Análise de Métodos de Ensemble Learning para Detecção de Spam

Lucas Marchesan da Silva[1], **Gabriel Stiegemeier**[1],
Rafaela Savian Colvero de Oliveira[2], **Marcia Henke**[2]

[1]Curso de Engenharia de Computação, Ciência da Computação
Universidade Federal de Santa Maria (UFSM) – Santa Maria, RS – Brazil.

[2]Tecnologia em Redes de Computadores.
Colégio Técnico Industrial de Santa Maria - CTISM
Universidade Federal de Santa Maria (UFSM) – Santa Maria, RS – Brazil.

{lucas.marchesan, gabriel.stiegemeier}@acad.ufsm.br

{rafaela.oliveira, henke}@redes.ufsm.br

Abstract. *The growing volume of spam and malicious emails demands the continuous evolution of filtering techniques. This work presents a comparative analysis of different ensemble learning architectures (Voting, Stacking and Bagging) applied to the Spambase dataset. Experiments conducted on the WEKA platform demonstrate the superiority of layered approaches. The Stacking architecture outperformed the Random Forest baseline, which suggest that stacking architectures are a promising approach for spam detection with greater robustness and accuracy.*

Resumo. *O crescente volume de spam e e-mails maliciosos exige a contínua evolução das técnicas de detecção. Este trabalho apresenta uma análise comparativa de diferentes arquiteturas de ensemble learning (Voting, Stacking e Bagging) aplicadas à base de dados Spambase. Os experimentos, conduzidos na plataforma WEKA, demonstram a superioridade das abordagens em camadas. A arquitetura de Stacking superou o baseline do Random Forest, o que sugere que arquiteturas stacking são uma abordagem promissora para detecção de spam com mais robustez e precisão.*

1. Introdução

O e-mail é um vetor comum para ameaças cibernéticas, oferecendo um ponto de entrada direto para redes corporativas. Um relatório de 2025 da Barracuda [Barracuda Networks 2025] indica que um em cada quatro e-mails é malicioso ou spam, o que evidencia a necessidade de aprimoramentos em sistemas anti-spam. Técnicas de Aprendizado de Máquina (AM) são amplamente utilizadas para mitigar esse problema [Charanarur et al. 2023, Zhang, Chenwei 2025]. Estudos anteriores, como o de [Bassiouni et al. 2018], demonstraram a superioridade do Random Forest para a detecção de spam no conjunto de dados Spambase. Nosso trabalho aprofunda essa investigação, explorando diferentes estratégias de ensemble learning.

Os resultados de [Bassiouni et al. 2018] motivaram os estudos inciais de um projeto de pesquisa que visa explorar o potencial de AM na área de segurança de redes, com

foco na detecção de ataques cibernéticos, com uma investigação mais profunda. Nesse contexto, o presente estudo busca explorar a aplicação de diferentes estratégias de ensemble. A Seção 2 descreve a metodologia proposta, a Seção 3 apresenta os resultados experimentais, e a Seção 4 conclui o trabalho e aponta possíveis direções para pesquisas futuras.

2. Metodologia

2.1. Conjunto de Dados

O conjunto de dados utilizado foi o Spambase, disponível no repositório UCI [Hopkins and Suermondt 1999]. Ele consiste em 4.601 amostras de e-mails, cada uma descrita por 57 atributos numéricos contínuos que representam a frequência de palavras e caracteres específicos. De modo geral, o conjunto é levemente desbalanceado, com 39,4% de amostras de spam e 60,6% de e-mails legítimos (ham).

2.2. Workbench WEKA

Os experimentos foram conduzidos utilizando o software *Waikato Environment for Knowledge Analysis* (WEKA) [Frank et al. 2016]. Esta ferramenta foi escolhida por ser uma plataforma consolidada e de código aberto, oferecendo uma ampla biblioteca de algoritmos de aprendizado de máquina. A interface gráfica Explorer foi utilizada para realizar todo o pipeline experimental: carregamento do conjunto de dados, configuração dos métodos de *ensemble* e coleta dos resultados por meio de validação cruzada 10-fold, conforme ilustrado na Figura 1.

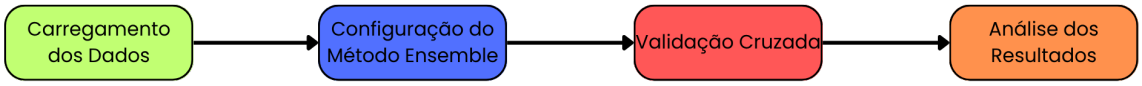

Figura 1. Protocolo experimental através de um Pipeline aplicado no WEKA

2.3. Algoritmos

Conforme motivado na Seção 1, este trabalho foca em métodos de *ensemble*, que combinam múltiplos classificadores para mitigar fraquezas individuais e melhorar robustez e acurácia.

Três arquiteturas de *ensemble* com diferentes níveis de complexidade foram implementadas e analisadas[Zhou 2012]:

- **Voting:** Dois comitês heterogêneos utilizando *Soft Voting* sobre as probabilidades médias previstas. Um combinou *Random Forest* (RF) e *Logistic Regression* (LR), e o outro RF com *k-NN*.
- **Stacking:** Abordagem hierárquica com RF e *k-NN* como classificadores base (Nível 0) e LR como meta-modelo (Nível 1), que aprende a combinar as previsões base.
- **Bagging:** *Bootstrap Aggregating* aplicado ao RF com 15 iterações, avaliando o efeito do re-bagging em um modelo já baseado em *ensemble*.

Para estabelecer uma linha de base para comparação direta, a parametrização dos classificadores individuais (RF, LR e *k-NN*) seguiu a mesma configuração usada no estudo de [Bassiouni et al. 2018], cujos principais parâmetros e métricas de desempenho estão resumidos na Tabela 1.

Tabela 1. Parametrização de classificadores individuais usados no baseline

Classificador	Parâmetros	Precisão	Recall	F1	Sens.	Spec.	Acc (%)
Random Forest	`Trees = 100, Seed = 1`	0.955	0.955	0.954	0.953	0.957	95,46
Logistic Regression	`MaxIter = -1, Ridge =` 1×10^{-8}	0.924	0.924	0.924	0.928	0.918	92,41
k-NN	`K = 1, Search = Linear`	0.908	0.908	0.908	0.921	0.887	90,78

2.4. Métricas de Avaliação

Para garantir comparação direta com o baseline, a avaliação dos modelos propostos seguiu a mesma abordagem de [Bassiouni et al. 2018]. Foram utilizadas sete métricas principais para avaliar o desempenho dos classificadores: Acurácia(Acc), Precisão, *Recall*, F1-*Score*, Sensibilidade e Especificidade.

3. Resultados e Discussão

Esta seção apresenta e discute os resultados obtidos das arquiteturas de *ensemble learning* propostas. A Tabela 2 sintetiza as métricas de desempenho, comparando os modelos desenvolvidos com o classificador *Random Forest*, adotado como *baseline*.

Tabela 2. Resultados das arquiteturas de ensemble propostas neste artigo.

Classificador	Precisão	Recall	F1	Sens.	Spec.	Acc (%)
Random Forest (Baseline)	0.955	0.955	0.954	**0.953**	0.957	95.46
Voting (RF+Logistic)	0.948	0.948	0.948	0.925	0.963	94.78
Voting (RF+k-NN)	0.909	0.909	0.909	0.879	0.929	90.89
Bagging (c/ Random Forest, 15 it.)	0.954	0.954	0.954	0.928	0.971	95.39
Stacking (RF+k-NN – Meta: Logistic)	**0.959**	**0.959**	**0.959**	0.940	**0.972**	**95.94**

Os resultados da Tabela 2 mostram que a arquitetura baseada em *Stacking* obteve o melhor desempenho geral. Utilizando *Random Forest* e *k-NN* como classificadores base, combinados com *Logistic Regression* como meta-modelo, o *Stacking* foi o único método que superou consistentemente o *baseline*, alcançando a maior acurácia (95,94%). Esse resultado reforça a hipótese de que arquiteturas hierárquicas, capazes de aprender a ponderar as previsões de seus modelos constituintes, oferecem maior efetividade na tarefa de detecção de spam.

As demais arquiteturas de *ensemble*, embora robustas, tiveram desempenho inferior ao *baseline*. Esse resultado pode ser atribuído à inclusão de classificadores com desempenho individual relativamente inferior nos comitês. No caso do *Voting*, por exemplo, o desempenho isolado do *Random Forest* (95,46%) foi reduzido pela influência da *Logistic Regression* (resultando em 94,78%) e, de forma mais acentuada, do *k-NN* (resultando em 90,89%). Esse fenômeno demonstra que, mesmo utilizando média das probabilidades (*Soft Voting*), o potencial do classificador mais eficaz pode ser limitado pelo impacto de modelos menos precisos.

A análise detalhada das métricas de sensibilidade e especificidade revela um padrão consistente: com exceção do *Voting* (RF+*k-NN*), todas as arquiteturas de *ensemble* apresentaram um trade-off entre esses dois indicadores em relação ao *baseline*. O modelo *Random Forest* alcançou a maior sensibilidade (0,953), enquanto *Stacking* (Sens. 0,940 / Spec. 0,972), *Bagging* (Sens. 0,928 / Spec. 0,971) e *Voting* (RF+LR) (Sens. 0,925 / Spec. 0,963) reduziram ligeiramente a taxa de detecção em favor de maior especificidade. Esse

padrão sugere uma tendência a decisões mais conservadoras, valorizando a redução de falsos positivos. Esse comportamento é especialmente relevante em aplicações de detecção de spam, onde classificar incorretamente uma mensagem legítima como indesejada é, na maioria dos cenários, um erro mais crítico do que permitir que uma mensagem de spam chegue à caixa de entrada.

4. Conclusão

Este trabalho propôs diferentes arquiteturas de *ensemble learning* para detecção de spam utilizando o conjunto de dados Spambase. Os resultados mostraram que ensembles hierárquicos, que aprendem combinações ponderadas dos modelos base, superam métodos mais simples como o *Random Forest*.

O modelo de *Stacking*, que combina *Random Forest* e *k-NN* como classificadores base e *Logistic Regression* como meta-modelo, alcançou a maior acurácia (95,94%) e destacou métricas como a especificidade, importante para reduzir falsos positivos.

As limitações incluem a dependência do conjunto Spambase que está desatualizado e que pode não refletir características atuais do spam, como engenharia social e uso de imagens. Além disso, os parâmetros foram fixos, sem otimização extensa de hiperparâmetros.

Este estudo em estágio inicial visa explorar o potencial da Aprendizagem de Máquina aplicada à segurança de redes, com foco na detecção de ataques cibernéticos. Como trabalhos futuros, pretende-se ampliar a avaliação das arquiteturas propostas utilizando conjuntos de dados mais atuais e diversificados, realizar uma otimização aprofundada dos hiperparâmetros e analisar os custos computacionais envolvidos, visando verificar a viabilidade da implementação desses modelos em sistemas de detecção de spam em tempo real.

Referências

Barracuda Networks (2025). 2025 email threats report: Key findings about the evolution of email-based threats. Relatório Técnico. Acessado em: 05 de agosto de 2025.

Bassiouni, M., Shafaey, M., and El-Dahshan, E.-S. (2018). Ham and spam e-mails classification using machine learning techniques. *Journal of Applied Security Research*, 13:315–331.

Charanarur, P., Jain, H., Gundu, S., Samanta, D., Singh Sengar, S., and Hewage, C. (2023). Machine-learning-based spam mail detector. *SN Computer Science*, 4.

Frank, E., Hall, M. A., and Witten, I. H. (2016). The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", Fourth Edition.

Hopkins, Mark, R.-E. F. G. and Suermondt, J. (1999). Spambase. UCI Machine Learning Repository.

Zhang, Chenwei (2025). Enhancing spam filtering: A comparative study of modern advanced machine learning techniques. *ITM Web Conf.*, 70:04013.

Zhou, Z.-H. (2012). *Ensemble Methods: Foundations and Algorithms*. Chapman & Hall/CRC, 1st edition.