Cluster Fusion Training: Exploring Cluster Analysis to Enhance Cross-Domain Sentiment Classification

Victor Akihito Kamada Tomita[1], **Angelo Cesar Mendes da Silva** [1], **Ricardo Marcondes Marcacini**[1]

[1]Instituto de Ciências Matemáticas e de Computação (ICMC)
Universidade de São Paulo (USP)
Caixa Postal 668 – 13566-590 – São Carlos – SP – Brasil

{akihito012,angelo.mendes,ricardo.marcacini}@usp.br

Abstract. *Due to the data scarcity for specific domains, many studies choose to train models in cross-domains. The most common approach involves training models on all source domains and then validating their performance on the target domain. However, this approach does not consider that different words may have distinct semantics depending on the domain. In this article, a new method is proposed that utilizes clustering techniques to group similar data. Expert models are trained from these groups, which are used in a fusion process. The proposed method demonstrates significant improvements up to* 5% *in accuracy for classification models.*

Resumo. *Devido à escassez de dados para domínios específicos, muitos estudos optam por treinar modelos em domínios cruzados. A abordagem mais comum consiste em treinar modelos em todos os domínios-fonte e, em seguida, validar seu desempenho no domínio-alvo. Mas, essa abordagem não leva em consideração que diferentes palavras podem ter semânticas distintas dependendo do domínio. Neste artigo, é proposto um novo método que usa técnicas de clustering para agrupar dados semelhantes. A partir desses grupos, são treinados modelos especialistas que são usados em um processo de fusão. Através desse método, são demonstradas melhorias até significativas de até* 5% *de acurácia para modelos de classificação.*

1. Introdução

Com a popularização do acesso à internet, e o constante crescimento no número de pessoas conectadas às redes sociais, tem surgido uma quantidade massiva de dados contendo opiniões sobre uma ampla variedade de tópicos [Ortiz-Ospina and Roser 2023]. Com intuito de construir conhecimento sobre esse volume de dados, diversos setores, como partidos políticos e empresas, buscam extrair informações relevantes sobre seus produtos e candidatos políticos. Por exemplo, as empresas visam compreender como direcionar o desenvolvimento de seus produtos e suas estratégias de *marketing* [Sivarajah et al. 2020]. De forma similar, os partidos políticos têm interesse em compreender a intenção de voto da população, analisando as opiniões públicas sobre assuntos políticos [Liao et al. 2020]. A abundância de dados gerados pelas mídias sociais tornou-se um recurso crucial para a tomada de decisões, impulsionando a necessidade de técnicas eficientes para análise desses dados não estruturados [Bousdekis et al. 2021].

Ao considerar a volumosa quantidade de dados, a análise manual dessas informações se torna inviável, principalmente devido ao tempo necessário para realizar tal avaliação. Em resposta a esse problema, a análise de sentimentos foi desenvolvida para extrair automaticamente as reações emocionais expressas por um determinado usuário [Zhang et al. 2018]. As abordagens atuais de análise de sentimentos utilizam redes neurais que, em geral, se beneficiam do uso de dados anotados para o treinamento desses modelos [Ain et al. 2017, Zhang et al. 2018, Dang et al. 2020].

Este constante crescimento de dados, não necessariamente se converte em um melhor desempenho para domínios específicos, isso implica em um baixo desempenho ao se considerar cenários específicos de empresas [Tseng et al. 2020]. Por essa razão, o uso de técnicas de domínio cruzado tem ganhado destaque, em que tradicionalmente treinam-se modelos de língua (LM) em diferentes domínios e avalia-se seu desempenho no domínio alvo [Dos Santos et al. 2021, Ma et al. 2023]. No entanto, utilizar diferentes domínios durante o treinamento também gera limitações no treinamento dos modelos. Em particular, a polaridade do sentimento pode divergir dependendo do contexto em que uma palavra é utilizada. Por exemplo, a palavra *"quente"* na frase *"O café está quente"* apresenta uma polaridade de sentimento positiva, enquanto na frase *"A cerveja está quente"* uma polaridade de sentimento negativa. Como resultado, mesmo considerando modelos dependentes de contexto, o treinamento por meio de dados de múltiplos domínios, que não correspondem ao domínio-alvo, pode levar a resultados insatisfatórios.

Neste trabalho, é proposto uma abordagem que combina técnicas de *clustering* e fusão de modelos de classificação para construir um método capaz de lidar com o problema de classificação de sentimentos em dados textuais contidos em domínios cruzados. O método proposto, denominado *Clustering Fusion Training* (CFT), tem como objetivo incorporar o conhecimento obtido por múltiplos modelos, treinados sobre dados agrupados, em um único modelo utilizando um processo de fusão para classificar o sentimento em dados pertencentes a um domínio desconhecido aos modelos treinados. A etapa de *clustering* aplicada em dados de múltiplos domínios permite identificar subconjunto de dados compartilhado entre os vários domínios fonte. Assim, modelos mais especialistas são treinados em cada *cluster*, mas mantendo características desejáveis de treinamento envolvendo domínio cruzado. Em resumo, as principais contribuições do trabalho são:

- Introduziu-se o método CFT como uma abordagem eficaz para lidar com o treinamento em múltiplos domínios. O CFT utiliza técnicas de *clustering* para identificar *clusters* semelhantes de dados de diferentes domínios, sobre os quais treina modelos especialistas para cada *cluster*. Em seguida, aplica técnicas de fusão para combinar essas representações geradas pelos modelos especialistas, com o intuito de treinar modelos de classificação robustos à mudança de domínio;
- Ao utilizar o CFT, foi possível melhorar o desempenho em domínios específicos, evitando a perda de desempenho causada pelo treinamento em múltiplos domínios que não correspondem ao domínio-alvo. Os resultados demonstram a eficácia do método em comparação com abordagens tradicionais de treinamento em múltiplos domínios;
- O método CFT é flexível e pode ser aplicado a diferentes problemas de classificação em diversos domínios, permitindo que o modelo seja facilmente ajustado e refinado para lidar com novos conjuntos de dados.

2. Fundamentos e Trabalhos Relacionados

A análise de sentimentos é uma técnica que permite extrair automaticamente o sentimento contido em uma opinião expressa por um usuário em relação a um determinado alvo [Zhang et al. 2018, Yadav and Vishwakarma 2020, Kaur et al. 2021]. Considerando o contexto de dados textuais, essa técnica pode ser aplicada em diferentes níveis, como documento, sentença, palavra e aspecto. No nível de documento, busca-se avaliar a orientação da opinião em relação a uma entidade específica em todo o documento. Nessa tarefa, o objetivo é determinar a polaridade do sentimento $o(d)$ em relação ao documento d como um todo [Wankhade et al. 2022].

Umas das dificuldades encontradas na análise de sentimentos a nível de documento, diz respeito à quantidade de tópicos e opiniões que são apresentadas ao longo do texto. Mesmo para análises mais simples, como polaridade expressa pelo usuário, ainda há flutuações e variações. Por exemplo, o usuário pode gostar do enredo de um livro, mas detestar os personagens. Essas alterações de polaridade, podem gerar inconsistências ao se gerar apenas um sentimento expresso ao longo de um documento. Em função disso, é comum ocorrer o uso de análise de sentimentos a nível de sentença, o que permite reduzir significativamente a variação de sentimentos [Nandwani and Verma 2021]. Por isso, muitas aplicações de processamento de linguagem natural utilizam o nível sentença como unidade de processamento, por exemplo *parsing* sintático, que busca gerar as dependências e relações sintáticas dos elementos de uma sentença [Birjali et al. 2021]. Isto posto, o objetivo da análise de sentimentos a nível de sentença é: dado uma sentença s, busca-se determinar a polaridade do sentimento $o(s)$.

Uma das desvantagens da análise de sentimentos a nível de sentença é que por vezes o sentimento precisa ser anotado manualmente, enquanto o sentimento a nível de documento geralmente é encontrado naturalmente em uma anotação feita pelo próprio usuário, como em casos de fóruns e páginas de avaliações de produtos. Essa disponibilidade de dados anotados facilita a criação de *datasets* para análise de sentimentos a nível de documento, como o caso do *IMDB dataset* [Maas et al. 2011, Sikhi et al. 2022]. Em geral, a análise de sentimentos em termos de sentença é complexa porque a orientação semântica das palavras depende muito do contexto. Já em termos de documento, a complexidade é gerada pela possibilidade de haver divergência de opinião em relação à mesma entidade [Araújo et al. 2020].

Análise de sentimentos a nível de palavras busca especificar a polaridade de sentimento $o(p)$ para determinada palavra p. Ao se considerar um corpus, o objetivo desta tarefa se torna equivalente à construção de léxico de sentimento. Sob uma perspectiva geral, a maioria dos léxicos de sentimento de propósito geral são construídos de forma manual, isso funciona para muitos casos, porém para domínios específicos estas regras podem falhar. Em meio a isso, observa-se que a construção desses léxicos se demonstra custosa em questão de tempo e trabalho humano, por isso muitas pesquisas têm focado na construção de léxicos de sentimento de forma automática [Zong et al. 2021].

Pesquisas relacionadas à análise de sentimentos tem apresentado um progresso significativo a partir do surgimento de métodos baseados em redes neurais profundas (DNN) [Zhang et al. 2018, Yadav and Vishwakarma 2020, Sarker 2021]. Este progresso está associado à crescente disponibilidade de dados gerada por aplicações *web* que produzem dados em grande volume, permitindo a realização de um processo de *fine-tuning*

efetivo dos pesos das DNNs [Habimana et al. 2020].

Um modelo notável empregado para tarefas de análise de sentimentos é o *Bidirectional Encoder Representations from Transformers* (BERT), que é um modelo bidirecional baseado no *encoder* dos *transformers* treinado para as tarefas de *Masked Language Modeling* e *Next Sentence Prediction* [Devlin et al. 2018, Singh et al. 2021]. Esses modelos baseados em *transformers* são frequentemente considerados o estado da arte para análise de sentimentos em textos [Silva and Marcacini 2021]. No entanto, eles exigem uma grande quantidade de dados para um treinamento adequado. Para mitigar esse problema, técnicas de domínio cruzado podem ser utilizadas [Sun et al. 2021].

Uma das características mais significativas dessas arquiteturas baseadas *transformers* é o mecanismo de atenção [Vaswani et al. 2017, Niu et al. 2021, Guo et al. 2022]. Esse mecanismo busca solucionar o problema da perda de informações durante o processamento de sinais até o decodificador e determina a relação entre duas partes de sequências textuais [Nassiri and Akhloufi 2022].

Visando a melhoria de modelos de classificação, o uso de técnicas de agrupamento tem se mostrado relevantes para aprimorar redes neurais em tarefas de classificação [Alapati and Sindhu 2016, Song and Yang 2022]. Essas propostas possibilitam identificar possíveis grupos semânticos presentes dentro da base de treino, o que pode ser uma melhoria potencial para o treinamento por meio de domínios cruzados.

Os métodos tradicionais de domínios cruzados realizam o treinamento utilizando conjuntos de dados de diferentes domínios e aplicam o modelo resultante em um domínio específico [Piernik and Morzy 2021, Dos Santos et al. 2021, Ma et al. 2023]. No entanto, essa abordagem tende a desconsiderar as contradições entre tópicos diferentes, o que pode levar a resultados inferiores no domínio-alvo. Abordar esse problema de pesquisa pode ter implicações importantes para aplicações do mundo real, especialmente porque muitas empresas treinam classificadores de sentimentos para diferentes domínios.

Para lidar com essa questão, uma abordagem interessante é usar técnicas de agrupamento para definir diferentes domínios em um conjunto de dados. Dessa forma, é possível criar conjuntos de treinamento específicos para cada domínio, tendo-se como hipótese que o treino a partir das instâncias de um mesmo grupo podem melhorar o desempenho dos modelos de classificação. Ao estabelecer múltiplos modelos para cada domínio, é possível explorar técnicas de fusão para construir um modelo único que incorpore o conhecimento dos demais modelos.

3. Clustering Fusion Training (CFT)

Através deste artigo é proposto um novo método para classificação de sentimentos que gera comitês de modelos para domínios específicos por meio de *clustering* e utiliza modelos de fusão para combinar as representações geradas pelos diferentes modelos do comitê. O objetivo é prever o sentimento $o(d)$ associado a um determinado documento d. A abordagem proposta tem como objetivo lidar com o problema em casos em que há falta de dados rotulados ao considerar domínios específicos de análise de sentimentos. Para isso, adotou-se uma abordagem em três etapas, conforme mostrado na Figura 1.

Com o objetivo de representar esses casos, utilizou-se o domínio cruzado, conforme demonstrado na primeira etapa do método. A composição inicial dos conjuntos de

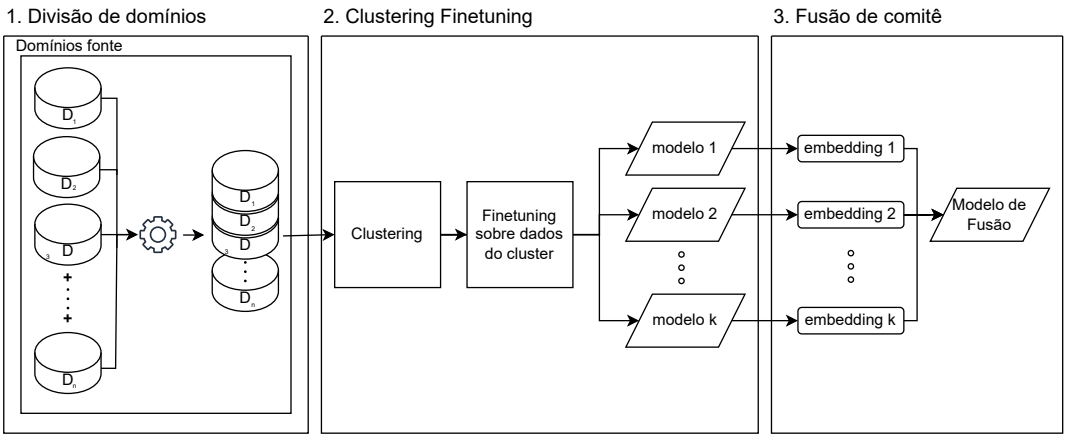

Figura 1. Método proposto de treinamento por meio de *clustering* associado à fusão. Sendo que o primeiro passo representa o domínio cruzado. Na segunda etapa, tem-se o processo de treinamento sobre *clusters*, que gera modelos especialistas para cada *cluster* identificado. Por fim, a terceira fase mostra o processo de fusão, realizado através da extração das *embeddings* de cada modelo especialista e sua combinação para obter uma melhor representação.

dados é realizada a partir de sete bases diferentes. Para garantir um balanceamento adequado entre os domínios e evitar que um domínio tenha mais dados do que outro, extraiu-se aleatoriamente dez mil exemplos de cada base, considerando também as classes de dados para essa amostragem. Dessa forma, a composição inicial consiste em setenta mil textos únicos, com trinta e cinco mil amostras de cada classe.

No decorrer do processo, em cada *fold*, utilizou-se cinco conjuntos de dados para o treinamento dos modelos, assim compondo os n domínios-fonte representados na primeiro bloco da imagem, um domínio D_{n+1} para validação e um domínio D_{n+2} para teste. Esse particionamento é realizado para mensurar a capacidade de generalização dos modelos em diferentes conjuntos de dados.

Sobre a base de treino, foram inicialmente extraídas as *embeddings* dos textos, que foram utilizadas para agrupar os dados em k *clusters*. O objetivo é gerar domínios específicos nos quais seja possível treinar k modelos de classificação, em vez de treinar apenas um modelo genérico em todo o conjunto de treinamento. Esses modelos são capazes de realizar predições sobre os textos do conjunto de teste, com base na estratégia de *clustering* utilizada. Em relação à base de dados de teste, ela pode ser representada por um domínio-alvo D_{n+2}, que se diferencia dos domínios-fonte.

Para isso, utilizou-se um modelo de *clustering* treinado com o conjunto de treino para identificar qual *cluster* os dados de teste pertencem. Dessa forma, pode-se classificar os textos do conjunto de teste utilizando a informação do *cluster* associado a cada dado.

Na terceira etapa, extraiu-se as *embeddings* dos textos por meio dos k modelos especialistas, com o objetivo de obter representações mais específicas. Essa abordagem é adotada para possibilitar a fusão tardia dessas representações. A fusão tardia é realizada por uma rede neural, que utiliza operações de atenção sobre essas representações. Essa técnica permite a integração das diferentes representações geradas pelos modelos especialistas, em que se espera uma melhoria na classificação final.

3.1. Clustering

Um dos principais pontos do método proposto está relacionado ao estabelecimento de domínios específicos. Para isso, propõe-se o uso do algoritmo *k-means* [Ahmed et al. 2020] para realizar o *clustering* dos dados, juntamente com o modelo de linguagem pré-treinado *MiniLM-v2* [Wang et al. 2020] para a extração das representações textuais. O *MiniLM-v2* é treinado usando o processo de destilação, tendo como modelo professor o $RoBERTa_{large}$ [Liu et al. 2019, Sirusstara et al. 2022]. Nesse método de compressão, um modelo menor, chamado de modelo aluno, é treinado para reproduzir o comportamento de um modelo maior, conhecido como modelo professor [Hinton et al. 2015]. Em específico, as camadas de *self-attention* são reduzidas a um menor tamanho, sendo a principal inovação a introdução de *multi-head self-attention relations*. Essas relações calculam o produto escalar entre pares de *queries*, *keys* e *values*, utilizados para guiar o treinamento do modelo aluno.

Através do processo de destilação, o *MiniLM-v2* é capaz de simular as relações estabelecidas entre as *multi-head self-attention* do modelo professor. Isso permite a geração de representações textuais significativas com um custo computacional relativamente baixo. A partir dessas representações, é possível formar *clusters* que serão utilizados para o treinamento dos modelos específicos.

Em relação à tarefa de *clustering* utilizando o algoritmo *k-means*, pode-se formular da seguinte forma: dado um conjunto de dados com n pontos, almeja-se particionar esses dados em k *clusters*, de modo a minimizar a soma dos quadrados das distâncias dentro de cada *cluster*. Denotando o conjunto de dados como $X = \{x_1, x_2, x_3, \cdots, x_n\}$, em que cada ponto x_i representa um vetor de dimensão d, e $C = \{c_1, c_2, \cdots, c_k\}$ como o conjunto de k centroides, em que cada centroide c_j é um vetor de dimensão d, e $S = \{s_1, s_2, s_3, \cdots, s_k\}$ como o conjunto de k *clusters*. O objetivo é minimizar a soma dos quadrados das distâncias, que pode ser descrito por:

$$\sum_{j=1}^{k} \sum_{i=1}^{t_j} \|x_i - c_j\|^2 \tag{1}$$

sendo t_j o número de pontos relacionado ao centroide c_j [Kriegel et al. 2017]. A partir dos k centroides obtidos por meio do algoritmo *k-means*, é feito a predição de qual *cluster* os dados de teste pertencem, atribuindo-os ao *cluster* que possui o centroide mais próximo. Essa etapa permite identificar qual modelo especialista será utilizado para realizar a classificação.

3.2. Modelos especialistas

A partir dos dados agrupados, é realizado o treinamento dos modelos de classificação específicos para cada *cluster* de forma individual. Neste trabalho, três modelos baseados em *transformers* foram escolhidos: *DistilBERT* [Sanh et al. 2019], *BERT* [Devlin et al. 2018] e *DeBERTa-v3* [He et al. 2021]. Esses modelos foram utilizados para avaliar o impacto do método proposto em diferentes domínios, mensurando sua capacidade de generalização.

O *DistilBERT* foi treinado para que suas *embeddings* sejam o mais próximo possível dos vetores gerados pelo *BERT* [Sanh et al. 2019, Adoma et al. 2020]. A ideia

geral do modelo é remover o maior número de camadas possível do *BERT*, mantendo ao mesmo tempo a distribuição probabilística gerada pelo *BERT*. Para alcançar esse objetivo, é utilizada a técnica conhecida como destilação de conhecimento.

No caso do *DistilBERT*, o modelo aluno é treinado utilizando uma *distillation loss* aplicada às probabilidades suavizadas do modelo professor, que nesse caso são as *embeddings* geradas pelo *BERT*. O objetivo é que o modelo aluno, o *DistilBERT*, seja capaz de capturar informações semelhantes ao modelo professor, mas em um formato mais compacto.

O *BERT* é um modelo pré-treinado baseado no *encoder* dos *transformers* [Devlin et al. 2018, Singh et al. 2021]. Ele é caracterizado como um modelo bidirecional, o que significa que ele pode capturar informações contextuais tanto à esquerda quanto à direita de uma determinada palavra em um texto. O *BERT* foi projetado para realizar tarefas como predição de palavras mascaradas e predição de próxima sentença. Essas tarefas são usadas no treinamento do modelo para que ele possa aprender a representação das palavras em um contexto mais amplo. O BERT teve um impacto significativo no campo do processamento de linguagem natural e tem sido amplamente utilizado como uma base para o desenvolvimento de modelos mais avançados [Lan et al. 2019, Liu et al. 2019].

O *DeBERTa* é um modelo baseado no *BERT* que traz uma inovação chamada *"desintangle attention"*, um mecanismo no qual cada palavra é representada por dois vetores diferentes que codificam tanto a posição quanto o conteúdo da palavra. Esse mecanismo permite que os pesos das matrizes de atenção entre palavras sejam computados usando matrizes separadas em relação aos conteúdos e posições relativas das palavras [He et al. 2020]. Essa abordagem do *DeBERTa* permite uma separação adequada entre os espaços de representação sintáticos e semânticos das palavras.

No contexto da classificação, uma camada densa foi adicionada aos modelos mencionados, que recebe como entrada a *embedding* gerada pelo *token CLS*. Essa camada tem como objetivo gerar a classificação final, associada à polaridade de sentimento do texto.

A partir dessas representações, utilizou-se a *embedding* associada ao *token CLS* para gerar a classificação. Essa *embedding* serve como entrada para uma camada densa de classificação, sobre a qual a função de ativação *softmax* foi aplicada. Dessa forma, a função de perda da rede pode ser formalizada da seguinte forma:

$$\begin{aligned} p_i = \text{softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{C} e^{x_j}} \\ L = -\sum_i y_i \cdot \log(p_i) \end{aligned} \tag{2}$$

Sendo i o índice da classe, p_i a probabilidade predita para a classe i, x_i a saída do neurônio i da camada de classificação, C o número de classe, y_i é uma variável binária relacionada à classe verdadeira i, podendo ser 0 ou 1.

Assim, é possível fazer *finetuning* destes modelos baseados em *transformers* para classificação de sentimentos. Estes treino é feito k vezes, uma vez para cada *cluster*, assim gerando k modelos especialistas diferentes.

3.3. Modelo de fusão

O uso de *clustering* sobre o conjunto de treino tem como objetivo estabelecer domínios específicos dentro do mesmo, os quais são utilizados para treinar modelos especialistas. Esses modelos especialistas são capazes de extrair representações mais precisas para cada *cluster* específico. A partir dessas novas representações, é possível fundi-las por meio de atenção, assim gerando uma representação ainda melhor.

Para o método de fusão proposto, são utilizadas duas camadas de *multihead attention* compostas por 8 *attention heads*, seguidas por camadas densas de classificação. O objetivo dessa abordagem é identificar qual das representações geradas pelos modelos especialistas apresenta maior relevância. A formalização de *multihead attention* é a seguinte:

$$\text{MultiHead}(q, K, V) = \text{Concat}(head_1, \cdots, head_i, \cdots, head_h)W_O$$
$$head_i = \text{Attention}(qW_Q^i, KW_K^i, VW_V^i)$$
$$\text{Attention}(q, K, V) = \text{softmax}\left(\frac{qK^T}{\sqrt{d_k}}\right)V$$

Sendo que *query*, as *keys*, e *values* são respectivamente representados por *q*, *K* e *V*, com d_k representado a dimensão da chave, *Concat* simboliza a concatenação da representação de cada cabeça, e W_Q^i, W_K^i, W_V^i e W_O denotam as matrizes de parâmetros de projeção [Vaswani et al. 2017].

Através de *multihead attention* busca-se ressaltar qual representação gerada pelos modelos especialistas é mais relevante para a classificação, assim melhorando o resultado final do método.

4. Avaliação Experimental

Nesta seção, apresenta-se os três experimentos realizados em que foram utilizados o modelo *MiniLM-v2* em conjunto com o algoritmo *k-means* para realizar a tarefa de agrupamento dos dados. A partir dos *clusters* formados, um modelo baseado em *transformers* foi treinado para cada *cluster*. Em seguida, as *embeddings* de todos os dados de treinamento são extraídas e inseridas como entrada para o treinamento de um modelo de fusão composto por duas camadas de *multihead attention* seguido de uma camada densa de classificação.

Sete conjuntos de dados diferentes foram utilizados para avaliação. Para cada iteração, selecionamos uma base de teste, cinco bases de treinamento e uma base de validação. A base de validação é mantida constante em todas as iterações, a fim de evitar múltiplas execuções apenas trocando a base de validação. Essa configuração experimental foi adotada para simular o treinamento em domínios cruzados, em que não há dados rotulados para os domínios-alvo.

4.1. Bases de dados

Entre as sete base de dados utilizadas, pode-se citar o *Tweet Eval*, o qual é composto por *tweets* coletados da quarta tarefa do SemEval-2017 [Rosenthal et al. 2019], esta base é composta de *tweets* baseados em eventos e tópicos populares dentro da plataforma,

variando desde entidades específicas, como Donald Trump, às entidades geopolíticas, como Palestina e outras questões políticas [Rosenthal et al. 2017].

Outra base de dados escolhida foi o *Rotten Tomatoes*, esta base é composta por *reviews* de filmes extraídos da plataforma Rotten Tomatoes, sendo que textos que apresentavam indicadores explícitos da nota, foram removidos do documento [Pang and Lee 2005].

O *Multilingual Amazon Reviews Corpus* (MARC), é uma base de dados compostas por *reviews* de produtos extraídos do *marketplace* da Amazon, todos produtos apresentados nesta base precisam conter mais de dois *reviews* associados. Além disso, não há mais de 20 *reviews* do mesmo produto nem mais de 20 *reviews* do mesmo usuário [Keung et al. 2020].

O *Yelp dataset* é composto por *reviews* extraídos de publicações feitas no site *Yelp*, sendo composto por várias *reviews* de negócios, produtos e serviços, esta base também apresenta as informações referentes aos negócios e usuários [Asghar 2016].

O *Sequence Labelling Evaluation Benchmark for Spoken Language* (SILICONE) é um *benchmark* criado para treinar e avaliar sistemas de *natural language understanding*. Este conjunto de dados contém uma variedade de domínios como textos referentes aos dia-a-dia, cenários planejados, conversas ao telefone e conversas de televisão [Chapuis et al. 2020].

O *Twitter Reddit Sentimental Analysis Dataset* é uma base composta por textos extraídos dos *sites Reddit* e do *Twitter*, para isso foi feito o uso das *APIs Tweetpy* e *PRAW* [Gowda et al. 2019]. Para o uso desta base foram removidos os dados de *tweets* uma vez que já há dados de *tweets* referentes ao *Tweet Eval*.

O *App Reviews dataset* é composta por *reviews* de *apps* para *antroid*, sendo que essas aplicações pertencem à 23 diferentes categorias [Grano et al. 2017]. Estes dados foram extraídos da *Google Play store* e do repositório *F-Droid*. Para o processo de classificação, é feito o uso de dois classificadores automáticos. Neste projeto, utilizou-se esta base para validação, consequentemente ela não foi usada para teste e nem para treino.

4.1.1. Similaridades entre Bases de dados

Uma das principais questões relacionadas à análise de domínios cruzados está relacionada à similaridade entre conjuntos de dados. A Tabela 1 apresenta a média da similaridade entre os conjuntos de dados, considerando as trinta instâncias mais similares ao dado em análise. Através dessa tabela, espera-se que os domínios que possuem maior similaridade com os demais apresentem maior acurácia, uma vez que os textos de treinamento serão mais semelhantes aos textos de teste. O oposto também é verdadeiro, ou seja, conjuntos de dados mais dissimilares tendem a resultar em desempenho inferior.

4.2. Resultados dos experimentos propostos

A Tabela 4.2 apresenta os resultados dos experimentos realizados neste projeto, como todas as bases são balanceadas escolhemos apresentar a acurácia. Cada linha representa um *fold* diferente no qual se apresenta o desempenho obtido ao considerar uma base de teste específica, enquanto as outras bases mencionadas na tabela foram usadas para treinamento. Os modelos testados agrupam duas colunas: *"base"* e *"CFT"*.

dataset	Similaridade Média
Amazon Reviews	0.4067 ± 0.0747
Reddit Sentiment	0.4183 ± 0.0534
Rotten Tomatoes	0.4167 ± 0.0558
Silicon	0.3899 ± 0.0557
Tweet Eval	0.3968 ± 0.0471
Yelp	0.4588 ± 0.0461

Tabela 1. Comparação da similaridade média entre as diferentes bases de dados sob análise, ao se considerar os trinta dados mais similares dada uma determinada instância, cada linha representa uma base diferente.

A coluna *"base"* indica a acurácia do modelo base, quando treinado com as outras bases mencionadas, enquanto a coluna *"CFT"* mostra os resultados alcançados pelo método proposto. O melhor desempenho obtido entre as duas abordagens é destacada em negrito. Para este experimento, os modelos *"base"* foram treinados com apresentação do resultado da época que apresenta melhor desempenho na base de teste. Em relação ao *"CFT"*, apresenta-se o resultado para $k = 4$, assim gerando quatro *clusters* a partir da base de treino, e quatro modelos especialistas. Feito isso, escolhe-se a época dos modelos especialistas com melhor desempenho com base no *dataset* de validação, para extrair as *embeddings* utilizadas para treinar o modelo de fusão. Em seguida, treina-se o modelo de fusão e o resultado da época de melhor resultado na base de teste é reportado.

Pode-se destacar que o método proposto apresenta melhorias para a maioria dos casos. Em particular, é importante ressaltar os ganhos significativos observados no conjunto de dados *YELP*. Devido à sua natureza multidomínio, o uso dos modelos de fusão atingiu resultados substancialmente melhores para esse conjunto de dados.

	DistilBERT		Bert		DeBERTa-v3	
dataset	base	CFT	base	CFT	base	CFT
Amazon Reviews	0.893	**0.894**	0.740	**0.752**	**0.917**	0.916
Reddit Sentiment	0.656	**0.662**	0.651	**0.669**	0.656	**0.679**
Rotten Tomatoes	0.742	**0.754**	0.792	**0.798**	0.809	**0.814**
Silicon	0.786	**0.789**	0.764	**0.808**	0.801	**0.814**
Tweet Eval	0.793	**0.819**	0.880	**0.893**	0.837	**0.841**
Yelp	0.876	**0.906**	0.865	**0.914**	0.913	**0.942**

Tabela 2. Comparação dos modelos utilizados ao longo deste projeto. Cada linha representa o resultado para um *fold*, considerando o conjunto de dados como teste e os demais como treino. A coluna *"base"* representa a acurácia obtida pelo modelo base, enquanto as colunas *"CFT"* representam os resultados obtidos pelo método proposto. As duas colunas estão agrupadas pelo modelo utilizado, e o melhor resultado comparativo está destacado em negrito

4.3. Exemplos de resultados experimentais

A Tabela 4.3 apresenta uma amostragem de casos nos quais o modelo base comete erros, enquanto o método proposto acerta. A primeira coluna contém o texto, a segunda coluna mostra o modelo em análise, a terceira coluna representa a classe prevista pelo modelo base, a quarta coluna mostra a previsão do método proposto e a última coluna apresenta a classe correta.

Através dessa tabela, observa-se que o método proposto melhora em relação ao modelo base, principalmente em casos nos quais palavras podem ter tanto um sentimento negativo quanto positivo. Isso pode ser observado nos exemplos um e dois, nos quais a palavra *"hot"* tem um sentimento negativo na primeira ocorrência, mas um sentimento positivo no segundo caso. O mesmo pode ser afirmado em relação aos exemplos três e quatro, nos quais a palavra *"cold"* tem uma conotação negativa ao se referir à *"pizza"*, mas é positiva ao se referir à *"cerveja"*. O último caso reforça essa ideia ao mencionar a palavra *"fast"*, que apresenta um sentimento negativo nos exemplos cinco, ao se referir a uma queda rápida, enquanto no exemplo seis, a palavra *"fast"* tem uma conotação positiva ao se referir à forma como um produto funciona.

Esses exemplos apresentam evidências de que o método proposto tem mostrado melhorias ao considerar domínios específicos que modificam a polaridade do sentimento do texto em análise.

	Texto	Modelo	Base	CFT	Classe
1	ouch ! the water's too **hot** !	BERT	positivo	negativo	negativo
2	... because they 're **hot** from the oven .	BERT	negativo	positivo	positivo
3	... every topping on the pizza was **cold** ...	DeBERTa-v3	positivo	negativo	negativo
4	Grate food **cold** beer and ...	DeBERTa-v3	negativo	positivo	positivo
5	when modi retires bjp going fall hard and fall **fast**	DistilBERT	positivo	negativo	negativo
6	Works great Works amazing, very **fast** and easy to use...	DistilBERT	negativo	positivo	positivo

Tabela 3. Exemplos de casos em que o método proposto acerta e o modelo base erra. Em muitos casos o método proposto apresenta melhorias, para palavras que podem ser positivas ou negativas, dependendo do contexto.

5. Conclusão

Neste artigo é apresentado um novo método, CFT, que representa uma nova abordagem para melhorar modelos de classificação. Para avaliar método proposto, utilizou-se a tarefa de análise de sentimentos, e por meio de domínios cruzados apresenta-se como o método pode ser utilizado sobre domínios em que não há dados rotulados. É reportado a efetividade do método em seis conjunto de dados diferentes e três modelos baseados em *transformers*.

A flexibilidade do CFT permite customização e melhorias de diferentes módulos. De tal forma que, para pesquisas futuras será explorado novos métodos de *clustering*, como técnicas de agrupamento hierárquico [Murtagh and Contreras 2012]. Além disso, o método proposto também pode comportar técnicas de *few-shot learning* para treinar *LLMs* como o GPT [Brown et al. 2020]. Ademais, apesar deste trabalho focar em análise de sentimentos, CFT tem o potencial para ser utilizado em diferentes domínios e tarefas.

Em relação às limitações do método proposto, a abordagem requer a extração de *embeddings* dos múltiplos modelos especialistas, aumentando as demandas computacionais para previsão. O método apresenta uma solução promissora para aprimorar modelos de classificação em domínios nos quais dados rotulados são escassos. O uso de técnicas de *clustering* combinadas com técnicas de fusão é utilizado como recurso para melhorar o desempenho dos modelos de análise de sentimentos.

Agradecimentos

Este trabalho foi financiado pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) no processo #88887.893806/2023-00. Os autores agradecem ao Centro de Inteligência Artificial (C4AI-USP) pelo suporte a partir da agência de financiamento FAPESP no processo 2019/07665-4 e da IBM Corporation. Este trabalho foi apoiado pelo Ministério da Ciência, Tecnologia e Inovações com fundos a partir da Lei 8248, de 23 de Outubro de 1991, PPI-SOFTEX, coordenado por Softex [DOU 01245.010222/2022-44].

Referências

Adoma, A. F., Henry, N.-M., and Chen, W. (2020). Comparative analyses of bert, roberta, distilbert, and xlnet for text-based emotion recognition. In *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pages 117–121.

Ahmed, M., Seraj, R., and Islam, S. M. S. (2020). The k-means algorithm: A comprehensive survey and performance evaluation. *Electronics*, 9(8):1295.

Ain, Q. T., Ali, M., Riaz, A., Noureen, A., Kamran, M., Hayat, B., and Rehman, A. (2017). Sentiment analysis using deep learning techniques: a review. *International Journal of Advanced Computer Science and Applications*, 8(6).

Alapati, Y. K. and Sindhu, K. (2016). Combining clustering with classification: a technique to improve classification accuracy. *Lung Cancer*, 32(57):3.

Araújo, M., Pereira, A., and Benevenuto, F. (2020). A comparative study of machine translation for multilingual sentence-level sentiment analysis. *Information Sciences*, 512:1078–1102.

Asghar, N. (2016). Yelp dataset challenge: Review rating prediction. *arXiv preprint arXiv:1605.05362*.

Birjali, M., Kasri, M., and Beni-Hssane, A. (2021). A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowledge-Based Systems*, 226:107134.

Bousdekis, A., Lepenioti, K., Apostolou, D., and Mentzas, G. (2021). A review of data-driven decision-making methods for industry 4.0 maintenance applications. *Electronics*, 10(7):828.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Chapuis, E., Colombo, P., Manica, M., Labeau, M., and Clavel, C. (2020). Hierarchical pre-training for sequence labelling in spoken dialog. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2636–2648, Online. Association for Computational Linguistics.

Dang, N. C., Moreno-García, M. N., and De la Prieta, F. (2020). Sentiment analysis based on deep learning: A comparative study. *Electronics*, 9(3):483.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dos Santos, B. N., Marcacini, R. M., and Rezende, S. O. (2021). Multi-domain aspect extraction using bidirectional encoder representations from transformers. *IEEE Access*, 9:91604–91613.

Gowda, C., Anirudh, Pai, A., and kumar A, C. (2019). Twitter and reddit sentimental analysis dataset.

Grano, G., Di Sorbo, A., Mercaldo, F., Visaggio, C. A., Canfora, G., and Panichella, S. (2017). Android apps and user feedback: a dataset for software evolution and quality improvement. In *Proceedings of the 2nd ACM SIGSOFT international workshop on app market analytics*, pages 8–11.

Guo, M.-H., Xu, T.-X., Liu, J.-J., Liu, Z.-N., Jiang, P.-T., Mu, T.-J., Zhang, S.-H., Martin, R. R., Cheng, M.-M., and Hu, S.-M. (2022). Attention mechanisms in computer vision: A survey. *Computational visual media*, 8(3):331–368.

Habimana, O., Li, Y., Li, R., Gu, X., and Yu, G. (2020). Sentiment analysis using deep learning approaches: an overview. *Science China Information Sciences*, 63:1–36.

He, P., Gao, J., and Chen, W. (2021). Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

He, P., Liu, X., Gao, J., and Chen, W. (2020). Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Hinton, G., Vinyals, O., Dean, J., et al. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7).

Kaur, H., Ahsaan, S. U., Alankar, B., and Chang, V. (2021). A proposed sentiment analysis deep learning algorithm for analyzing covid-19 tweets. *Information Systems Frontiers*, pages 1–13.

Keung, P., Lu, Y., Szarvas, G., and Smith, N. A. (2020). The multilingual amazon reviews corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

Kriegel, H.-P., Schubert, E., and Zimek, A. (2017). The (black) art of runtime evaluation: Are we comparing algorithms or implementations? *Knowledge and Information Systems*, 52:341–378.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Liao, Q., Yuan, J., Dong, M., Yang, L., Fielding, R., and Lam, W. W. T. (2020). Public engagement and government responsiveness in the communications about covid-19 during the early epidemic stage in china: infodemiology study on social media data. *Journal of medical Internet research*, 22(5):e18796.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ma, T., Sun, Y., Yang, Z., and Yang, Y. (2023). Prod: Prompting-to-disentangle domain knowledge for cross-domain few-shot image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19754–19763.

Maas, A., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Murtagh, F. and Contreras, P. (2012). Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(1):86–97.

Nandwani, P. and Verma, R. (2021). A review on sentiment analysis and emotion detection from text. *Social Network Analysis and Mining*, 11(1):81.

Nassiri, K. and Akhloufi, M. (2022). Transformer models used for text-based question answering systems. *Applied Intelligence*, pages 1–34.

Niu, Z., Zhong, G., and Yu, H. (2021). A review on the attention mechanism of deep learning. *Neurocomputing*, 452:48–62.

Ortiz-Ospina, E. and Roser, M. (2023). The rise of social media. *Our world in data*.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the ACL*.

Piernik, M. and Morzy, T. (2021). A study on using data clustering for feature extraction to improve the quality of classification. *Knowledge and Information Systems*, 63(7):1771–1805.

Rosenthal, S., Farra, N., and Nakov, P. (2017). Semeval-2017 task 4: Sentiment analysis in twitter. In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, pages 502–518.

Rosenthal, S., Farra, N., and Nakov, P. (2019). Semeval-2017 task 4: Sentiment analysis in twitter. *arXiv preprint arXiv:1912.00741*.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Sarker, I. H. (2021). Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions. *SN Computer Science*, 2(6):420.

Sikhi, Y., Devi, S. A., Jasti, S. K., and Ram, M. S. (2022). Sentimental analysis through speech and text for imdb dataset. In *2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, pages 1519–1522. IEEE.

Silva, E. H. d. and Marcacini, R. M. (2021). Aspect-based sentiment analysis using bert with disentangled attention. In *Proceedings*.

Singh, M., Jakhar, A. K., and Pandey, S. (2021). Sentiment analysis on the impact of coronavirus in social life using the bert model. *Social Network Analysis and Mining*, 11(1):33.

Sirusstara, J., Alexander, N., Alfarisy, A., Achmad, S., and Sutoyo, R. (2022). Clickbait headline detection in indonesian news sites using robustly optimized bert pre-training approach (roberta). In *2022 3rd International Conference on Artificial Intelligence and Data Sciences (AiDAS)*, pages 1–6.

Sivarajah, U., Irani, Z., Gupta, S., and Mahroof, K. (2020). Role of big data and social media analytics for business to business sustainability: A participatory web context. *Industrial Marketing Management*, 86:163–179.

Song, H. and Yang, W. (2022). Gscctl: a general semi-supervised scene classification method for remote sensing images based on clustering and transfer learning. *International Journal of Remote Sensing*, 43(15-16):5976–6000.

Sun, J., Lapuschkin, S., Samek, W., Zhao, Y., Cheung, N.-M., and Binder, A. (2021). Explanation-guided training for cross-domain few-shot classification. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 7609–7616.

Tseng, H.-Y., Lee, H.-Y., Huang, J.-B., and Yang, M.-H. (2020). Cross-domain few-shot classification via learned feature-wise transformation. *arXiv preprint arXiv:2001.08735*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, W., Bao, H., Huang, S., Dong, L., and Wei, F. (2020). Minilmv2: Multi-head self-attention relation distillation for compressing pretrained transformers. *arXiv preprint arXiv:2012.15828*.

Wankhade, M., Rao, A. C. S., and Kulkarni, C. (2022). A survey on sentiment analysis methods, applications, and challenges. *Artificial Intelligence Review*, 55(7):5731–5780.

Yadav, A. and Vishwakarma, D. K. (2020). Sentiment analysis using deep learning architectures: a review. *Artificial Intelligence Review*, 53(6):4335–4385.

Zhang, L., Wang, S., and Liu, B. (2018). Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1253.

Zong, C., Xia, R., and Zhang, J. (2021). *Text Data Mining*, volume 711. Springer.