Particle Competition for Pseudo-Labeling in Positive and Unlabeled Learning

Hemilyn S. A. Porfirio[1], **Guilherme Henrique Messias**[1], **Alan Demétrius Baria Valejo**[1]

[1]Departamento de Computação – Universidade Federal de São Carlos (UFSCar)
São Carlos – SP – Brazil

{hemilyn, ghmessias}@estudante.ufscar.br; alanvalejo@ufscar.br

Abstract. *Supervised algorithms require labeled examples from all classes, but obtaining negative labels can be costly or infeasible. This setting characterizes Positive and Unlabeled Learning (PUL), in which only positive examples and unlabeled data are available. This paper introduces PC-PUL, which adapts Particle Competition to select reliable negatives by exploiting data topology. Autonomous particles compete for node ownership and form territories, from which nodes in negative territories that are topologically distant from positive ones are selected. Experiments on five datasets show superior results on CiteSeer and competitive performance on PubMed and MNIST, indicating that PC-PUL is a promising alternative for reliable-negative selection in PUL.*

1. Introdução

A escassez de dados rotulados limita o treinamento de modelos em domínios nos quais a anotação exige especialistas, dispositivos ou experimentos custosos [Zhu and Goldberg 2009]. Em aplicações como detecção de notícias falsas [Souza et al. 2021], classificação de textos [Li and Liu 2003] e identificação de genes associados a doenças [Yang et al. 2012], frequentemente estão disponíveis apenas um pequeno conjunto de exemplos positivos P e um conjunto não rotulado U, contendo exemplos positivos e negativos. Essa configuração caracteriza o Aprendizado Positivo e Não Rotulado (*Positive and Unlabeled Learning* – PUL) [Bekker and Davis 2020, Jaskie and Spanias 2019].

Uma estratégia difundida em PUL é a abordagem de duas etapas [Bekker and Davis 2020]: primeiro, seleciona-se em U um conjunto de negativos confiáveis (*reliable negatives* – RN); depois, treina-se um classificador binário com P e RN. O desempenho final depende da qualidade dos RN selecionados, visto que, se este conjunto estiver contaminado com exemplos positivos, as fronteiras de decisão do classificador serão comprometidas [Jaskie and Spanias 2019, Bekker and Davis 2020]. Além disso, [Messias et al. 2025] observa que os trabalhos da literatura avaliam apenas o resultado conjunto das duas fases, de modo que deficiências na pseudo-rotulagem permanecem ocultas pelo classificador final.

Muitos métodos de seleção de RN operam exclusivamente sobre atributos, desconsiderando as relações existentes entre os exemplos. Abordagens baseadas em grafos procuram superar essa limitação ao explorar a topologia dos dados. O LP-PUL [Carnevali et al. 2021] emprega distâncias de menores caminhos calculadas por Dijkstra [Dijkstra 1959], enquanto o PU-LP [Ma and Zhang 2017] utiliza similaridades baseadas

no índice de Katz. Embora considerem a topologia, esses métodos atribuem escores diretamente aos vértices, sem organizar previamente o grafo em regiões topologicamente coesas (comunidades).

A competição de partículas [Quiles et al. 2008, Breve et al. 2012] oferece uma alternativa ao formar comunidades por meio de objetos que exploram o grafo e disputam a posse dos vértices. Nesse modelo, partículas percorrem o grafo, disputam vértices e consolidam territórios. O potencial de cada partícula aumenta ao visitar seu próprio território e diminui ao invadir regiões rivais; de modo análogo, a posse dos vértices é reforçada pelo proprietário e enfraquecida por concorrentes. Essa dinâmica conduz as partículas a regiões topologicamente coesas.

Neste artigo, propomos o *Particle Competition for Positive and Unlabeled Learning* (PC-PUL). O método forma territórios por competição de partículas, classifica-os conforme a presença de positivos conhecidos e seleciona como RN os vértices de territórios negativos com maior dissimilaridade topológica em relação aos positivos. Até onde sabemos, este é o primeiro trabalho a aplicar competição de partículas especificamente à seleção de negativos confiáveis em PUL.

As contribuições são: (i) um método que combina detecção de comunidades e dissimilaridade topológica para selecionar RN; (ii) uma avaliação separada da pseudo-rotulagem e de seu impacto no classificador final; e (iii) experimentos em cinco conjuntos de dados, comparando o PC-PUL com métodos baseados em atributos, agrupamento, menores caminhos e propagação de rótulos.

Os resultados mostram que o PC-PUL alcança desempenho superior ou competitivo na seleção de negativos confiáveis em diferentes configurações, com ganhos que também se refletem na classificação final em parte dos cenários avaliados. Em conjunto, os resultados sustentam a competição de partículas como uma alternativa promissora, mas não universal, para pseudo-rotulagem em PUL.

2. Trabalhos Relacionados

2.1. Competição de Partículas

A competição de partículas foi proposta por [Quiles et al. 2008] para detecção de comunidades em redes complexas. A movimentação combina exploração e consolidação. Com probabilidade p_{det}, a partícula realiza um movimento determinístico em seu território; com probabilidade $1 - p_{det}$, explora um vértice vizinho. Esse equilíbrio permite detectar comunidades por interações locais, sem otimizar diretamente uma função global de particionamento [Quiles et al. 2008]. Extensões do modelo apresentaram robustez a rótulos incorretos e desempenho competitivo com poucos exemplos rotulados [Breve et al. 2010, Breve et al. 2012].

Essas características motivam sua aplicação em PUL, pois permitem organizar o grafo antes da pseudo-rotulagem. O PC-PUL utiliza os territórios formados para restringir e ordenar os candidatos a negativos confiáveis, considerando sua relação topológica com os positivos conhecidos.

2.2. Seleção de Negativos Confiáveis em PUL

Na abordagem PUL de duas etapas, um conjunto de negativos confiáveis RN é inicialmente extraído dos dados não rotulados e, posteriormente, utilizado

com os positivos conhecidos para treinar um classificador [Jaskie and Spanias 2019, Bekker and Davis 2020]. Os métodos diferem principalmente na representação dos dados e no critério adotado para selecionar RN.

Entre os métodos vetoriais, o Rocchio-SVM [Li and Liu 2003] constrói protótipos para os conjuntos positivo e não rotulado e seleciona negativos por similaridade de cosseno. O MCLS [Chaudhari and Shevade 2012] combina uma inicialização baseada em k-means com atualizações sucessivas de um LS-SVM, impondo uma proporção estimada de positivos. O C-CRNE [Liu and Peng 2014], por sua vez, agrupa os exemplos positivos, calcula regiões de influência ao redor de seus centroides e considera negativos os exemplos não rotulados externos a essas regiões. Embora explorem diferentes critérios de separação, esses métodos utilizam apenas os atributos e não incorporam relações explícitas entre os exemplos.

Métodos baseados em grafos exploram a posição topológica dos vértices. O LP-PUL [Carnevali et al. 2021] calcula, por meio de menores caminhos, a distância média de cada vértice não rotulado aos positivos conhecidos e seleciona aqueles com maiores distâncias. O PU-LP [Ma and Zhang 2017] utiliza o índice de Katz em um processo bidirecional: primeiro expande o conjunto positivo e, em seguida, seleciona como negativos os vértices com menor similaridade em relação aos positivos expandidos. Esses métodos demonstram a utilidade da topologia, mas realizam a seleção principalmente por escores individuais de distância ou similaridade.

O PC-PUL distingue-se dessas abordagens ao introduzir uma etapa anterior de formação de comunidades. A competição de partículas organiza o grafo em regiões topologicamente coesas, que são classificadas conforme a presença de positivos conhecidos. Apenas os vértices pertencentes a comunidades negativas são posteriormente ordenados por sua dissimilaridade em relação as comunidades positivas. Assim, o método combina uma decisão em nível global, baseado em comunidades, com uma decisão em nível individual, baseada na distância topológica.

A Tabela 1 sintetiza a representação dos dados e a principal técnica empregada por cada método para selecionar negativos confiáveis.

Tabela 1. Comparação entre os métodos de seleção de negativos confiáveis.

Método	**Representação**	**Técnica de seleção de RN**
Rocchio-SVM [Li and Liu 2003]	Vetores	Protótipos e similaridade cosseno
MCLS [Chaudhari and Shevade 2012]	Vetores	k-means e LS-SVM
C-CRNE [Liu and Peng 2014]	Vetores	Agrupamento e centroides
LP-PUL [Carnevali et al. 2021]	Grafo	Menores caminhos
PU-LP [Ma and Zhang 2017]	Grafo	Propagação e índice de Katz
PC-PUL (Proposta)	Grafo	Competição de partículas e comunidades

3. Pseudo-Rotulagem via Competição de Partículas

O método proposto, denominado *Particle Competition for Positive and Unlabeled Learning* (PC-PUL), seleciona negativos confiáveis (RN) explorando a estrutura topológica dos dados. O método corresponde à primeira etapa da abordagem PUL de duas etapas:

partículas autônomas competem pela posse dos vértices, formando comunidades a partir das quais os candidatos a negativos são identificados e ordenados por dissimilaridade topológica.

3.1. Construção do Grafo

Para dados com estrutura relacional inerente, como redes de citação e redes sociais, utiliza-se o grafo original. O grafo é restrito ao maior componente conectado, assegurando a existência de caminhos entre os vértices considerados no cálculo das distâncias topológicas. Para dados representados apenas por vetores de atributos, constrói-se um grafo de k vizinhos mais próximos (k-NN), conforme a configuração descrita na Seção 4.1.

3.2. Competição de Partículas para Pseudo-rotulagem

O PC-PUL recebe como entrada um grafo $G = (V, E)$ e um conjunto de positivos conhecidos $P \subset V$. Cada vértice $v_i \in V$ possui um rótulo observado $y_i^{obs} \in \{0, 1\}$, tal que $y_i^{obs} = 1$ se $v_i \in P$ e $y_i^{obs} = 0$ se v_i pertence ao conjunto não rotulado $U = V \setminus P$. Os principais parâmetros são o número de partículas n_p, a probabilidade de movimento determinístico p_{det}, os fatores de atualização δ_p e δ_v, o limiar de potencial médio θ_{avg} e o número de iterações de paciência utilizado no critério de estabilização.

3.2.1. Inicialização

Inicialmente, cada vértice v_i recebe potencial $V_i = 0{,}05$ e nenhum proprietário, isto é, $owner_i = \emptyset$. As n_p partículas são posicionadas segundo uma de duas estratégias: aleatória uniforme ou ponderada pelo grau dos vértices. Na segunda estratégia, vértices com maior grau possuem maior probabilidade de receber uma partícula. Em ambos os casos, uma partícula é posicionada inicialmente em um vértice positivo $v \in P$, assegurando que a competição se inicie com ao menos uma partícula associada aos positivos conhecidos.

3.2.2. Dinâmica de Movimentação

A cada iteração, cada partícula p_j seleciona seu próximo vértice combinando dois comportamentos complementares: exploração da vizinhança e consolidação da comunidade sob sua posse. Com probabilidade p_{det}, a partícula realiza um movimento determinístico para um vértice vizinho pertencente à sua comunidade, reforçando sua ocupação. Com probabilidade complementar $1 - p_{det}$, realiza um movimento exploratório para um vértice vizinho, escolhido uniformemente ou com probabilidade proporcional ao seu grau. Para reduzir oscilações entre dois vértices consecutivos, o vértice visitado na iteração anterior é excluído do conjunto de candidatos, exceto quando é o único vizinho disponível.

3.2.3. Atualização de Potenciais

Após o movimento da partícula p_j para o vértice v_i, os potenciais e a posse são atualizados de acordo com três situações. Se v_i não possui proprietário, a partícula assume sua posse, isto é, $owner_i \leftarrow j$, e o potencial do vértice recebe o potencial atual da partícula, $V_i \leftarrow P_j$. Se v_i já pertence à partícula p_j, o potencial da partícula é reforçado conforme:

$$P_j^{(t+1)} = \min\left(1, P_j^{(t)} + \left(1 - P_j^{(t)}\right)\delta_p\right). \tag{1}$$

Se v_i pertence a outra partícula, ocorre um conflito. Nesse caso, os potenciais da partícula invasora e do vértice são reduzidos:

$$P_j^{(t+1)} = P_j^{(t)} - \left(P_j^{(t)} - P_{\min}\right)\delta_p. \tag{2}$$

$$V_i^{(t+1)} = V_i^{(t)}\left(1 - \delta_v\right), \tag{3}$$

em que $P_{\min} = V_{\min} = 0{,}05$ representa o potencial mínimo do sistema. Quando P_j atinge esse limite, a partícula é reinicializada em um vértice sem proprietário. Analogamente, quando V_i atinge o potencial mínimo, o vértice perde sua posse, $owner_i \leftarrow \emptyset$, e torna-se novamente disponível para ocupação.

3.2.4. Critérios de Convergência

O potencial médio dos vértices é definido por

$$\bar{V}^{(t)} = \frac{1}{|V|}\sum_{i=1}^{|V|} V_i^{(t)}. \tag{4}$$

Valores elevados de $\bar{V}^{(t)}$ indicam maior consolidação das comunidades formadas pelas partículas. O algoritmo termina quando uma das seguintes condições é satisfeita: (i) $\bar{V}^{(t)} \geq \theta_{avg}$; ou (ii) o desvio-padrão de $\bar{V}$ nas últimas *patience* iterações é inferior a $\tau = 0{,}001$, indicando estabilização da dinâmica competitiva.

3.2.5. Pseudo-rotulagem

Ao término da competição, os vértices com proprietário são agrupados de acordo com a partícula que os domina, formando comunidades $\mathcal{C} = \{C_1, \ldots, C_{n_p}\}$. Para cada comunidade C_j, calcula-se a proporção de positivos conhecidos:

$$r(C_j) = \frac{|C_j \cap P|}{|C_j|}. \tag{5}$$

Uma comunidade é classificada como positiva quando $r(C_j) \geq r_p$ (limiar mínimo de proporção de positivos); as demais são classificadas como negativas. Caso nenhuma comunidade satisfaça esse critério, aquela com maior proporção de positivos é definida como positiva. Os vértices sem proprietário permanecem no conjunto não rotulado e não são considerados candidatos a negativos confiáveis.

Para cada vértice v_n pertencente a uma comunidade negativa, calcula-se o escore de dissimilaridade em relação ao conjunto V^+ de vértices pertencentes às comunidades

positivas:

$$d(v_n) = \frac{1}{|V^+|} \sum_{v_p \in V^+} \mathrm{dist}(v_n, v_p), \tag{6}$$

em que $\mathrm{dist}(v_n, v_p)$ representa a distância de menor caminho entre v_n e v_p. Por fim, os n_{RN} vértices com maiores escores de dissimilaridade são selecionados como negativos confiáveis, formando o conjunto RN.

4. Análise Experimental

A avaliação experimental investiga a sensibilidade do PC-PUL aos seus hiperparâmetros e compara sua capacidade de selecionar negativos confiáveis com métodos da literatura. As adaptações em relação ao modelo original incluem um critério adicional de convergência por estabilização, estratégias de inicialização e movimentação ponderadas pelo grau, a alocação de uma partícula em um vértice positivo e a etapa final de pseudo-rotulagem.

A qualidade da seleção de RN é avaliada diretamente e também por seu impacto na segunda etapa de PUL. Para esta última, todos os métodos utilizam o mesmo classificador SVM com kernel RBF, treinado com os positivos conhecidos P e os negativos selecionados RN. Essa padronização reduz a influência da escolha do classificador e permite uma comparação mais controlada entre os métodos.

4.1. Configuração Experimental

Os experimentos utilizaram quatro grafos relacionais: Cora e CiteSeer [Sen et al. 2008], PubMed [Namata et al. 2012] e Twitch [Rozemberczki et al. 2021]; e o conjunto vetorial MNIST [LeCun et al. 1998]. Para o MNIST, foram amostradas 3.000 instâncias e construído um grafo k-NN com $k = 3$. Em todas as bases, utilizou-se o maior componente conectado, obtido com a biblioteca NetworkX [Hagberg et al. 2008]. A Tabela 2 apresenta suas principais características.

Tabela 2. Características dos conjuntos de dados.

Base	Vértices	Arestas	Atributos	Positivos (%)
Cora	2.485	5.069	1.433	29,22
CiteSeer	2.120	3.679	3.703	25,09
PubMed	19.717	44.324	500	39,94
Twitch	2.442	42.950	4	51,68
MNIST	3.000	6.961	784	49,17

Para obter tarefas binárias, definiu-se como positiva a classe 3 no Cora, a classe 2 no CiteSeer e no PubMed, a classe 1 no Twitch e os dígitos pares no MNIST. Essas são as classes com maior frequência de elementos nos conjuntos de dados, como reproduzido em [Wu et al. 2021], as demais classes foram consideradas negativas. Em cada execução, uma fração p dos exemplos originalmente positivos foi selecionada como conjunto positivo conhecido P, com $p \in \{0{,}01, 0{,}05, 0{,}10, 0{,}15, 0{,}20, 0{,}25\}$. Os demais exemplos, independentemente da classe real, foram tratados como não rotulados.

Foram selecionados $n_{RN} = 200$ negativos confiáveis em Cora, CiteSeer, PubMed e Twitch, e $n_{RN} = 300$ no MNIST. Para cada configuração, realizaram-se 3 execuções

com diferentes amostras de positivos, reportando-se a média e o desvio-padrão do F1-score macro. A análise de sensibilidade considerou os hiperparâmetros apresentados na Tabela 3.

Tabela 3. Hiperparâmetros avaliados do PC-PUL.

Parâmetro	Valores	Descrição
init	*random*, *degree*	Inicialização
movement	*uniform*, *degree*	Exploração
n_p	2, 5, 10, 20, 50, 100	Partículas
θ_{avg}	0,2 a 0,9	Convergência
strategy	*majority*, *percentage*	Comunidades positivas
r_p	0,1 a 0,5	Proporção positiva

Na comparação final, o PC-PUL utilizou $n_p = 100$, inicialização aleatória uniforme, exploração uniforme e $\theta_{avg} = 0{,}8$. Os baselines foram configurados conforme seus trabalhos originais: MCLS com $k = 7$ e $ratio = 0{,}1$; Rocchio-SVM com $\alpha = 0{,}7$ e $\beta = 0{,}3$; C-CRNE com $ratio = 0{,}1$; PU-LP com $\alpha = 0{,}1$, $m = 3$ e $l = 1$; e LP-PUL sem parâmetros adicionais.

4.2. Análise Paramétrica

Nesta seção, investiga-se a sensibilidade do PC-PUL às variações de hiperparâmetros. Uma vez que o modelo de competição de partículas foi originalmente proposto para detecção de comunidades, a análise busca identificar quais configurações são mais adequadas ao cenário de pseudo-rotulagem em PUL.

4.2.1. Impacto do Número de Partículas e das Estratégias

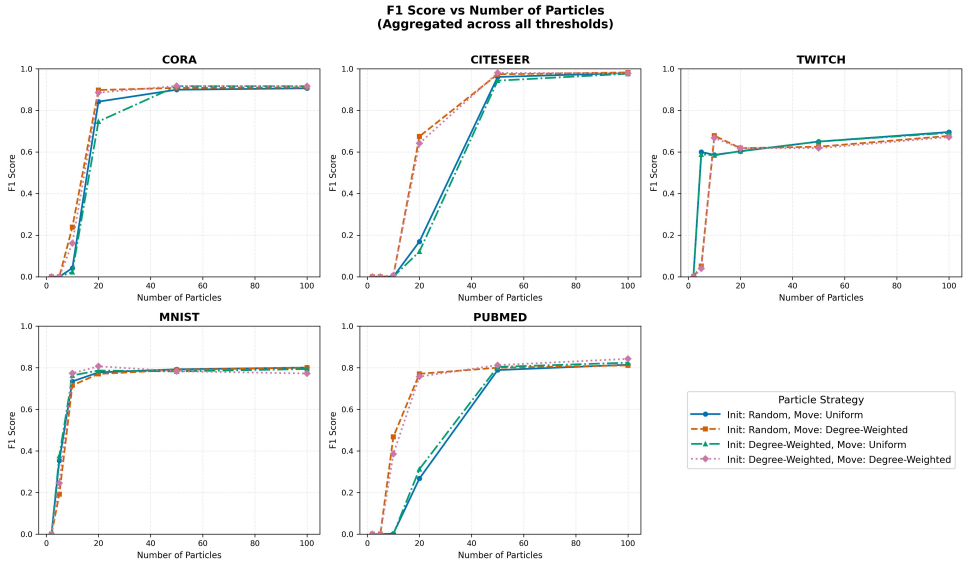


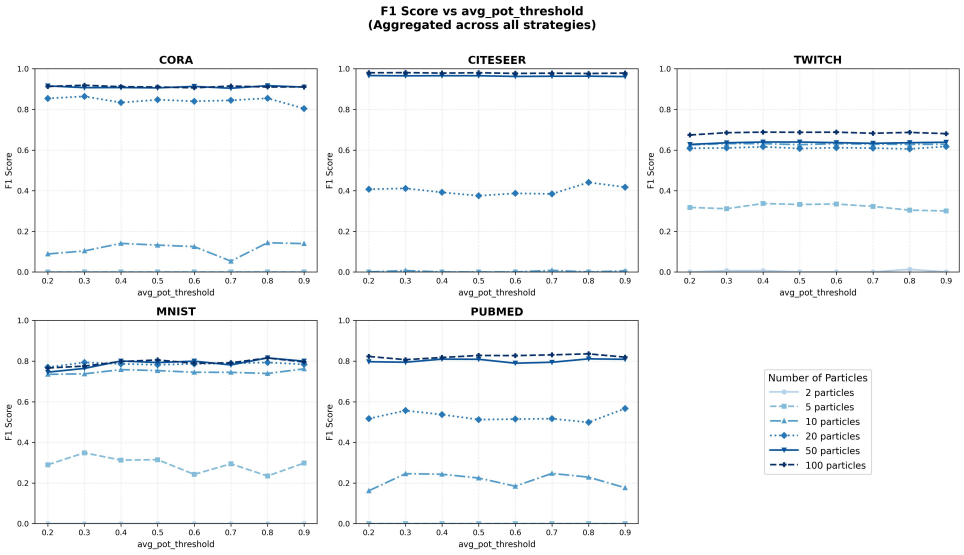
Figura 1. F1-score médio em função do número de partículas para as quatro combinações de estratégias, agregado sobre os limiares.

A Figura 1 ilustra o impacto do número de partículas e das estratégias de exploração no desempenho do algoritmo. Observa-se uma tendência clara de aumento do F1-score com o incremento do número de partículas em todos os conjuntos de dados, independentemente da estratégia adotada. Um número maior de partículas possibilita uma cobertura mais ampla do grafo, aumentando a probabilidade de que os agrupamentos formados separem adequadamente regiões positivas e negativas. Além disso, com $n_p = 100$, as diferentes combinações de inicialização e movimentação convergem para desempenhos similares, indicando que a quantidade de partículas é o fator dominante quando em valores elevados.

Esse comportamento contrasta com a recomendação do modelo original, em que o número de partículas é definido conforme o número esperado de comunidades a serem detectadas (e.g., $n_p = K$, o que resultaria em apenas duas partículas para uma tarefa binária) [Quiles et al. 2008]. No cenário de PUL, no entanto, valores tão baixos de n_p mostraram-se insuficientes: a seleção de RN depende da formação de múltiplos subgrupos dentro de cada classe, permitindo que os candidatos a negativos confiáveis sejam comparados topologicamente a diferentes regiões positivas, em vez de a uma única comunidade global.

4.2.2. Interação entre Número de Partículas e Critério de Parada

Figura 2. F1-score médio em função do limiar de potencial médio θ_{avg} para cada conjunto de dados.



A Figura 2 apresenta a relação entre o limiar de potencial médio θ_{avg} e o desempenho. Diferentemente do cenário original de detecção de comunidades, onde valores elevados de θ_{avg} (e.g., 0,9) garantem territórios consolidados, no contexto de pseudo-rotulagem esse parâmetro apresenta influência reduzida sobre o F1-score. Isso ocorre porque, na maioria das execuções, o critério de parada por estabilização é acionado antes que o potencial médio atinja o limiar configurado. Na prática, o algoritmo converge para um estado suficientemente informativo para a pseudo-rotulagem sem necessitar da consolidação territorial completa exigida na detecção de comunidades. Os resultados indicam que um F1-score elevado pode ser obtido com um número alto de partículas

mesmo quando θ_{avg} é baixo (e.g., 0,2), confirmando que o número de partículas é o hiperparâmetro mais determinante para o desempenho.

4.3. Avaliação de Desempenho

Esta seção apresenta a comparação entre o PC-PUL e os métodos da literatura em duas perspectivas: a qualidade da pseudo-rotulagem (primeira etapa) e o desempenho do classificador SVM treinado através de *hold-out* com os negativos selecionados (segunda etapa).

4.3.1. Pseudo-rotulagem

A Tabela 4 apresenta o F1-score da seleção de negativos confiáveis para diferentes frações de positivos conhecidos.

No conjunto CiteSeer, o PC-PUL obteve o melhor desempenho em todas as configurações, superando o LP-PUL (segundo colocado) em todas as frações e apresentando ampla vantagem sobre o PU-LP. No PubMed, o Roc-SVM liderou nas frações de 1%, 5% e 10% , o PU-LP foi superior na fração de 15%, e o PC-PUL alcançou os melhores resultados nas frações de 20% e 25%.

No MNIST, o PU-LP apresentou o melhor desempenho em todas as frações. O PC-PUL obteve resultados competitivos nas frações mais elevadas, alcançando 0,941 em 20% e 0,963 em 25%. No Cora, o PU-LP dominou em todas as configurações. Nas frações de 1% e 5%, o Roc-SVM ocupou a segunda posição, com o PC-PUL em terceiro. A partir de 20%, entretanto, o PC-PUL superou o Roc-SVM e passou a ocupar a segunda posição.

O Twitch representou o cenário mais desafiador para o PC-PUL, que obteve F1-score em torno de 0,70, abaixo do C-CRNE (melhor método nas frações de 5% a 25%) e do LP-PUL (melhor em 1%). O MCLS não foi capaz de selecionar o número mínimo de negativos confiáveis nesse conjunto.

4.3.2. Classificação

A Tabela 5 apresenta a acurácia do classificador SVM treinado com os conjuntos P e RN produzidos por cada método. Esta métrica reflete o impacto da qualidade da pseudo-rotulagem no desempenho final do sistema PUL completo.

No CiteSeer, o PC-PUL obteve a maior acurácia nas frações de 10%, 20% e 25% de positivos rotulados. No Cora, o PU-LP alcançou os melhores resultados nas frações de 1% e 10% enquanto o Roc-SVM liderou em 5%. A partir de 15%, o PC-PUL passou a obter os melhores resultados para as frações de 15%, 20% e 25%.

No MNIST, o Roc-SVM demonstrou superioridade consistente na classificação final a partir de 5%, sugerindo que, para dados vetoriais densos, a representação baseada em atributos do Roc-SVM se traduz melhor em fronteiras de decisão discriminativas no espaço do SVM. O PC-PUL liderou apenas em 1%.

No Twitch, o LP-PUL obteve os melhores resultados de classificação a partir de 5% de positivos. O PC-PUL ficou em segundo lugar na maioria das frações, com

Tabela 4. F1-score na pseudo-rotulagem para diferentes conjuntos de dados e frações de positivos rotulados.

Dataset	Model	0.01	0.05	0.10	0.15	0.20	0.25
Cora	Roc-SVM	0.907 ± 0.015	0.921 ± 0.010	0.929 ± 0.008	0.934 ± 0.008	0.939 ± 0.006	0.945 ± 0.007
	C-CRNE	0.867 ± 0.000	0.867 ± 0.000	0.870 ± 0.003	0.875 ± 0.003	0.877 ± 0.003	0.883 ± 0.005
	PU-LP	$\mathbf{0.980 \pm 0.000}$	$\mathbf{0.978 \pm 0.002}$	$\mathbf{0.979 \pm 0.002}$	$\mathbf{0.982 \pm 0.001}$	$\mathbf{0.984 \pm 0.003}$	$\mathbf{0.986 \pm 0.003}$
	MCLS	0.175 ± 0.392	0.538 ± 0.492	0.709 ± 0.399	0.552 ± 0.504	0.531 ± 0.485	0.708 ± 0.399
	LP-PUL	0.869 ± 0.033	0.886 ± 0.009	0.895 ± 0.008	0.904 ± 0.010	0.906 ± 0.009	0.912 ± 0.012
	PC-PUL	0.872 ± 0.061	0.891 ± 0.054	0.924 ± 0.037	0.929 ± 0.028	0.945 ± 0.027	0.952 ± 0.017
Citeseer	Roc-SVM	0.898 ± 0.021	0.928 ± 0.016	0.941 ± 0.012	0.952 ± 0.010	0.959 ± 0.007	0.965 ± 0.005
	C-CRNE	0.848 ± 0.001	0.850 ± 0.003	0.856 ± 0.006	0.861 ± 0.003	0.869 ± 0.008	0.872 ± 0.007
	PU-LP	0.292 ± 0.047	0.312 ± 0.034	0.344 ± 0.053	0.366 ± 0.014	0.415 ± 0.027	0.475 ± 0.034
	MCLS	0.922 ± 0.033	0.946 ± 0.024	0.951 ± 0.018	0.934 ± 0.027	0.952 ± 0.024	0.948 ± 0.017
	LP-PUL	0.975 ± 0.006	0.975 ± 0.002	0.976 ± 0.003	0.977 ± 0.004	0.979 ± 0.005	0.982 ± 0.004
	PC-PUL	$\mathbf{0.978 \pm 0.015}$	$\mathbf{0.985 \pm 0.008}$	$\mathbf{0.989 \pm 0.006}$	$\mathbf{0.986 \pm 0.005}$	$\mathbf{0.990 \pm 0.004}$	$\mathbf{0.988 \pm 0.003}$
MNIST	Roc-SVM	0.838 ± 0.033	0.888 ± 0.007	0.905 ± 0.016	0.912 ± 0.011	0.912 ± 0.009	0.920 ± 0.006
	C-CRNE	0.685 ± 0.002	0.673 ± 0.022	0.668 ± 0.021	0.547 ± 0.307	0.554 ± 0.311	0.415 ± 0.379
	PU-LP	$\mathbf{0.893 \pm 0.043}$	$\mathbf{0.959 \pm 0.024}$	$\mathbf{0.973 \pm 0.006}$	$\mathbf{0.979 \pm 0.006}$	$\mathbf{0.982 \pm 0.008}$	$\mathbf{0.986 \pm 0.005}$
	MCLS	0.655 ± 0.089	0.724 ± 0.158	0.564 ± 0.176	0.830 ± 0.044	0.878 ± 0.014	0.871 ± 0.020
	LP-PUL	0.868 ± 0.057	0.881 ± 0.021	0.862 ± 0.016	0.874 ± 0.020	0.880 ± 0.012	0.884 ± 0.013
	PC-PUL	0.708 ± 0.093	0.878 ± 0.051	0.903 ± 0.025	0.930 ± 0.045	0.941 ± 0.011	0.963 ± 0.023
Twitch	Roc-SVM	0.794 ± 0.014	0.794 ± 0.007	0.807 ± 0.006	0.818 ± 0.010	0.827 ± 0.007	0.834 ± 0.009
	C-CRNE	0.765 ± 0.093	$\mathbf{0.878 \pm 0.038}$	$\mathbf{0.910 \pm 0.047}$	$\mathbf{0.933 \pm 0.048}$	$\mathbf{0.938 \pm 0.044}$	$\mathbf{0.956 \pm 0.033}$
	PU-LP	0.680 ± 0.017	0.673 ± 0.011	0.683 ± 0.006	0.689 ± 0.012	0.698 ± 0.013	0.710 ± 0.028
	MCLS	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
	LP-PUL	$\mathbf{0.818 \pm 0.017}$	0.814 ± 0.007	0.824 ± 0.009	0.833 ± 0.006	0.839 ± 0.007	0.847 ± 0.006
	PC-PUL	0.691 ± 0.020	0.702 ± 0.034	0.715 ± 0.041	0.684 ± 0.033	0.705 ± 0.038	0.691 ± 0.020
PubMed	Roc-SVM	$\mathbf{0.865 \pm 0.012}$	$\mathbf{0.891 \pm 0.006}$	$\mathbf{0.903 \pm 0.005}$	0.909 ± 0.009	0.912 ± 0.009	0.920 ± 0.007
	C-CRNE	0.769 ± 0.000	0.772 ± 0.002	0.774 ± 0.012	0.788 ± 0.009	0.796 ± 0.008	0.800 ± 0.010
	PU-LP	0.662 ± 0.211	0.826 ± 0.053	0.889 ± 0.021	$\mathbf{0.918 \pm 0.013}$	0.913 ± 0.018	0.910 ± 0.017
	MCLS	0.783 ± 0.050	0.781 ± 0.071	0.805 ± 0.032	0.702 ± 0.052	0.770 ± 0.076	0.852 ± 0.065
	LP-PUL	0.846 ± 0.010	0.851 ± 0.006	0.860 ± 0.007	0.869 ± 0.008	0.875 ± 0.011	0.883 ± 0.009
	PC-PUL	0.840 ± 0.076	0.882 ± 0.037	0.882 ± 0.025	0.901 ± 0.018	$\mathbf{0.914 \pm 0.010}$	$\mathbf{0.929 \pm 0.009}$

acurácias próximas ao LP-PUL, apesar de seu desempenho inferior na pseudo-rotulagem. Observa-se que o C-CRNE, apesar de apresentar F1-scores elevados na pseudo-rotulagem do Twitch, não traduziu essa vantagem em acurácia de classificação proporcionalmente superior, sugerindo que a pureza dos negativos selecionados não é o único fator determinante para o classificador final. No PubMed, o Roc-SVM obteve a maior acurácia em cinco das seis frações avaliadas (5% a 25%), com o PC-PUL liderando apenas em 1%.

Tabela 5. Acurácia do classificador SVM para diferentes conjuntos de dados e fracções de positivos rotulados.

Dataset	Model	0.01	0.05	0.10	0.15	0.20	0.25
Cora	Roc-SVM	0.709 ± 0.003	$\mathbf{0.726 \pm 0.005}$	0.752 ± 0.006	0.756 ± 0.021	0.759 ± 0.011	0.767 ± 0.019
	C-CRNE	0.708 ± 0.004	0.721 ± 0.004	0.740 ± 0.006	0.735 ± 0.011	0.737 ± 0.008	0.741 ± 0.011
	PU-LP	$\mathbf{0.711 \pm 0.001}$	0.723 ± 0.003	$\mathbf{0.767 \pm 0.010}$	0.753 ± 0.034	0.684 ± 0.021	0.603 ± 0.023
	MCLS	0.514 ± 0.170	0.540 ± 0.235	0.602 ± 0.172	0.567 ± 0.185	0.523 ± 0.181	0.533 ± 0.164
	LP-PUL	0.710 ± 0.001	0.722 ± 0.005	0.758 ± 0.009	0.789 ± 0.018	0.761 ± 0.017	0.713 ± 0.014
	PC-PUL	0.708 ± 0.006	0.722 ± 0.001	0.751 ± 0.004	$\mathbf{0.793 \pm 0.009}$	$\mathbf{0.778 \pm 0.034}$	$\mathbf{0.768 \pm 0.029}$
Citeseer	Roc-SVM	0.751 ± 0.000	0.759 ± 0.001	0.779 ± 0.003	$\mathbf{0.813 \pm 0.004}$	0.844 ± 0.009	0.829 ± 0.016
	C-CRNE	0.751 ± 0.000	0.759 ± 0.000	0.771 ± 0.002	0.786 ± 0.008	0.789 ± 0.020	0.754 ± 0.036
	PU-LP	0.751 ± 0.000	0.759 ± 0.000	0.770 ± 0.002	0.759 ± 0.020	0.677 ± 0.059	0.605 ± 0.050
	MCLS	$\mathbf{0.751 \pm 0.000}$	$\mathbf{0.760 \pm 0.001}$	0.738 ± 0.036	0.622 ± 0.059	0.589 ± 0.104	0.540 ± 0.011
	LP-PUL	0.751 ± 0.000	0.760 ± 0.001	0.780 ± 0.003	0.812 ± 0.002	0.839 ± 0.009	0.831 ± 0.020
	PC-PUL	0.750 ± 0.000	0.759 ± 0.000	$\mathbf{0.783 \pm 0.001}$	0.812 ± 0.001	$\mathbf{0.856 \pm 0.002}$	$\mathbf{0.852 \pm 0.020}$
MNIST	Roc-SVM	0.524 ± 0.007	$\mathbf{0.786 \pm 0.031}$	$\mathbf{0.830 \pm 0.015}$	$\mathbf{0.837 \pm 0.009}$	$\mathbf{0.837 \pm 0.013}$	$\mathbf{0.840 \pm 0.010}$
	C-CRNE	0.517 ± 0.005	0.637 ± 0.042	0.613 ± 0.080	0.620 ± 0.125	0.599 ± 0.126	0.533 ± 0.045
	PU-LP	0.532 ± 0.018	0.767 ± 0.054	0.797 ± 0.024	0.783 ± 0.021	0.778 ± 0.033	0.777 ± 0.015
	MCLS	0.518 ± 0.055	0.580 ± 0.080	0.549 ± 0.025	0.606 ± 0.028	0.618 ± 0.018	0.600 ± 0.014
	LP-PUL	0.537 ± 0.021	0.695 ± 0.025	0.683 ± 0.023	0.694 ± 0.016	0.685 ± 0.017	0.666 ± 0.016
	PC-PUL	$\mathbf{0.550 \pm 0.011}$	0.705 ± 0.033	0.732 ± 0.012	0.750 ± 0.019	0.762 ± 0.018	0.753 ± 0.006
Twitch	Roc-SVM	$\mathbf{0.720 \pm 0.026}$	0.623 ± 0.009	0.609 ± 0.017	0.604 ± 0.012	0.610 ± 0.009	0.608 ± 0.007
	C-CRNE	0.509 ± 0.051	0.593 ± 0.026	0.585 ± 0.016	0.578 ± 0.009	0.575 ± 0.019	0.564 ± 0.015
	PU-LP	0.509 ± 0.052	0.506 ± 0.039	0.498 ± 0.050	0.507 ± 0.037	0.530 ± 0.038	0.517 ± 0.057
	MCLS	0.580 ± 0.021	0.589 ± 0.013	0.551 ± 0.020	0.539 ± 0.020	0.520 ± 0.004	0.506 ± 0.003
	LP-PUL	0.654 ± 0.034	$\mathbf{0.671 \pm 0.022}$	$\mathbf{0.689 \pm 0.013}$	$\mathbf{0.682 \pm 0.013}$	$\mathbf{0.692 \pm 0.007}$	$\mathbf{0.694 \pm 0.011}$
	PC-PUL	0.578 ± 0.003	0.629 ± 0.011	0.666 ± 0.024	0.630 ± 0.019	0.648 ± 0.015	0.614 ± 0.020
PubMed	Roc-SVM	0.712 ± 0.014	$\mathbf{0.752 \pm 0.012}$	$\mathbf{0.674 \pm 0.006}$	$\mathbf{0.614 \pm 0.003}$	$\mathbf{0.568 \pm 0.003}$	$\mathbf{0.532 \pm 0.005}$
	C-CRNE	0.663 ± 0.005	0.740 ± 0.013	0.606 ± 0.006	0.534 ± 0.004	0.477 ± 0.013	0.433 ± 0.014
	PU-LP	0.457 ± 0.090	0.487 ± 0.026	0.482 ± 0.009	0.472 ± 0.007	0.449 ± 0.009	0.426 ± 0.009
	MCLS	0.611 ± 0.011	0.544 ± 0.032	0.477 ± 0.011	0.434 ± 0.007	0.408 ± 0.007	0.390 ± 0.007
	LP-PUL	0.625 ± 0.021	0.595 ± 0.018	0.533 ± 0.006	0.492 ± 0.006	0.460 ± 0.005	0.431 ± 0.005
	PC-PUL	$\mathbf{0.716 \pm 0.041}$	0.697 ± 0.028	0.628 ± 0.029	0.577 ± 0.025	0.520 ± 0.014	0.514 ± 0.012

5. Conclusão

Este trabalho apresentou o PC-PUL, uma adaptação da competição de partículas para a seleção de negativos confiáveis em PUL. A análise paramétrica mostrou que o número de partículas exerce o efeito mais consistente sobre o desempenho, enquanto o limiar de convergência θ_{avg} apresenta influência reduzida. Nos experimentos, o PC-PUL obteve resultados superiores ou competitivos em parte dos cenários avaliados, com destaque para a seleção de negativos confiáveis, embora não tenha superado os métodos de comparação em todas as bases. Os ganhos na pseudo-rotulagem também se refletiram na classificação final em algumas configurações.

Os resultados indicam que a formação de comunidades por competição de partículas constitui uma alternativa promissora para pseudo-rotulagem baseada em grafos. Entretanto, a variabilidade observada entre os conjuntos de dados sugere que sua eficácia depende das características topológicas e da relação entre a estrutura do grafo e as classes. Como trabalhos futuros, pretende-se avaliar essa relação por meio de medidas de homofilia e estrutura de comunidades, investigar estratégias automáticas para definir o número de partículas e ampliar a avaliação para outros conjuntos de dados e métodos de PUL.

Agradecimentos

Este estudo foi financiado em parte pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001. Os autores também agradecem à Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) (Processo nº 22/03090-0) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) (Processos nº 420025/2023-5). Os autores também reconhecem o uso de ferramentas baseadas em IA para auxiliar na revisão do texto e na organização dos experimentos.

Referências

Bekker, J. and Davis, J. (2020). Learning from positive and unlabeled data: a survey. *Machine Learning*, 109(5):703–750.

Breve, F., Zhao, L., Quiles, M., Pedrycz, W., and Liu, J. (2012). Particle competition and cooperation in networks for semi-supervised learning. *Knowledge and Data Engineering, IEEE Transactions on*, 24:1686–1698.

Breve, F. A., Zhao, L., and Quiles, M. G. (2010). Semi-supervised learning from imperfect data through particle cooperation and competition. In *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Carnevali, J. C., Rossi, R. G., Milios, E., and Lopes, A. d. A. (2021). A graph-based approach for positive and unlabeled learning. *Information Sciences*, 580:655–672.

Chaudhari, S. and Shevade, S. (2012). Learning from positive and unlabelled examples using maximum margin clustering. In Huang, T. et al., editors, *Neural Information Processing*, pages 465–473, Berlin, Heidelberg. Springer Berlin Heidelberg.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271.

Hagberg, A. A., Schult, D. A., and Swart, P. J. (2008). Exploring network structure, dynamics, and function using networkx. In Varoquaux, G., Vaught, T., and Millman, J., editors, *Proceedings of the 7th Python in Science Conference*, pages 11–15, Pasadena, CA USA.

Jaskie, K. and Spanias, A. (2019). Positive and unlabeled learning algorithms and applications: A survey. In *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*, pages 1–8.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Li, X. and Liu, B. (2003). Learning to classify texts using positive and unlabeled data. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, pages 587–592.

Liu, L. and Peng, T. (2014). Clustering-based method for positive and unlabeled text categorization enhanced by improved TFIDF. *Journal of Information Science and Engineering*, 30(5):1463–1481.

Ma, S. and Zhang, R. (2017). PU-LP: A novel approach for positive and unlabeled learning by label propagation. In *2017 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pages 537–542.

Messias, G. H., Iasulaitis, S., and Valejo, A. D. B. (2025). Graph neural networks for positive and unlabeled learning: a rewiring approach. *Knowledge and Information Systems*.

Namata, G., London, B., Getoor, L., Huang, B., and Edu, U. (2012). Query-driven active surveying for collective classification. In *Proceedings of the 10th International Workshop on Mining and Learning with Graphs*.

Quiles, M. G., Zhao, L., Alonso, R. L., and Romero, R. A. F. (2008). Particle competition for complex network community detection. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 18(3):033107.

Rozemberczki, B., Allen, C., and Sarkar, R. (2021). Multi-scale attributed node embedding. In *Journal of Complex Networks*, volume 9, page cnab014.

Sen, P., Namata, G., Bilgic, M., Getoor, L., Galligher, B., and Eliassi-Rad, T. (2008). Collective classification in network data. *AI Magazine*, 29(3):93–106.

Souza, M. C. d., Rossi, R. G., Rezende, S. O., and Carvalho, A. C. P. L. F. d. (2021). A network-based positive and unlabeled learning approach for fake news detection. *Machine Learning*, 111(10):3549–3592.

Wu, M., Pan, S., Du, L., and Zhu, X. (2021). Learning graph neural networks with positive and unlabeled nodes. *ACM Transactions on Knowledge Discovery from Data*, 15(6):1–25.

Yang, P., Yang, Y. H., Zhou, B. B., and Zomaya, A. Y. (2012). Positive-unlabeled learning for disease gene identification. *Bioinformatics*, 28(20):2640–2647.

Zhu, X. and Goldberg, A. B. (2009). *Introduction to Semi-Supervised Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan and Claypool Publishers.