritmos implementados no WEKA.

### 3.3. Classificação dos Artigos

Para a classificação automática dos artigos, foram usados quatro algoritmos de classificação: `ZeroR`, `Naive Bayes`, `J48` [Woodson et al. 2018] e *Máquina de vetores de suporte* (`SVM`) [Vapnik 2000]. O primeiro é o método de classificação mais simples, que depende do alvo e ignora todos os preditores. Ele é útil para determinar o desempenho da linha de base como referência para outros métodos de classificação. O segundo é um classificador probabilístico baseado no teorema de *Bayes*, com as premissas de independência entre preditores. O terceiro é uma árvore de decisão que por sua vez é um sistema de suporte à decisão que usa decisões de gráfico semelhantes a uma árvore e seu possível efeito posterior, incluindo resultados de eventos aleatórios, custos de recursos e utilidade. O quarto e último, executa a classificação localizando o hiperplano que maximiza a margem entre as duas classes. Os casos que definem o hiperplano são os vetores de suporte. Este algoritmo é ideal para classes não sobrepostas.

### 3.4. Métricas

Para medir os resultados da classificação pelos algoritmos selecionados, utilizamos as métricas *precisão*, *revocação*, $f_1$-*score* e *acurácia* [Woodson et al. 2018]. Enquanto a *precisão* mede a fração da classificação automática correta, a *revocação* mede a fração da classificação manual que a classificação automática conseguiu acertar. De forma complementar, o $f_1$-*score* faz uma média harmônica das duas métricas anteriores. Por fim, a *acurácia* mede a fração de acertos dos algoritmos, seja por classificar corretamente, seja por não classificar errado.

## 4. Resultados

Nesta seção são apresentados os resultados obtidos no desenvolvimento deste artigo. Assim como na pesquisa original [Woodson et al. 2018], os modelos foram avaliados usando validação cruzada, e o método de validação *k-fold*. O método consiste em dividir a base de dados em *k-partes*, usando $k - 1$ partes para treinamento e a parte restante para teste, realizando esse processo $k$ vezes [Schneider 1997]. Assim como Woodson et al.(2018) o presente trabalho utilizou validação cruzada dividida em 10, 20, 30 e 40 *folds*. Além disso, para realizar a análise dos resultados foram coletadas as médias das execuções.

A Figura 2 apresenta os resultados quanto a *precisão* da classificação (a) dos artigos empíricos e (b) dos artigos de requisitos. De maneira análoga, a Figura 3 apresenta

os resultados da *revocação*, a Figura 4 apresenta a $f_1$-*score* e por fim a Figura 5 mostra a *acurácia*.

(a) Artigos Empíricos (b) Artigos de Requisitos

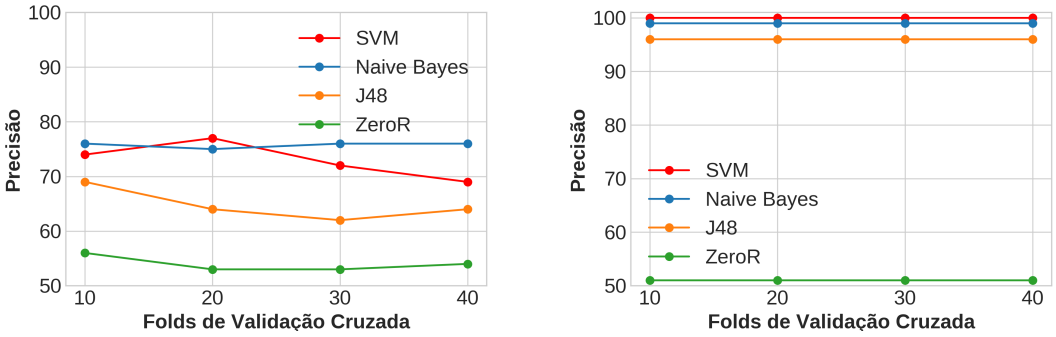


**Figura 2. Precisão da classificação de artigos.**

(a) Artigos Empíricos (b) Artigos de Requisitos

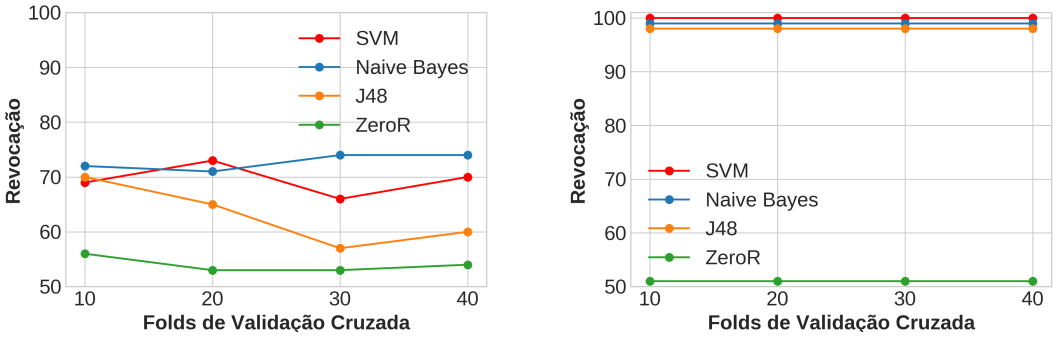


**Figura 3. Revocação da classificação de artigos.**

(a) Artigos Empíricos (b) Artigos de Requisitos

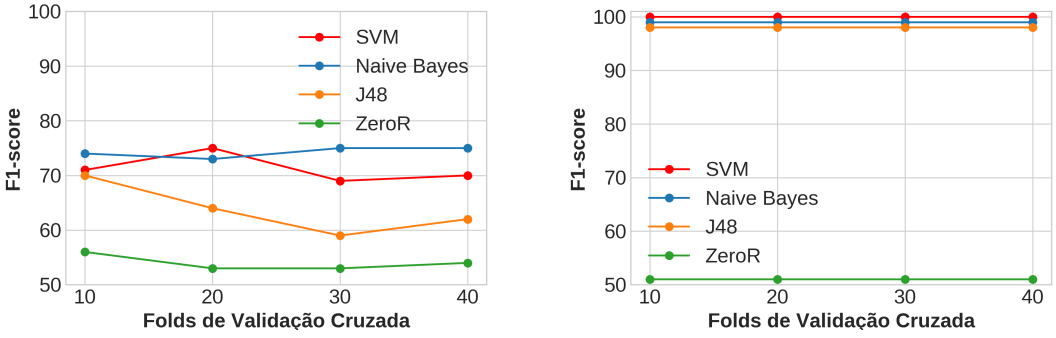


**Figura 4. $F_1$-score da classificação de artigos.**

(a) Artigos Empíricos (b) Artigos de Requisitos

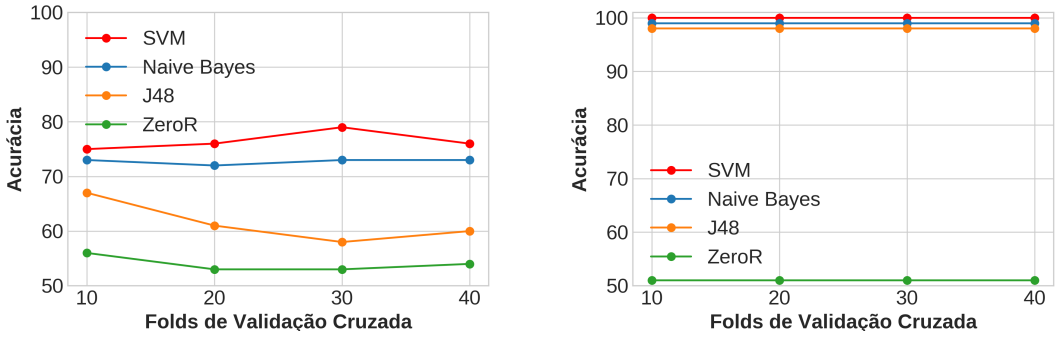


**Figura 5. Acurácia da classificação de artigos.**

O `Naive Bayes` apresentou uma *acurácia* de em média 73% para classificar empíricos e cerca de 99% para classificar requisitos. Por sua vez o classificador `J48` teve uma *acurácia* de 61% em média na classe de empíricos e 98% para a classe de requisitos. Por fim, o algoritmo `SVM` apresentou resultados de em média 76% de *acurácia* para empíricos e 99% para requisitos. Com base nisso, nota-se que os algoritmos de classificação tiveram um melhor desempenho na classe de requisitos do que na classe de empíricos. Isso se dá devido a especificidade do vocabulário nessa subárea, gerando assim uma maior extração de características e facilitando a classificação dessa classe pelos algoritmos.

## 5. Análise e Comparação com trabalho original

### QP1: É possível replicar os resultados alcançados no estudo original [Woodson et al. 2018]?

Os resultados aqui obtidos foram comparados aos da pesquisa original [Woodson et al. 2018], resultados esses que se mostraram melhores em relação ao estudo comparado, em ambas as classes. A Tabela 2 apresenta uma comparação das médias das métricas de avaliação utilizadas aqui e no trabalho original [Woodson et al. 2018]. Os campos indicados com um '-' indicam a inexistência da análise do algoritmo no artigo original, já os '?' indicam que apesar do algoritmo ter sido analisado, os dados não foram disponibilizados.

Na classe de empíricos os classificadores `Naive Bayes` e `J48` superaram o método de linha base demonstrando um ótimo desempenho. Os mesmos se mantiveram instáveis em relação ao número de *folds* de validação. Nessa classe os métodos aplicados foram superiores aos da pesquisa original [Woodson et al. 2018]. O classificador `Naive Bayes` superou o deles em cerca de 20% de *acurácia*, como também se mostrou superior nas outras métricas. Quanto ao classificador `J48`, demonstrou cerca de 7% de *acurácia* a mais em relação aos do trabalho comparado, e também foi superior em todas as outras métricas.

Quanto a classe de requisitos os algoritmos demonstraram um desempenho excelente, superando o método de linha base em mais de 40% de *acurácia*. Os classificadores se mantiveram estáveis em todos os *folds* de validação. O classificador `Naive Bayes`

**Tabela 2. Comparação de resultados com relação às médias de cada métrica.**

| | | Empíricos | | | Requisitos | | |
|---|---|---|---|---|---|---|---|
| **Métrica** | **Estudo** | **Bayes** | **J48** | **SVM** | **Bayes** | **J48** | **SVM** |
| *precisão* | [Woodson et al. 2018] | 64% | 50% | - | ? | ? | - |
| | Esta replicação | 75% | 64% | 73% | 99% | 96% | 99% |
| *revocação* | [Woodson et al. 2018] | ? | 50% | - | ? | 90% | - |
| | Replicação | 72% | 63% | 69% | 99% | 99% | 99% |
| $f_1$-*score* | [Woodson et al. 2018] | 50% | ? | - | 90% | ? | - |
| | Replicação | 74% | 63% | 71% | 99% | 98% | 99% |
| *acurácia* | [Woodson et al. 2018] | 53% | 54% | - | 90% | 89% | - |
| | Replicação | 73% | 61% | 76% | 99% | 98% | 99% |

superou o deles em cerca de 9% de *acurácia*, assim como também se mostrou superior em todas as outras métricas. Quanto ao classificador `J48`, superou o deles em cerca de 9% de *acurácia*, demonstrando também um maior desempenho em relação as outras métricas.

**QP2: Como o `SVM` se comporta em comparação com os outros algoritmos avaliados?**

Assim como os algoritmos `Naive Bayes` e `J48`, o classificador `SVM` também se mostrou superior ao método de linha base em ambas as classes. O `SVM` também se mostrou superior aos métodos do estudo comparado [Woodson et al. 2018], apresentando um melhor desempenho tanto na classe de empíricos quanto na de requisitos.

Em comparação com o `Naive Bayes` e `J48`, na classe de empíricos, o classificador `SVM` mostrou um melhor desempenho em relação aos outros quanto a *acurácia*, superando o classificador `Naive Bayes` com em média 3% a mais e o `J48` com cerca de 15% a mais. Quanto as outras métricas, ambos os algoritmos demonstraram equivalência nos resultados. Quanto a classe de requisitos, o classificador `SVM` se comportou surpreendentemente bem, obtendo resultados equivalentes aos classificadores, tanto na *acurácia* como nas demais métricas de avaliação.

## 6. Ameaças à Validade

Embora estudos empíricos tenham que ser exaustivamente planejados para aumentar a confiança do estudo, nem sempre é possível contornar todas as variáveis existentes. Este estudo não foge à realidade. Desta forma, faz-se necessário a discussão de algumas ameaças à validade que podem ser identificadas neste estudo. São elas: *(i)* para estudos de classificação de dados, o ideal seria executar o estudo com a maior base de dados possível, no entanto, por se tratar de uma replicação, resolvemos utilizar a mesma base do estudo original, sabendo que a mesma não tem o tamanho ideal; *(ii)* os algoritmos utilizados pela pesquisa original foram executados no WEKA, e não foram disponibilizadas as configurações utilizadas para a execução;

## 7. Conclusão e Trabalhos Futuros

O estudo de replicação experimental conduzido mostrou os seguintes resultados: os métodos utilizados nessa pesquisa apresentaram um desempenho satisfatório, superando os resultados da pesquisa original [Woodson et al. 2018]. Percebeu-se que os algoritmos `Naive Bayes` e `SVM` demonstraram um melhor desempenho em relação ao `J48`. Pode-se perceber também que os algoritmos tiveram um melhor desempenho na classificação dos artigos de requisitos, por conta do vocabulário especifico da área. Embora os métodos utilizados se mostrem propícios para a classificação de textos, existe um espaço para melhorar os mesmos.

Com base nisso, em relação aos trabalhos futuros, pretende-se ampliar a base de artigos e utilizar outros algoritmos/métodos de classificação para efeito de comparação e generalização dos resultados.

## Referências


Ahmed, E., Ashraf, S., and Shahzad, W. (2020). An effective way to enhance classifications for the semi-structured research articles. *University of Sindh Journal of Information and Communication Technology*, 4(1):45–51.

Dosciatti, M. M., Ferreira, L., and Paraiso, E. (2013). Identificando emoções em textos em português do brasil usando máquina de vetores de suporte em solução multiclasse. *ENIAC-Encontro Nacional de Inteligência Artificial e Computacional. Fortaleza, Brasil*.

Lagerkrants, E. and Holmström, J. (2016). Using machine learning to classify news articles.

Monard, M. C. and Baranauskas, J. A. (2003). Conceitos sobre aprendizado de máquina. *Sistemas inteligentes-Fundamentos e aplicações*, 1(1):32.

Popper, K. (2005). *The Logic of Scientific Discovery. This Edotion Published*, volume 2. New York: The Taylor & Francis e-Library.

Santos, J. A. M., Santos, A. R., and Mendonça, M. G. (2015). Investigating bias in the search phase of software engineering secondary studies. In *CIbSE*, page 488.

Schneider, J. (1997). Cross validation, feb 7, 1997. *URL: https://www.cs.cmu.edu/˜schneide/tut5/node42.html (visited on 06/07/2019)*.

Vapnik, V. N. (2000). Direct methods in statistical learning theory. In *The nature of statistical learning theory*, pages 225–265. Springer.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., and Wesslén, A. (2012). *Experimentation in software engineering*. Springer Science & Business Media.

Woodson, C., Hayes, J. H., and Griffioen, S. (2018). Towards reproducible research: automatic classification of empirical requirements engineering papers. In *Proceedings of the ACMSE 2018 Conference*, pages 1–7.