Entity Matching com *Large Language Models*: estudo comparativo com abordagem de *Entity Blocking*

Rodolfo Bolconte Donato[1], **Tiago Brasileiro Araújo**[1]

[1]Instituto Federal da Paraíba *campus* João Pessoa (IFPB-JP)
Av. Primeiro de Maio, 720 - Jaguaribe, João Pessoa - PB, 58015-435
João Pessoa, Paraíba, Brasil

rodolfo@copin.ufcg.edu.br, tiago.brasileiro@ifpb.edu.br

Abstract. *Entity Matching is essential to integrate data from different sources that refer to the same entity. Although pre-trained models that adopt Entity Blocking techniques are widely used in this task, the advancement of Large Language Models (LLMs) suggests new possibilities. This work compares Ditto, which applies optimization techniques to traditional models, with Orca2, an LLM based on Llama2 focused on reasoning. Despite its inferior initial performance, Orca2 demonstrates competitive potential, especially with future computational improvements. Thus, we seek to evaluate the feasibility of LLMs in Entity Matching, analyzing accuracy and computational cost.*

Resumo. *Entity Matching é fundamental para integrar dados de diferentes fontes que se referem à mesma entidade. Embora modelos Pré-Treinados que adotam técnicas de Entity Blocking sejam amplamente utilizados nessa tarefa, o avanço dos Large Language Models (LLMs) sugere novas possibilidades. Este trabalho compara o Ditto, que aplica técnicas de otimização com modelos tradicionais, com o Orca2, um LLM baseado no Llama2 voltado para raciocínio. Apesar do desempenho inicial inferior, o Orca2 demonstra potencial competitivo, sobretudo com futuras melhorias computacionais. Assim, busca-se avaliar a viabilidade dos LLMs em Entity Matching, analisando precisão e custo computacional.*

1. Introdução

Com o aumento da quantidade e diversidade de dados disponíveis, torna-se essencial organizá-los de forma sistemática. Uma das abordagens fundamentais nesse contexto é a resolução de entidades, que busca identificar registros distintos que se referem à mesma entidade real. No campo da Ciência e Engenharia de Dados, tal atividade é conhecida como *Entity Matching* (no português, Resolução de Entidades), que sendo um aspecto vital do Processamento de Linguagem Natural (PLN), tem como objetivo identificar e conectar informações, determinando se diferentes registros se referem à mesma entidade do mundo real [Barlaug and Gulla 2021].

Sendo um processo essencial para limpeza e junção de dados em conjuntos únicos ou distribuídos, a realização da tarefa de *Entity Matching* com precisão e rapidez tem implicações práticas em aplicações comerciais, científicas e de segurança, embora seja um problema recorrente nos contextos de integração e limpeza de dados. Tal processo visando responder se informações são de uma mesma entidade ou não pode ser demorado,

especialmente quando utilizadas fontes grandes e/ou heterogêneas de dados. Sendo assim, *Entity Matching* continua sendo uma tarefa desafiadora pois requer uma profundidade de compreensão de linguagem e conhecimento de domínio para corresponder e distinguir informações de entidades [Zhang et al. 2024].

Pesquisas recentes estudam a realização de tarefas de *Entity Matching* utilizando abordagens centradas em indexação, ou blocagem, evidenciando então a área de *Entity Blocking*, em que métodos de Aprendizagem Profunda se tornaram o padrão de fato para lidar com este tipo de tarefa. A partir de exemplos rotulados, os *Pre-trained Language Models (PLM)* se mostram como soluções centradas em indexes, ou blocos, visando a redução do número de pares de registros comparados em detalhes, removendo pares que provavelmente não correspondem a mesma entidade durante a etapa de comparação [Christen and Christen 2012].

Devido a avanços recentes no âmbito do Processamento de Linguagem Natural, foi possível o desenvolvimento dos *Large Language Models (LLMs)*, modelos treinados com grandes quantidades de informações textuais com recursos avançados na compreensão e também na própria geração de texto semelhante aos humanos [Kuang et al. 2024]. A partir de uma ampla gama de aplicações do mundo real em vários domínios como *chatbots*, assistentes de escrita, mecanismos de busca, sistemas modais, etc. se faz importante a experimentação de *LLMs* também no âmbito da *Entity Matching*, uma vez que tais modelos não precisam de quantidades significativas de exemplos de treinamento específicos para a realização de uma tarefa, sendo a principal vantagem em relação aos *PLMs* [Peeters et al. 2023b].

Os *Pre-trained Language Models* ainda possuem bastante utilização em atividades de *Entity Matching*, devido a sua capacidade de ajuste para tarefas específicas através da utilização de dados rotulados, porém, estudos descobriram sua tendência de aprender padrões incertos nos dados, o que significa que tais modelos tendem a adquirir uma compreensão superficial da tarefa em questão e são mais propensos a falhar em diferentes circunstâncias [Niven and Kao 2019]. Para contornar tais problemas inerentes ao funcionamento dos *PLMs*, é idealizada a utilização dos *Large Language Models*, em que modelos como *ChatGPT, Copilot* e *Gemini* tem demonstrado resultados promissores para abordar as deficiências dos *PLMs*, devido o seu treinamento com grandes quantidades de dados textuais, que mostram um melhor desempenho para a realização de atividades sem nenhuma ou pouca quantidade de exemplos específicos [Peeters et al. 2023b].

Considerando as vantagens associadas ao uso de *Large Language Models* e as limitações observadas em *Pre-trained Language Models* tradicionais, este trabalho propõe um estudo comparativo entre dois modelos aplicados à tarefa de *Entity Matching*: o *Ditto*, que emprega técnicas de *Entity Blocking*, e o *Orca2*, um *LLM* com foco na otimização de raciocínio via *prompting*. Ambos os modelos são avaliados sob as mesmas condições e conjuntos de dados de entidades, de modo a garantir uma comparação equitativa. O objetivo é analisar o desempenho do *Orca2* e investigar o potencial dos *LLMs* como alternativa mais eficiente para essa tarefa.

2. Trabalhos Relacionados

O trabalho de [Li et al. 2020] propôs o *Ditto*, um sistema baseado em *Entity Blocking* para tarefas de *Entity Matching*, otimizado com três estratégias principais: uso de conhe-

cimento de domínio, resumo de sequências e aumento de dados. Essas técnicas visam melhorar a capacidade do modelo de identificar correspondências relevantes entre entidades. Segundo os autores, o *Ditto* obteve até 29 pontos percentuais de ganho em *F1-Score* em relação a abordagens anteriores, resultado também validado por estudos como [Brasileiro Araújo et al. 2025] e [Peeters et al. 2023a].

O trabalho de [Wang and Yan 2024] apresenta a abordagem *UDAEB* como exemplo do uso de *LLMs* em tarefas de *Entity Matching*, porém apenas para a etapa de aumento de dados e não como o modelo principal de decisão. A proposta envolve três etapas: representação de amostras em espaços de similaridade (aquecimento), geração de atributos sintéticos com o *Mistral-7B-Instruct-v0.2* (enriquecimento), e aplicação de um modelo pré-treinado para a resolução das entidades (iteração). Os resultados mostram ganhos expressivos sobre técnicas tradicionais de *entity blocking*, sendo 50,9 e 48,3 pontos percentuais maior que abordagens de *entity blocking* consolidadas na literatura para *Recall* e *Precision*, respectivamente.

O trabalho de [Arvanitis-Kasinikos and Papadakis 2025] investiga o uso direto de *LLMs* na tarefa de *Entity Matching*, com foco em modelos leves de 7 bilhões de parâmetros. Utilizando os conjuntos de dados *Abt-Buy* e *Walmart-Amazon*, os autores avaliam seis modelos distintos sob duas estratégias de *prompting* — *ZeroShot* e *FewShot* — com o objetivo de verificar a capacidade desses modelos em reconhecer correspondências entre entidades. Entre os avaliados, o *Orca2* se destacou, alcançando quase 80% de *F1-Score* em ambas as abordagens. Esses resultados demonstram que, mesmo com limitações de custo computacional, alguns *LLMs* conseguem desempenho competitivo na tarefa, especialmente quando bem orientados por estratégias de *prompting*.

3. Metodologia

A ideia do presente trabalho é um estudo comparativo de duas diferentes abordagens para a realização de uma tarefa de *entity matching*, a fim de verificar o desempenho classificatório de um *Large Language Model* na resolução de entidades referente a produtos eletrônicos, em contraste com um modelo que utiliza técnicas de blocagem de informações para realizar a mesma tarefa. Para um melhor entendimento, o presente estudo possui três etapas de execução: o carregamento do conjunto de dados, a execução dos modelos para realização da tarefa de *entity matching* e uma análise dos resultados obtidos nas execuções. Todas as etapas executadas, bem como informações técnicas de dados, modelos e métricas são detalhadas nas próximas subseções.

3.1. Conjunto de Dados e Carregamento

Devido a grande aderência do conjunto de dados *Abt-Buy* para tarefas de resolução de entidades na literatura, sua utilização é preferida para o presente trabalho. O *Abt-Buy* se trata de um conjunto de produtos eletrônicos de duas empresas de vendas online, a *abt.com* e a *buy.com*, contendo três informações de cada produto: nome, descrição e preço.

Com relação ao tamanho do *Abt-Buy*, ele possui nativamente 1028 tuplas que representam a mesma entidade de produto eletrônico, ou seja, duas informações de produtos que descrevem o mesmo, e 8547 tuplas que descrevem produtos diferentes. O conjunto de dados é dividido em subconjuntos de treino, validação e teste: 5127 tuplas negativas e

616 positivas como dados de treino, 1710 positivas e 206 negativas para validação e estas mesmas quantidades para o subconjunto de teste também. Para a utilização de tais subconjuntos, estes são carregados e organizados utilizando a linguagem *Python* juntamente com o pacote *Pandas*.

3.2. Modelos e Execuções

Com a comparação de duas abordagens para a realização da tarefa de *entity matching*, são executados dois modelos, o primeiro sendo o *Ditto* como uma abordagem que utiliza técnica de blocagem de informações e o segundo modelo é o *Orca2*, como sendo o *Large Language Model* a ser comparado e seu desempenho analisado em relação à realização da atividade de resolução de entidades.

Comentado anteriormente na Seção 2, o *Ditto* possui três otimizações visando a melhora de resultados: conhecimento de domínio, resumo de sequências longas e aumento de dados. Além destas otimizações, é possível citar a utilização dos modelos *BERT* de 12 camadas, *RoBERTa* e *DistilBERT* de 6 camadas, utilizando conjuntos de dados voltados para *entity matching* com diversas amostras positivas e negativas nas etapas de autoajuste [Li et al. 2020]. Tais fatores relacionados a construção e funcionamento do *Ditto* se mostraram determinantes para a escolha e utilização do modelo.

O *Large Language Model* utilizado no presente trabalho é *Orca2* com 7 bilhões de parâmetros, que, sendo desenvolvido e gerenciado pela Microsoft para fins de pesquisa, se trata de uma versão refinada do modelo *Llama2* voltada para uma otimização de raciocínio e não de conversação, sendo indicado para tarefas específicas e não multitarefas [Mitra et al. 2023]. Como discutido na Seção 2, o *Orca2* obteve os melhores resultados para a atividade de *entity matching* no trabalho realizado em [Arvanitis-Kasinikos and Papadakis 2025], justificando a preferência por sua utilização.

Para a criação e execução de uma instância do *Orca2*, é utilizado o *framework Ollama* em conjunto com a linguagem *Python*, proporcionando uma implementação simplificada para as etapas de criação e execução do modelo. Com um *prompt* do *Ollama*, é possível realizar a construção de uma instância do *Orca2*, especificando que o mesmo apenas receberá informações de entidades, e em que sua saída deve ser apenas um valor específico de acordo com seu entendimento, na Figura 1 é possível conferir o *prompt* utilizado para a criação do modelo.

```
SYSTEM You are a crowdsourcing worker, working on an entity resolution task. You will be
given two record descriptions and your task is to identify if the records refer to the
same entity or not. You must answer with just one word: True. if the records are referring
to the same entity, False. if the records are referring to a different entity.

MESSAGE user record 1: Sony MDREX35LP VB Colorful Headphone with Case - Violet
BlueMDREX35LPVB13.540.05Sony7.25 x 2.0 x 1.25 inches record 2: Sony MDR-EX35LP VB EX Style
Headphones with Deep Bass Sound Violet BlueMDR-EX35LPVB12.991Sony7.2 x 2.0 x 1.2 inches

MESSAGE assistant True.

MESSAGE user record 1: Sony MDREX35LP VB Colorful Headphone with Case - Violet
BlueMDREX35LPVB13.540.05Sony7.25 x 2.0 x 1.25 inches record 2: Sony MDRJ10 LTPNK Clip
Style Headphones PinkMDRJ10LTPNK9.061Sony7.2 x 4.0 x 1.5 inches

MESSAGE assistant False.
```

Figura 1. *Prompt* utilizado na criação do modelo *Orca2*.

3.3. Análise de Resultados

Após a execução dos modelos, tanto do *Ditto* quanto do *Orca2*, é necessário analisar seus resultados de forma estatística para validar o desempenho de ambos. Apesar de ser uma tarefa de *entity matching*, logo reconhecer se as duas informações são referentes a um mesmo produto, ainda se trata de uma tarefa de classificação binária de informações, em que apenas dois valores são possíveis no seu resultado e comparado com uma resposta já existente. Sendo assim, é possível a utilização de métricas já consolidadas na literatura para tarefas de classificação binária, sendo elas: *Precision*, *Recall* e *F1-Score*.

4. Experimentos e Resultados

Antes da discussão dos resultados, é importante destacar que foram realizadas 10 execuções para cada um dos modelos avaliados. As métricas apresentadas correspondem à média dessas execuções, com o objetivo de proporcionar uma análise mais robusta e generalizável dos desempenhos observados.

Para que seja possível utilizar o *Ditto* na tarefa de resolução de entidades, antes é preciso realizar a construção de uma instância do modelo através de uma etapa de treino utilizando parte do conjunto de dados, que como descrito na Seção 3.1, são utilizadas as 5743 tuplas do subconjunto de treino, com 5127 tuplas negativas e 616 positivas, levando em torno de 18 minutos para a realização desta etapa ao executar 40 épocas de treino. Durante a realização do treino, o modelo é validado com o subconjunto de validação a cada época, atingindo um *F1-Score* máximo de 80% na segunda época.

Utilizando o conjunto de testes com 1916 tuplas, sendo 1710 negativas e 206 positivas, o *Ditto* é capaz de atingir 79% de *Precision*, 84% de *Recall* e 81% de *F1-Score*, sendo estes valores levando em consideração as duas classes de resultados no cálculo, negativa e positiva. Ao verificar os valores métricos para cada uma das classificações possíveis, o *Ditto* alcança valores elevados para as amostras negativas, sendo 97%, 94% e 95% para *Precision*, *Recall* e *F1-Score*, respectivamente, enquanto que para as amostras positivas atinge 61%, 73% e 66% para as mesmas métricas. É importante apresentar também que toda a execução do *Ditto* para o conjunto de dados de teste levou cerca de 30 segundos para ser realizada.

Para a execução do *Orca2*, como não há uma etapa de treino do modelo, é utilizado apenas o conjunto de dados de teste do *Abt-Buy*. Sendo assim, após a execução da etapa de teste, o *Orca2* consegue atingir 71%, 61% e 64% para *Precision*, *Recall* e *F1-Score*, respectivamente, levando em consideração as amostras das duas classificações possíveis, que embora não sejam valores tão próximos do *Ditto*, se mostram promissores atingindo mais que 50%. Ao analisar os valores métricos para as amostras de cada uma das classificações possíveis, o *Orca2*, assim como o *Ditto*, consegue valores satisfatórios para a classe negativa, com 92% de *Precision*, 97% de *Recall* e 94% de *F1-Score*, porém para a classe positiva, o desempenho é significativamente inferior ao observado no *Ditto*, atingindo apenas 50%, 25% e 34% para *Precision*, *Recall* e *F1-Score*, respectivamente, evidenciando um alto número de tuplas positivas que são classificadas como negativas. O tempo de execução do *Orca2* é outro fator discrepante em relação ao *Ditto*, em que o modelo realiza a tarefa de resolução de entidades para a mesma quantidade de dados em cerca de 15 minutos e 34 segundos, sendo um tempo de execução 29 vezes maior que o tempo de execução do *Ditto*.

Na Tabela 1 é possível visualizar os valores obtidos para cada uma das métricas separadas por modelo e amostras, sejam somente negativas, positivas ou todas em conjunto.

Tabela 1. Resultados das classificações executadas pelo *Ditto* e *Orca2*.

Modelo	Amostras	*Precision* (%)	*Recall* (%)	*F1-Score* (%)
Ditto	Negativas	97	94	95
	Positivas	61	73	66
	Todas	79	84	81
Orca2	Negativas	92	97	94
	Positivas	50	25	34
	Todas	71	61	64

Com os resultados obtidos no presente estudo, é evidente o desempenho inferior do *Orca2* em relação ao *Ditto*, tanto na resolução de entidades quanto no tempo de execução. Apesar de apresentar bons resultados para amostras negativas, o *Orca2* mostrou limitações na identificação de tuplas positivas e exigiu um tempo significativamente maior para obter resultados menos precisos[1].

5. Considerações Finais

Os resultados experimentais evidenciam dois caminhos contrastantes na realização da resolução de entidades: enquanto o *Ditto* apresenta desempenho superior em métricas globais e tempo de execução, o *Orca2* demonstra a viabilidade dos *LLMs* como alternativa promissora mesmo sem treinamento supervisionado. Sua elevada performance na identificação de tuplas negativas, aliada à simplicidade de uso – possibilitando sua aplicação em cenários de poucas amostras –, reforça o potencial dessas arquiteturas em cenários onde adaptabilidade e escalabilidade podem ser mais relevantes do que os resultados métricos de classificação.

Embora o *Orca2* seja o representante dos *Large Language Models* no presente estudo comparativo, não é possível afirmar que os *LLMs* não sejam adequados para a realização da atividade de *Entity Matching*, pelo contrário, é necessário em estudos futuros a investigação dos casos de acerto que o modelo foi capaz de obter, a fim de verificar toda a lógica por trás dos resultados e confirmar se há abertura para novas execuções e melhora do modelo ou não. Outras possibilidades de investigações relacionadas ao *Orca2* seriam: 1) testes de melhorias de desempenho, como: utilização de mais amostras positivas na criação do modelo, conjunto de dados com quantidade balanceada de amostras, diversificação de *prompts*, etc.; e 2) utilização do *Orca2* apenas em execuções de amostras que gerem conflitos em outros modelos, cabendo a este, a classificação final do dado.

Como última consideração, o presente trabalho abre espaço para a realização de novos estudos adotando outros *Large Language Models* na tarefa de *Entity Matching*. Devido a grande quantidade e concorrência dos *LLM* na literatura, é interessante a realização de estudos com modelos amplamente consolidados, como o *ChatGPT*, *Copilot* e *Gemini*, para verificar o quão bons estes podem vir a ser em tarefas de resolução de entidades.

[1]Informações das execuções realizadas podem ser encontradas em:
https://github.com/rodolfobolconte/SBBD25-paper-em-llm

Referências

Arvanitis-Kasinikos, I. and Papadakis, G. (2025). Entity matching with 7b llms: A study on prompting strategies and hardware limitations. *CEUR Workshop Proceedings*.

Barlaug, N. and Gulla, J. A. (2021). Neural networks for entity matching: A survey. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 15(3):1–37.

Brasileiro Araújo, T., Efthymiou, V., Christophides, V., Pitoura, E., and Stefanidis, K. (2025). Treats: Fairness-aware entity resolution over streaming data. *Information Systems*, 129:102506.

Christen, P. and Christen, P. (2012). Data matching systems. *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*, pages 229–242.

Kuang, W., Qian, B., Li, Z., Chen, D., Gao, D., Pan, X., Xie, Y., Li, Y., Ding, B., and Zhou, J. (2024). Federatedscope-llm: A comprehensive package for fine-tuning large language models in federated learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5260–5271.

Li, Y., Li, J., Suhara, Y., Doan, A., and Tan, W.-C. (2020). Deep entity matching with pre-trained language models. *Proceedings of the VLDB Endowment*, 14(1):50–60.

Mitra, A., Del Corro, L., Mahajan, S., Codas, A., Simoes, C., Agarwal, S., Chen, X., Razdaibiedina, A., Jones, E., Aggarwal, K., et al. (2023). Orca 2: Teaching small language models how to reason. *arXiv preprint arXiv:2311.11045*.

Niven, T. and Kao, H.-Y. (2019). Probing neural network comprehension of natural language arguments. *arXiv preprint arXiv:1907.07355*.

Peeters, R., Der, R. C., and Bizer, C. (2023a). Wdc products: A multi-dimensional entity matching benchmark. *arXiv preprint arXiv:2301.09521*.

Peeters, R., Steiner, A., and Bizer, C. (2023b). Entity matching using large language models. *arXiv preprint arXiv:2310.11244*.

Wang, Y. and Yan, M. (2024). Unsupervised domain adaptation for entity blocking leveraging large language models. In *2024 IEEE International Conference on Big Data (BigData)*, pages 159–164. IEEE.

Zhang, J., Sun, H., and Ho, J. C. (2024). Emba: Entity matching using multi-task learning of bert with attention-over-attention. In *EDBT*, pages 281–293.