Analysing a New Experimental Design Impact on the Performance of FlexCon-CE

Renan M. R. A. Costa[1], **Luiz M. S. Silva**[1], **Arthur C. Gorgônio**[2]
Flavius L. Gorgônio[3], **Karliane M. O. Vale**[3]

[1]Laboratório de Inteligência Computacional Aplicada a Negócios (LABICAN)
Universidade Federal do Rio Grande do Norte (UFRN)
Rua Joaquim Gregório, 296 – 59.300-000 – Caicó – RN – Brasil

[2]Departamento de Informática e Matemática Aplicada (DIMAP)
Universidade Federal do Rio Grande do Norte (UFRN)
Campus Universitário – Lagoa Nova – 59.078-970 – Natal – RN – Brasil

[3]Departamento de Computação e Tecnologia (DCT)
Universidade Federal do Rio Grande do Norte (UFRN)
Rua Joaquim Gregório, 296 – 59.300-000 – Caicó – RN – Brasil

{renan.costa.117,luiz.santos.090, arthur.gorgonio.099}@ufrn.edu.br

{flavius.gorgonio,karliane.vale@}ufrn.br

Abstract. *Semi-supervised learning is a subfield of machine learning that explores the combined use of labelled and unlabeled instances, with the latter being more numerous than the former. This type of learning is essentially an intersection of supervised and unsupervised learning. Self-training stands out among the various algorithms used in this context due to its wide application in the literature. Over the years, several variants of Self-training have been developed to enhance its performance, including FlexCon-CE, which serves as the basis for this work. FlexCon-CE is an algorithm that employs classifier committees to select and label unlabelled instances, integrating them into the labelled dataset. This study aims to analyse the performance of FlexCon-CE by expanding the number of analysed datasets, adding performance evaluation metrics and varying the percentages of initially labelled instances. The results show that FlexCon-CE performs well regardless of the experimental design and across different datasets.*

Resumo. *O aprendizado semissupervisionado é um subcampo do aprendizado de máquina que explora o uso combinado de instâncias rotuladas e não rotuladas, sendo estas últimas mais numerosas que as primeiras. Este tipo de aprendizado é essencialmente uma interseção do aprendizado supervisionado e não supervisionado. O Self-training se destaca dentre os diversos algoritmos utilizados nesse contexto devido à sua ampla aplicação na literatura. Ao longo dos anos, diversas variantes do Self-training foram desenvolvidas para melhorar seu desempenho, incluindo o FlexCon-CE, que serve de base para este trabalho. O FlexCon-CE é um algoritmo que emprega comitês de classificadores para selecionar e rotular instâncias não rotuladas, integrando-as ao conjunto de dados rotulado. Este estudo tem como objetivo analisar o desempenho do FlexCon-CE*

ampliando o número de bases de dados analisados, adicionando métricas de avaliação de desempenho e variando os percentuais de instâncias inicialmente rotuladas. Os resultados mostram que o FlexCon-CE tem um bom desempenho, independentemente do design experimental e em diferentes conjuntos de dados.

1. Introdução

O aprendizado de máquinas (AM) refere-se ao desenvolvimento de programas e algoritmos computacionais capazes de melhorar seu desempenho em tarefas específicas à medida que acumulam experiência [Monard and Baranauskas 2003]. Tradicionalmente, o AM foi categorizado em duas abordagens principais com base no grau de supervisão empregado durante o treinamento: aprendizado supervisionado e não supervisionado. No aprendizado supervisionado, durante o treinamento, os algoritmos recebem como entrada instâncias que contêm a informação de saída desejada, representando a classe a que cada instância pertence. Em contraste, o aprendizado não supervisionado trabalha com dados não rotulados, explorando padrões subjacentes para alcançar uma compreensão ou segmentação eficaz dos dados [Monard and Baranauskas 2003] [Sanches 2003].

Com o avanço da área surgiu um novo tipo de aprendizado, conhecido como semissupervisionado (SSL), que combina elementos dos dois anteriores utilizando bases de dados em que apenas uma parte das instâncias é rotulada [Chapelle et al. 2009]. Diversos algoritmos têm sido desenvolvidos para lidar com esse tipo de dado, destacando-se entre eles o Self-training. Este algoritmo automatiza o processo de treinamento, começando com um pequeno conjunto de dados rotulados, usado para treinar um classificador inicial. Este classificador prediz os rótulos das instâncias não rotuladas e aquelas com maior confiança na predição são incorporadas ao conjunto de treinamento. O processo se repete até que todas as instâncias sejam rotuladas [Yarowsky 1995].

Apesar de sua popularidade, o Self-training original enfrenta desafios de implementação, especialmente na determinação dos critérios para a inclusão de novas instâncias no conjunto de dados rotulados. Em resposta a essas limitações, várias abordagens foram propostas ao longo dos anos, visando melhorar a eficácia e a confiabilidade do processo de rotulagem [Li et al. 2024a] [Vale et al. 2021] [Gorgônio et al. 2019] [Rodrigues et al. 2014]. Dentre estas contribuições, destaca-se o algoritmo FlexCon-CE (Flexible Confidence with Classifier Ensembles) [Medeiros et al. 2023], que é uma evolução do FlexCon-C (Flexible Confidence with Classifier), desenvolvido por [Vale et al. 2018]. O FlexCon-CE aprimora o desempenho de seu antecessor ao introduzir comitês de classificadores para a seleção e rotulagem de instâncias, conforme será detalhado nas seções subsequentes.

Dado o potencial do FlexCon-CE, este estudo visa avaliar o desempenho desse algoritmo sob um novo design experimental. Para tanto, foram adicionadas aos experimentos sete novas bases de dados, uma métrica de avaliação de desempenho e as porcentagens de instâncias inicialmente rotuladas foram ajustadas para valores intermediários em relação aos testados por [Medeiros et al. 2023]. O artigo está estruturado da seguinte forma: a seção dois apresenta os aspectos teóricos relevantes, incluindo aprendizado semissupervisionado, FlexCon-CE e comitês de classificadores. A seção três discute trabalhos relacionados a esta pesquisa. Os aspectos metodológicos são abordados na seção quatro, e os resultados são discutidos na seção cinco. Finalmente, a seção seis inclui as considerações finais e

sugestões para pesquisas futuras.

2. Aspectos Teóricos

2.1. Aprendizado Semissupervisionado

Em tarefas de classificação do mundo real, é possível encontrar conjuntos de dados em que apenas uma parte dos dados são rotulados, enquanto o restante não possui rótulo. Sendo assim, o mecanismo de aprendizagem semissupervisionada se propõe a tratar dados com essas características objetivando alcançar melhor classificação [Wang et al. 2016]. Desta forma, pode-se dizer que o aprendizado semissupervisionado é um meio termo entre o aprendizado supervisionado e o não supervisionado. O aprendizado semissupervisionado tem sido explorado em diversas aplicações, desde reconhecimento de imagens [Li et al. 2024b] e processamento de linguagem natural [Knisely and Pavliscsak 2023] até bioinformática [Ma et al. 2024] e análise de redes sociais [Ninalga 2023].

Entre os diversos algoritmos de aprendizado semissupervisionados existentes na literatura, um bem conhecido é o Self-training [Yarowsky 1995], que envolve o uso iterativo de um modelo inicial treinado com um pequeno conjunto de dados rotulados para fazer previsões sobre o conjunto de dados não rotulados. As previsões mais confiáveis são então adicionadas ao conjunto de dados rotulados e o modelo é treinado novamente com esse conjunto expandido. Esse processo se repete até que um critério de parada seja alcançado [Chapelle et al. 2006]. Ao longo dos anos, diversos trabalhos foram desenvolvidos com o intuito de aprimorar o desempenho do Self-training [Gorgônio et al. 2019] [Vale et al. 2021] [Jan and Verma 2019] [Xinqin et al. 2019] [Parvin and Saleena 2020] [Wei et al. 2022], entre eles surgiu um algoritmo conhecido como FlexCon-CE [Medeiros et al. 2023], que é a base para a avaliação realizada neste trabalho cujos detalhes serão explicados a seguir.

2.1.1. Algoritmo FlexCon-CE

Conforme discutido anteriormente, o objetivo deste trabalho é avaliar o impacto do percentual de instâncias inicialmente rotuladas e de diferentes bases de dados no desempenho do FlexCon-CE. Este algoritmo emprega comitês de classificadores para selecionar e rotular instâncias em bases de dados utilizadas em aprendizado semissupervisionado. Portanto, é de fundamental importância compreender em detalhes o funcionamento desse algoritmo.

O Algoritmo 1 apresenta o fluxo do FlexCon-CE. Inicialmente, a base de dados é dividida em duas partes: dados rotulados (D_l) e dados não rotulados (D_u) e é definida uma lista de classificadores (F_n). Em seguida, um comitê de classificadores vazio é criado (linha 2) antes do início do loop, que repete o processo de rotulagem até que todas as instâncias da base de dados estejam rotuladas (linhas 3-13). O processo de rotulagem começa com o treinamento de cada classificador (f) na lista de classificadores (F_n), que são, em seguida, adicionados ao comitê (linhas 4-7). Posteriormente, o comitê de classificadores é utilizado para predizer os rótulos das instâncias não rotuladas (linha 8). Um novo limiar de confiança é então calculado (linha 9) e utilizado para selecionar as instâncias que serão adicionadas ao conjunto de dados rotulados, juntamente com seus rótulos preditos (linha 10). Finalmente, as instâncias selecionadas são removidas de D_u e incorporadas a D_l, continuando o processo iterativo até que D_u esteja vazio. Para uma descrição mais

detalhada do funcionamento do FlexCon-CE, recomenda-se consultar [Medeiros et al. 2023].

Algorithm 1: *Flexible Confidence with Ensemble* (FlexCon-CE)

Entrada : instâncias rotuladas D_l, instâncias não-rotuladas D_u, Pool de classificadores F_n

$D_l = (x_i, y_i) \mid i = 1, \ldots, l$ e $D_u = x_j \mid j = l+1, \ldots, l+u$;
Criar o comitê e vazio;
while $D_u \neq \emptyset$ **do**
  for $f \in F_n$ **do**
    Treinar o classificador f com o conjunto D_l utilizando aprendizado supervisionado;
    Adicionar f ao comitê e;
  end
  Predizer as instâncias em D_u utilizando o comitê e;
  Calcular o novo limiar de confiança;
  Selecionar e rotular cada instância de D_u para o conjunto S;
  Remover as instâncias do subconjunto $S = \{s_1, s_2, \ldots, s_n\}$ de D_u, cujo valor de confiança de $e(s_x)$ é maior ou igual ao limiar de confiança;
  Adicionar o subconjunto $\{(x, e(x)) \mid x \in S\}$ para o conjunto rotulado D_l;
end

Result: dados rotulados

2.2. Comitê de Classificadores

Os sistemas de classificação vem apresentando aumento da sua complexidade e consequentemente do número de suas aplicações, o que estimula a pesquisa por mais abordagens e metodologias nesta área. Contudo, um único classificador pode não ter bom desempenho em todas as atividades ou com todas as formas de dados. Diante do exposto, a possibilidade de combinar diferentes classificadores surge como uma possibilidade promissora de resolver esse problema, a esta abordagem é dado o nome de comitê de classificadores ou sistemas multiclassificadores [Wolpert and Macready 1997] [Nascimento et al. 2014].

Em atividades de classificação, um comitê consiste em vários submodelos conhecidos como classificadores base, que são geralmente obtidos a partir do treinamento de diferentes algoritmos de aprendizado de máquina. Exemplos comuns de algoritmos incluem: Árvores de Decisão, k-Vizinhos mais Próximos e Naive Bayes, entre outros. Neste artigo, foram utilizados os três primeiros para compor os comitês, explorando suas diferentes capacidades de generalização e complementaridade para melhorar a precisão do modelo final. Os comitês de classificadores podem ser construídos com o mesmo tipo de classificador base, resultando em comitês homogêneos, ou por classificadores base diferentes, o que leva à formação de comitês heterogêneos [Gharroudi 2017].

Gerado um grupo de classificadores base, segue-se para a escolha do método de combinação da saída do comitê. A literatura dispõe de diversas possibilidades de combinação, portanto, neste trabalho optou-se pela votação por ter sido o método utilizado na proposta original do FlexCon-CE. O método de votação por maioria é frequentemente

utilizado para combinar classificadores, sua combinação é feita através da votação dos resultados de cada classificador ao ser apresentada uma nova instância [Kuncheva 2014].

3. Trabalhos Relacionados

Ao longo dos anos, vem sendo realizadas diversas pesquisas na área de aprendizado de máquina utilizando o Self-training [Li et al. 2024a] [Ninalga 2023] [Xu and Li 2023] [Vale et al. 2021] e/ou comitês de classificadores [Wei et al. 2022] [Parvin and Saleena 2020] [Xinqin et al. 2019] [Jan and Verma 2019] em diversas áreas, tais como, saúde, finanças, redes sociais, entre outros. Conforme explicado anteriormente, o FlexCon-CE propõe uma mesclagem destas duas abordagens com o intuito de melhorar o desempenho de um algoritmo de aprendizado semissupervisionado, chamado FlexCon-C [Vale et al. 2021].

Entre os trabalhos que propõem mudanças no processo de rotulagem do Self-training, o mais semelhante ao desta pesquisa foi desenvolvido por [Vale et al. 2021] e utilizado como base para criação do FlexCon-CE. Este estudo propôs alterações nos algoritmos Self-training e Co-training para automatizar o processo de rotulagem de instâncias não rotuladas, de forma a flexibilizar o limiar de confiança para inclusão de novas instâncias no conjunto de dados rotulados. No trabalho de [Li et al. 2024a], foi criado um método semissupervisionado, baseado no Self-training, para identificação de falhas sísmicas. No artigo de [Ninalga 2023], foi apresentado um sistema de detecção de gravidade de depressão, embasado no Self-training, para prever se uma postagem em rede social é de um usuário que está apresentando níveis graves, moderados ou baixos de depressão. Já em [Xu and Li 2023], as tarefas de classificação, utilizando o Self-training, são exploradas usando a relação de imagem e texto de redes sociais para detecção de sarcasmo, classificação de sentimento e detecção de discurso de ódio.

Outros trabalhos apresentam propostas de algoritmos que combinam aprendizado semissupervisionado com comitês de classificadores. Em [Wei et al. 2022], foi elaborado um algoritmo de aprendizado semissupervisionado com comitê de classificadores projetado para prever falhas de software. Este algoritmo emprega uma combinação por votação entre os classificadores base para gerar a saída final do comitê. Já [Parvin and Saleena 2020], aplicaram múltiplos modelos de classificadores combinados para prever a pontuação de crédito de clientes em bancos e instituições financeiras. No estudo de [Xinqin et al. 2019], os pesquisadores investigaram dados de riscos nas inspeções de segurança ferroviária. Para antecipar situações de risco, foi proposto um modelo preditivo baseado em comitê de classificadores, combinando os resultados por votação para formar o modelo final.

4. Metodologia

Para reavaliar a eficiência e viabilidade do FlexCon-CE, será adotada uma abordagem de análise empírica. A seguir, serão detalhados os principais aspectos da metodologia experimental, incluindo as características das bases de dados utilizadas e o design dos experimentos. Neste estudo foram utilizadas 27 bases de dados, sendo 20 provenientes do trabalho de [Medeiros et al. 2023] e 7 adicionais, as quais estão disponíveis em repositórios e plataformas, tais como GitHub [Breiman 1996], UCI Machine Learning [Dheeru and Taniskidou 2017] e Kaggle datasets [Smith et al. 1988]. As características dessas bases de dados são apresentadas na Tabela 1: a primeira coluna tem o nome da base, a segunda

(#Inst) possui o número de instâncias, a terceira (#Att) mostra o número de atributos, a quarta (#CL) indica o número de classes, a quinta (TP) categoriza os tipos de dados ('I' = inteiro e 'R' = real). Por fim, a última coluna (BL) classifica as bases de dados quanto ao balanceamento, indicando se são balanceadas ('B') ou desbalanceadas ('U').

Tabela 1. Propriedades das Bases de Dados

Bases de Dados	#Inst	#Att	#CL	TP	BL
Blood Transfusion Service Center	748	5	2	R	U
Car	1728	22	4	I	U
Cnae-9	1080	857	9	I	B
Dermatology	366	131	6	I, R	U
German Credit	1000	62	2	I, R	U
Haberman	306	15	2	I, R	U
Hill Valley	1212	101	2	R	B
Image Segmentation	2310	20	7	I, R	B
King-Rook vs King-Pawn	3196	40	2	I	B
Madelon	2600	501	2	R	B
Mammographic Mass	961	6	2	R	B
Multiple Features Karhunen	2000	65	10	R	B
Mushroom	8124	113	2	I	B
Ozone Level Detection	2536	73	2	R	U
Pima	768	9	2	I, R	U
Planning Relax	182	13	2	R	U
Seeds	210	8	3	R	B
Semeion	1593	257	10	I	B
Solar Flare	1389	21	6	I	U
Spect Heart	349	45	2	R	U
Tic Tac Toe Endgame	958	28	2	I	U
Twonorm	7400	21	2	R	B
Vehicle	846	19	4	R	B
Waveform	5000	41	3	R	B
Wilt	4839	6	2	R	U
Wine	4898	12	7	R	U
Yeast	1484	9	10	R	U

Na implementação dos experimentos usou-se a biblioteca scikit-learn [Pölsterl 2020], disponível na linguagem Python. A escolha dessa biblioteca se justifica por sua ampla gama de métodos de aprendizado de máquina para tarefas como classificação, regressão e agrupamento. Além disso, o scikit-learn integra-se de forma eficiente com outras bibliotecas populares em Python, como Matplotlib [Bisong et al. 2019], Numpy [Harris et al. 2020] e Pandas [Nelli 2018], o que facilita a execução dos experimentos de aprendizado de máquina. Para avaliar o desempenho do FlexCon-CE os percentuais de instâncias inicialmente rotuladas utilizados nesta pesquisa foram valores médios em relação aos de [Medeiros et al. 2023], que usou múltiplos de 5, a saber: 3%, 8%, 13%, 18% e 23%. Para facilitar a comparação entre os resultados destes experimentos com o de [Medeiros et al. 2023], adotou-se os mesmos classificadores: Naive Bayes (NB), Árvore

de Decisão (DT) e k-Vizinhos mais Próximos (k-NN).

Seguindo o mesmo procedimento de [Medeiros et al. 2023], os classificadores treinados foram incorporados a dois tipos de comitês de classificadores: homogêneo - FlexCon-CE (Hom) e heterogêneo - FlexCon-CE (Het). As saídas dos classificadores base do comitê foram combinadas através de votação para determinar o rótulo de cada instância. Este método envolve uma combinação das saídas dos classificadores, em que o processo consiste em identificar a classe vencedora com base no número total de votos recebidos por cada classificador [Xinqin et al. 2019] [Wei et al. 2022] [Kuncheva 2014]. Por fim, o FlexCon-CE terá seu desempenho avaliado considerando duas métricas: acurácia e F1-Score.

5. Resultados

Esta seção apresenta os resultados obtidos ao avaliar o desempenho do FlexCon-CE em 27 bases de dados, variando 10 percentuais de instâncias inicialmente rotuladas. A seguir, é realizada uma análise empírica dos resultados com base nas métricas de acurácia e F1-score, além de uma análise estatística detalhada.

5.1. Análise de Performance

A Tabela 2 apresenta os resultados obtidos a partir da aplicação de quatro comitês de classificadores distintos — três homogêneos e um heterogêneo — em 27 conjuntos de dados. Para facilitar a interpretação e otimizar o uso do espaço, foi calculada a média aritmética das métricas de acurácia e F1-score, além do desvio padrão, considerando todas as bases de dados. A organização da tabela é a seguinte: a primeira coluna indica o percentual de instâncias inicialmente rotuladas, enquanto as colunas subsequentes exibem as médias das métricas de acurácia e F1-score, acompanhadas do desvio padrão, conforme cada comitê de classificadores.

Os comitês homogêneos, identificados como FlexCon-CE (Hom), têm seus nomes simplificados na tabela pelas siglas de seus classificadores base: NB para Naive Bayes, DT para Árvore de Decisão e KNN para k-Vizinhos mais Próximos. O comitê heterogêneo, FlexCon-CE (Het), é representado pela sigla HET. Adicionalmente, os melhores resultados estão destacados em negrito para facilitar a visualização.

Tabela 2. Acurácia e F1-score médios de cada comitê

	Acurácia				F-Score			
	NB	**DT**	**KNN**	**HET**	**NB**	**DT**	**KNN**	**HET**
3%	63,35 ± 8,37	**68,84 ± 7,04**	63,19 ± 7,19	68,70 ± 6,81	52,52 ± 7,93	**59,94 ± 7,57**	47,20 ± 7,84	56,42 ± 8,40
5%	66,08 ± 8,14	72,40 ± 5,65	68,52 ± 6,52	**73,35 ± 5,77**	57,34 ± 8,17	**63,93 ± 7,03**	53,74 ± 7,61	62,48 ± 7,41
8%	66,85 ± 7,54	74,00 ± 5,67	71,71 ± 5,97	**75,77 ± 5,30**	59,39 ± 8,46	**65,93 ± 7,40**	57,72 ± 7,39	65,00 ± 7,18
10%	68,17 ± 7,19	74,89 ± 5,94	73,15 ± 5,23	**76,52 ± 5,23**	61,24 ± 7,61	**67,78 ± 7,22**	59,97 ± 6,61	66,53 ± 6,38
13%	68,19 ± 7,64	76,10 ± 5,15	74,69 ± 4,79	**76,99 ± 5,19**	61,98 ± 7,82	**68,79 ± 6,45**	61,55 ± 6,08	67,81 ± 6,91
15%	67,76 ± 7,49	75,77 ± 5,54	74,84 ± 4,86	**77,37 ± 5,02**	62,02 ± 7,80	68,40 ± 6,35	62,19 ± 6,32	**68,52 ± 6,58**
18%	68,71 ± 7,15	77,19 ± 5,48	75,80 ± 4,66	**78,45 ± 5,02**	62,99 ± 7,54	**70,47 ± 6,27**	64,00 ± 6,09	69,94 ± 6,35
20%	68,76 ± 7,42	77,35 ± 5,45	76,14 ± 4,32	**78,60 ± 5,10**	62,88 ± 7,83	**70,61 ± 6,26**	64,55 ± 5,57	69,78 ± 6,72
23%	68,52 ± 7,40	77,72 ± 5,11	76,45 ± 4,79	**78,35 ± 4,55**	63,25 ± 7,61	**70,10 ± 5,76**	65,17 ± 5,81	70,01 ± 5,74
25%	68,93 ± 7,16	78,16 ± 5,19	77,26 ± 4,64	**78,76 ± 4,99**	63,02 ± 7,54	**71,54 ± 6,22**	65,94 ± 5,68	71,05 ± 6,55

Ao analisar a acurácia na Tabela 2, é possível perceber que o FlexCon-CE usando o comitê heterogêneo - HET - obteve os melhores resultados em 90% dos casos analisados (9 dos 10 percentuais de instâncias inicialmente rotuladas). Na mesma tabela, considerando

as acurácias obtidas pelo comitê de classificadores DT, pode ser notado que o seu desempenho é ligeiramente superior ao do comitê heterogêneo quando usa 3% de instâncias inicialmente rotuladas. Com base nestes resultados, é possível afirmar que o comitê HET obteve desempenho significativamente melhor em relação aos comitês de classificadores homogêneos quando considerada a acurácia.

Ainda observando a Tabela 2, percebe-se que o padrão de resultados é semelhante ao que foi observado por [Medeiros et al. 2023], quanto maior o percentual de instâncias inicialmente rotuladas, melhor o desempenho, independente do tipo de comitê. Entretanto, os comitês NB e DT possuem um desempenho melhor usando 13% do que 15% de instâncias inicialmente rotuladas.

Quando o foco da análise é o desempenho em relação ao F1-score percebe-se, a partir da mesma tabela, que o comitê homogêneo DT possui melhor desempenho em comparação com os outros comitês, levando vantagem em 90% dos casos analisados (9 dos 10 percentuais de instâncias inicialmente rotuladas). Entretanto, de maneira análoga aos resultados obtidos na análise anterior, o classificador heterogêneo não apresenta resultados tão distantes em comparação com o comitê DT e se desempenha melhor quando é usado 15% de instâncias inicialmente rotuladas. Diante do exposto, é possível afirmar que o DT obteve desempenho significativamente melhore em relação aos demais comitês de classificadores analisados quando é considerado o F1-score.

Analisando as duas métricas em conjunto, percebe-se que existe um padrão entre os resultados de desempenho da acurácia e F1-score: os valores da métrica F1-score correspondem a aproximadamente 10 pontos percentuais a menos do que a acurácia. Este resultado pode ser explicado pelo fato de que a métrica acurácia apresenta melhores resultados diante de bases de dados desbalanceadas, já que a acurácia utiliza em seu cálculo apenas a diagonal principal da matriz de confusão. Tendo em vista que aproximadamente 50% das bases de dados utilizadas neste trabalho são desbalanceadas, a acurácia apresenta melhores resultados quando comparado ao F1-score, já que para o cálculo desta métrica as classes minoritárias acabam exercendo mais influência no resultado.

Em resumo, é possível afirmar que, independente do percentual de instâncias inicialmente rotuladas, o comitê heterogêneo - HET - e o homogêneo com Árvores de Decisão - DT - tiveram os melhores resultados de acurácia e F1-score, respectivamente. A próxima subseção apresenta a análise estatística dos resultados explanados nesta seção para demonstrar que o FlexCon-CE apresenta resultados promissores.

5.2. Análise Estatística

Para analisar estatisticamente os resultados dos experimentos realizados nesta pesquisa, foi aplicado o teste *post-hoc* de Friedman, cujos resultados são apresentados por meio dos diagramas de diferença crítica (CD) exibidos nas Figuras 1 e 2. Este tipo de teste analisa os resultados baseado em seus rankings, portanto nestes diagramas, os comitês posicionados mais à esquerda exibem melhores resultados, enquanto os comitês mais à direita apresentaram os piores desempenhos. Por fim, um comitê é considerado estatisticamente diferente de outro quando ambos não estão cobertos pela linha de diferença crítica (linha horizontal em negrito no diagrama). Caso contrário, quando esta linha cobre duas ou mais abordagens, significa que a hipótese nula do teste de Friedman não pode ser rejeitada.

A Figura1 apresenta o diagrama de diferença crítica gerado a partir da acurácia dos

comitês de classificadores. Ao analisar o diagrama mencionado observa-se que o comitê heterogêneo - HET - alcançou o maior ranking, confirmando a análise da acurácia realizada na seção anterior. Além disso, este comitê é estatisticamente superior aos demais, pois não apresenta nenhuma semelhança estatística.

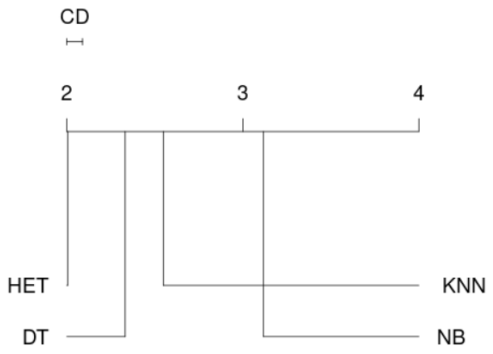

Figura 1. Diagramas de diferença crítica - Acurácia

A Figura 2 ilustra o diagrama de diferença crítica derivado do desempenho dos comitês considerando a métrica F1-score. Ao analisar os rankings de cada comitê, percebe-se que o HET obteve o maior ranking, mas é estatisticamente semelhante ao comitê homogêneo DT. Sendo assim, ambos são superiores aos demais em relação a métrica F1-score. Esta conclusão atesta o que foi relatado na seção anterior, onde explicou-se que, quando o foco é o desempenho de F1-score, o comitê HET e o DT são superiores aos demais, o que é comprovado também de maneira estatística.

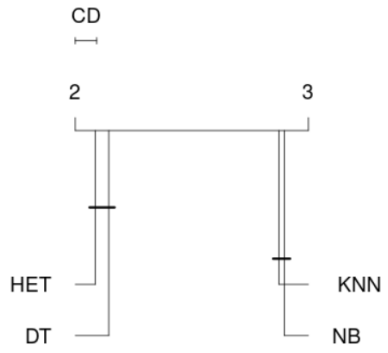

Figura 2. Diagramas de diferença crítica - F1-Score

Em suma, é possível afirmar que os resultados da análise estatística corroboram com os da seção anterior, demonstrando que mesmo com o aumento do número de bases de dados e de percentuais inicialmente rotulados e a utilização de mais de uma métrica de avaliação de desempenho, o FlexCon-CE manteve seu bom desempenho aferido em [Medeiros et al. 2023], provando ser um algoritmo semissupervisionado promissor. Por fim, conclui-se que o FlexCon-CE usando comitê heterogêneo (HET) é de fato o mais

eficiente para problemas de aprendizado semissupervisionado, conforme proposto em seus primeiros experimentos.

6. Conclusão e Trabalhos Futuros

Este artigo expande significativamente a análise do algoritmo de aprendizado semissupervisionado FlexCon-CE, introduzindo uma abordagem metodológica robusta e abrangente. Ao aumentar o número de bases de dados de 20 para 27, incorporar uma métrica adicional de avaliação de desempenho e utilizar 10 diferentes percentuais de instâncias inicialmente rotuladas, foi possível elevar substancialmente a quantidade de experimentos realizados. Essa expansão metodológica não apenas fortalece a validação da eficácia do FlexCon-CE, mas também oferece uma compreensão mais profunda e detalhada de seu desempenho em diferentes cenários.

A análise comparativa entre quatro comitês de classificadores, sendo três homogêneos e um heterogêneo, revelou que o comitê heterogêneo (HET) se destacou em termos de acurácia e F1-score, superando estatisticamente os demais. Este resultado não só confirma as descobertas anteriores de [Medeiros et al. 2023], mas também solidifica a posição do FlexCon-CE como uma ferramenta poderosa em contextos semissupervisionados. A contribuição deste trabalho é significativa, pois amplia o conhecimento sobre o comportamento do FlexCon-CE em cenários variados, proporcionando uma base sólida para futuras pesquisas. A importância deste estudo reside na validação estatística e na robustez metodológica aplicada, que juntas garantem a relevância dos resultados apresentados para a comunidade científica.

Para futuras investigações, recomenda-se a exploração de outros algoritmos de classificação, incluindo redes neurais, para avaliar o desempenho do FlexCon-CE em novos contextos. Além disso, a análise do custo computacional das diferentes configurações de comitês e classificadores individuais poderá fornecer insights valiosos para a otimização de desempenho e eficiência. A flexibilização do tamanho dos conjuntos de classificadores também é uma área promissora para estudo, potencializando ainda mais o impacto e aplicabilidade do FlexCon-CE.

Referências

Bisong, E. et al. (2019). *Building machine learning and deep learning models on Google cloud platform*. Springer.

Breiman, L. (1996). Bias, variance, and arcing classifiers. Technical report, Tech. Rep. 460, Statistics Department, University of California, Berkeley ....

Chapelle, O., Scholkopf, B., and Zien, A. (2006). *Semi-Supervised Learning*. The MIT Press.

Chapelle, O., Scholkopf, B., and Zien, A. (2009). Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542.

Dheeru, D. and Taniskidou, E. K. (2017). Uci machine learning repository.

Gharroudi, O. (2017). *Ensemble multi-label learning in supervised and semi-supervised settings*. PhD thesis, Université de Lyon.

Gorgônio, A. C., Alves, C. T., Lucena, A. J., Gorgônio, F. L., Vale, K. M., and Canuto, A. M. (2019). Análise da variação do limiar para o algoritmo de aprendizado semissupervisionado flexcon-c. *Brazilian Journal of Development*, 5(11):26654–26669.

Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., et al. (2020). Array programming with numpy. *Nature*, 585(7825):357–362.

Jan, M. Z. and Verma, B. (2019). A novel diversity measure and classifier selection approach for generating ensemble classifiers. *Ieee Access*, 7:156360–156373.

Knisely, B. M. and Pavliscsak, H. H. (2023). Research proposal content extraction using natural language processing and semi-supervised clustering: A demonstration and comparative analysis. *Scientometrics*.

Kuncheva, L. I. (2014). *Combining Pattern Classifiers: Methods and Algorithms*. John Wiley & Sons, Hoboken, NJ, 2nd edition.

Li, K., Li, X., Yin, R., and Wang, L. (2024a). A method for seismic fault identification based on self-training with high-stability pseudo-labels. *Applied Soft Computing*, page 111894.

Li, S., Kou, P., Ma, M., Yang, H., Huang, S., and Yang, Z. (2024b). Application of semi-supervised learning in image classification: Research on fusion of labeled and unlabeled data. *IEEE Access*, 12:27331–27343.

Ma, H., Jiang, F., Rong, Y., Guo, Y., and Huang, J. (2024). Toward robust self-training paradigm for molecular prediction tasks. *Journal of Computational Biology*, 31(3):213–228.

Medeiros, A., Gorgônio, A. C., Vale, K. M. O., Gorgônio, F. L., and Canuto, A. M. d. P. (2023). Flexcon-ce: A semi-supervised method with an ensemble-based adaptive confidence. In *Brazilian Conference on Intelligent Systems*, pages 95–109. Springer.

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

Nascimento, D. S., Canuto, A. M., and Coelho, A. L. (2014). An empirical analysis of meta-learning for the automatic choice of architecture and components in ensemble systems. In *2014 Brazilian Conference on Intelligent Systems*, pages 1–6. IEEE.

Nelli, F. (2018). *Python data analytics with Pandas, NumPy, and Matplotlib*. Springer.

Ninalga, D. (2023). Cordyceps@lt-edi: Depression detection with reddit and self-training.

Parvin, A. S. and Saleena, B. (2020). An ensemble classifier model to predict credit scoring-comparative analysis. In *2020 IEEE international symposium on smart electronic systems (iSES)(Formerly iNiS)*, pages 27–30. IEEE.

Pölsterl, S. (2020). scikit-survival: A library for time-to-event analysis built on top of scikit-learn. *Journal of Machine Learning Research*, 21(212):1–6.

Rodrigues, F. M., Câmara, C. J., Canuto, A. M., and Santos, A. M. (2014). Confidence factor and feature selection for semi-supervised multi-label classification methods. In *2014 International Joint Conference on Neural Networks (IJCNN)*, pages 864–871. IEEE.

Sanches, M. K. (2003). *Aprendizado de máquina semi-supervisionado: proposta de um algoritmo para rotular exemplos a partir de poucos exemplos rotulados*. PhD thesis, Universidade de São Paulo.

Smith, J. W., Everhart, J. E., Dickson, W., Knowler, W. C., and Johannes, R. S. (1988). Using the adap learning algorithm to forecast the onset of diabetes mellitus. In *Proceedings of the annual symposium on computer application in medical care*, page 261. American Medical Informatics Association.

Vale, K. M. O., Canuto, A. M. d. P., de Medeiros Santos, A., e Gorgônio, F. d. L., Tavares, A. d. M., Gorgnio, A. C., and Alves, C. T. (2018). Automatic adjustment of confidence values in self-training semi-supervised method. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Vale, K. M. O., Gorgônio, A. C., Flavius Da Luz, E. G., and Canuto, A. M. D. P. (2021). An efficient approach to select instances in self-training and co-training semi-supervised methods. *IEEE Access*, 10:7254–7276.

Wang, M., Fu, W., Hao, S., Tao, D., and Wu, X. (2016). Scalable semi-supervised learning by efficient anchor graph regularization. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1864–1877.

Wei, W., Jiang, F., Yu, X., and Du, J. (2022). An ensemble learning algorithm based on resampling and hybrid feature selection, with an application to software defect prediction. In *2022 7th International Conference on Information and Network Technologies (ICINT)*, pages 52–56. IEEE.

Wolpert, D. H. and Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82.

Xinqin, L., Tianyun, S., Ping, L., and Wen, Z. (2019). Application of bagging ensemble classifier based on genetic algorithm in the text classification of railway fault hazards. In *2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD)*, pages 286–290. IEEE.

Xu, C. and Li, J. (2023). Borrowing human senses: Comment-aware self-training for social media multimodal classification.

Yarowsky, D. (1995). Unsupervised word sense disambiguation rivaling supervised methods. In *33rd annual meeting of the association for computational linguistics*, pages 189–196.