An Ultra-Fine Entity Typing Method Based on Ensemble of Large Language Models

Carlos Alberto F. Ferreira[1], Anderson A. Ferreira[1]

Universidade Federal de Ouro Preto, Brazil
caffp@hotmail.com, anderson.ferreira@ufop.edu.br

Abstract. Ultra-fine Entity Typing (UFET) is an evolution of Named Entity Recognition (NER) that proposes the classification of entities at an ultra-fine level, based on a set of free-form phrases that adequately and comprehensively describe them. Increasing the number of labels while maintaining performance remains a challenge, and most current models are capable of classifying entities into a limited set of types. The proven efficiency of Large Language Models (LLMs) in improving performance in Natural Language Processing tasks presents an opportunity to enhance UFET results. In this sense, this paper proposes an ensemble method of fine-tuned public LLMs aiming to maximize the efficiency of entity classification at the ultra-fine level and expand the set of labels. Experiments show the effectiveness of the proposed method, which outperformed baselines metrics in almost all scenarios and improved the set of types.

CCS Concepts: • **Computing methodologies** → **Learning paradigms**; **Machine learning algorithms**.

Keywords: fine tuning, machine learning, NER, prompt learning

1. INTRODUÇÃO

Reconhecimento de Entidade Nomeada (NER) é uma das diversas tarefas de Processamento de Linguagem Natural (NLP) e desempenha papel fundamental como uma etapa de pré-processamento para uma variedade de aplicações. Destacam-se, por exemplo, resolução de correferência, recuperação de informação e sistemas de perguntas e respostas [Yadav and Bethard 2018]. Introduzida na 6th Message Understanding Conference (MUC) por Grishman and Sundheim [1996], seu objetivo é identificar e classificar menções de entidades que aparecem em textos a partir de categorias pré-definidas [Jehangir et al. 2023]. Assim, dado um fragmento de texto que contém uma menção de entidade, o objetivo é gerar rótulos que descrevam devidamente a menção. Considere, por exemplo, a sentença "Freddie Mercury ganha filme sobre sua carreira". Nela, o objetivo da tarefa é classificar de maneira correta e ampla a menção destacada que, neste exemplo, poderia ser descrita por uma variedade de rótulos diferentes como pessoa, cantor ou artista.

Ao longo dos anos, diversas abordagens foram propostas para essa tarefa [Jehangir et al. 2023; Yadav and Bethard 2018]. Entre elas, há: abordagens baseadas em regras, em aprendizado não supervisionado e em aprendizado supervisionado, incluindo aprendizado profundo. Outra forma de classificar a NER é com relação à granularidade das categorias de rótulos [Choi et al. 2018]. Neste caso, há trabalhos que propuseram a classificação a partir de rótulos genéricos (*coarse-grained types*) como pessoa, organização ou localização, passando por trabalhos que refinaram a granularidade dos tipos para um nível um pouco abrangente em termos de quantidade de rótulos (*fine-grained types*) e, finalmente, trabalhos que expandiram ainda mais o conjunto de rótulos para uma categoria mais refinada, irrestrita e sensível ao contexto, chamado ultrafino (detetive, filme, cidade, esporte, etc.).

O nível de granularidade dos rótulos gerados impacta diretamente na qualidade do resultado e nas

possibilidades de aplicação. Enquanto rótulos mais genéricos podem ser suficientes para uma gama de aplicações, em outros casos, respostas mais refinadas são necessárias, sobretudo que levem em consideração o contexto das sentenças. Deste modo, enquanto pesquisas tradicionais focavam em um número limitado de tipos, a tendência de modelos recentes é o esforço por níveis de granularidade mais refinados [Qian et al. 2021]. Neste sentido, na tentativa de avançar ainda mais em relação a cobertura de tipos e captura do contexto, Choi et al. [2018] introduziram em seu trabalho a *Ultra-fine Entity Typing* (UFET), como uma nova tarefa de previsão de conjuntos de frases livres para classificar as menções de entidade, no nível ultrafino.

Apesar dos avanços alcançados nos últimos tempos, expandir a capacidade de previsão dos modelos, em equilíbrio com a eficiência, segurança e custo, ainda é um desafio. Neste sentido, os modelos de linguagem de larga escala (LLM's) têm despontado como uma ferramenta chave. Autores têm explorado o aprendizado por transferência de conhecimento ao aplicar os LLM's para execução de tarefas secundárias (*downstream tasks*). Devido sua eficiência e flexibilidade, os modelos de linguagem têm revolucionado o estudo de PLN [Liu et al. 2023].

A disseminação das LLM's tem permitido o surgimento de uma nova abordagem, o *Fine Tuning*, caracterizada pela adaptação dos parâmetros dos LLM's a partir de um novo treinamento, delineado especificamente para cada tarefa subjacente, utilizando-se ou não novos dados de treinamento [Liu et al. 2023]. Durante o novo treinamento todos os parâmetros do modelo podem ser ajustados, mas devido ao alto custo envolvido, é usual optar-se pelo ajuste de apenas parte dos parâmetros, ou utilizar adaptadores que adicionam camadas à arquitetura pré-existente e realiza o ajuste apenas sobre estas camadas adicionais. Modelos de aprendizado supervisionado tradicionais usam exemplos de treinamento (x, y), sendo x características de uma instância e y seu rótulo correspondente [Liu et al. 2023]. Na aplicação do ajuste fino, uma saída y é obtida de maneira indireta a partir da entrada x. Nele, a entrada x é modificada a partir de *templates*, sendo transformada em uma entrada textual secundária $x^{'}$, que é preenchida pelo LLM, de onde são derivadas as predições y. Tais gabaritos são comumente chamados de *prompts* e, assim, o processo de ajuste fino (*fine tuning*) é também conhecido como *Prompt Tuning* ou *Prompt Learning* [Liu et al. 2023].

Assim como em [Choi et al. 2018], o objetivo deste trabalho é construir um método que, ao receber um texto que contém uma entidade nomeada, seja capaz de predizer um conjunto de classificações de forma livre para descrevê-la, maximizando a cobertura dos rótulos. Formalmente, similar à definição utilizada por Ding et al. [2022], dado um conjunto $D = \{x_1, ..., x_n\}$ de n sentenças em que cada sentença x possui uma menção m indicada, o objetivo é gerar um conjunto P de rótulos ou frases livres capaz de classificar cada menção m. Para isso, propõe -se um método para a UFET baseado na combinação de resultados, obtidos após ajuste fino, de diferentes LLM's disponibilizadas publicamente. Ressalta-se que este trabalho não encontra a entidade nomeada no texto. O método proposto recebe o texto e a entidade nomeada já encontrada e, então, retorna as possíveis classes dessa entidade. Os resultados observados superam as referências da área e demonstram a efetividade do método em melhorar o desempenho das previsões ao passo que também expande a granularidade de rótulos.

O restante deste texto está organizado da seguinte forma: a Seção 2 apresenta os trabalhos relacionados à UFET, a Seção 3 descreve o método proposto, seguida da Seção 4 com o relato dos experimentos e da Seção 5 com a conclusão.

2. TRABALHOS RELACIONADOS

Nesta seção, são descritos trabalhos relacionados à tarefa de reconhecimento ultrafino de entidades. A expansão das bases de treinamento e o aprimoramento dos modelos de previsão são abordagens comuns, descritas na literatura, utilizadas para tentar melhorar os resultados na tarefa. Nesta revisão, foram selecionados trabalhos que buscaram a expansão das bases de treinamento ou que propuseram modelos avaliados na base de dados ultrafina, *Open Entity*, proposta por Choi et al. [2018].

Expandir o tamanho das bases de treinamento, a partir da maior cobertura dos tipos de entidade é estratégia eficaz de aprimorar os modelos que trabalham com NER [Li et al. 2020]. Ling and Weld [2012] iniciaram o desenvolvimento de um corpus maior com 112 rótulos, extraídos do Freebase usando Campos Aleatórios Condicionais (CRF's) e supervisão distante. Posteriormente, adaptaram um modelo Perceptron para classificação multiclasse e multirrótulo. Yosef et al. [2012] introduziu o modelo HYENA para classificação multirrótulo em granularidade fina, usando um classificador SVM para categorizar menções de entidades em 505 rótulos, derivados de cinco classes principais e 500 subclasses baseadas na YAGO, Wikipedia e WordNet. Gillick et al. [2016] expandiram o trabalho de Ling e Weld para gerar uma base de dados para tipos contextuais finos, rotulando manualmente 12.017 menções usando supervisão distante com entidades do Wikipedia mapeadas no Freebase e aplicaram heurísticas para refinar os rótulos, utilizando dois classificadores: um local de regressão logística e outro global de Softmax. Mai et al. [2018] definiram a FG-NER (Fine Grained - NER), a partir do trabalho de Sekine [2008] rotulando 20 mil sentenças e 200 tipos de entidades em inglês e japonês. Na classificação aplicam a arquitetura LSTM + CNN combinada com CRF.

Choi et al. [2018] propuseram um novo *dataset* com aproximadamente 6.000 exemplos anotados manualmente e 2.300 tipos únicos. Esse *dataset* é classificado como ultrafino devido à sua quantidade de rótulos, comparado a *datasets* anteriores, como, por exemplo, FIGER [Ling and Weld 2012] e OntoNotes[Gillick et al. 2016]. Nestes, 7 e 4 rótulos, respectivamente, cobrem 80% dos dados, enquanto na nova base, são necessários 429 rótulos para tal cobertura. Além disso, os autores apresentam uma arquitetura baseada em atenção, combinando CNN e Bi-LSTM, capaz de predizer frases de formato livre para classificar as menções de entidade, incluindo menções pronominais e expressões nominais. Ao definirem a UFET e disponibilizarem uma nova base de dados, os autores estabeleceram-se como referência para a tarefa.

Dai et al. [2021] utilizaram LLM BERT [Devlin et al. 2019] e modelagem de linguagem mascarada para gerar mais dados de treinamento, através de hipernônimos e padrão de Hearst [Hearst 1992]. Para classificação, submetem consultas ao LLM a partir de *prompts* e adicionam uma camada de classificação linear para computar as probabilidades dos *tokens* de saída. A probabilidade é dada pela sigmóide da multiplicação do *token* de saída por uma matriz de pesos treinável. Assim, os autores treinam a matriz de pesos em duas etapas, uma com a base de dados gerada pelo LLM e uma rodada de auto-treinamento com uma mistura da base gerada pelo LLM e uma base manualmente anotada por eles com rótulos ultrafinos.

Diversos autores têm modelado o problema a partir de outros paradigmas e avaliando-os sobre a base de dados *Open Entity* . Em seu trabalho, Xiong et al. [2019] promoveram uma adaptação da arquitetura proposta por Choi et al. [2018], acrescentando uma camada de propagação de grafos que computa estatísticas de correlação e similaridade de palavras sobre os rótulos. O objetivo é eliminar tipologias contraditórias. Considerando-se a métrica *F1-score*, os modelos dos autores superaram o seu *baseline* [Choi et al. 2018] em $4,9$ pontos percentuais.

Wang et al. [2020] investigaram o efeito de diferentes modelos de *embedding* sobre o desempenho da NER substituindo, na arquitetura proposta por [Choi et al. 2018], o Glove por 7 diferentes tipos de modelos de *embeddings* pré-treinados. Os experimentos foram conduzidos na base *Open Entity* e OntoNotes, tendo o BERT alcançado o melhor resultado na base UFET, superando em $2,6$ pontos percentuais na métrica F1 o original proposto por Choi et al. [2018] na base de teste. Li et al. [2023] formularam a UFET como um problema de Inferência de Linguagem Natural (NLI), chamado de LITE. Os autores utilizam o texto de entrada e os candidatos a tipo de entidade para formular a previsão final dos tipos de entidades como uma tarefa de NLI e avaliaram sua abordagem com diferentes formas de supervisão e aninhados a diferentes modelos para rotulagem.

Feng et al. [2023] usam um modelo *seq2seq* para mapear menções de entidades (entrada) em um conjunto de tipos semânticos ultrafinos (saída), a partir do T5-large [Raffel et al. 2020], afinando os resultados com a base UFET e aplicam a busca em feixe (*beam search*) para mapear autoregressiva-

mente os conjuntos de rótulos candidatos. Utilizam um modelo de calibração para selecionar os tipos mais prováveis a partir de um limiar pré-definido. O trabalho dos autores obteve o melhor resultado, em termos de F1, e apresentou ganhos em termos de velocidade de execução comparada a propostas anteriores.

Ding et al. [2022] fizeram uma análise comparativa entre o *fine tuning* tradicional e diferentes modelos de *prompt tuning*. Eles utilizaram o BERT [Devlin et al. 2019] e, na base *Open Entity*, todos os modelos de *prompt tuning* superaram os resultados de Choi et al. [2018], por meio da estratégia de *soft prompt*, em que eles adicionam *tokens* extras ao *prompt* para que a LLM possa preencher com estruturas de ligação não contempladas no *prompt* fixo.

O método proposto neste artigo difere do primeiro grupo de autores, pois não se propõe a criação ou expansão de uma nova base de dados. Então, inserida no segundo grupo, se assemelha ao trabalho de Ding et al. [2022] ao aplicar a transferência de aprendizado, diferindo-se do mesmo ao propor um método que combina o resultado de diferentes LLM's para obter a melhor classificação final.

3. MÉTODO PROPOSTO

3.1 Arquitetura da proposta

O presente método classifica entidades no nível ultrafino a partir do *ensemble* de diferentes modelos de linguagem. Nele, os LLM's são ajustados individualmente para a UFET e são ordenados seguindo o valor da métrica de desempenho, função de perda, obtida após o ajuste dos parâmetros para esta tarefa. A ideia é usar inicialmente a LLM com o melhor desempenho durante o ajuste fino para classificar uma entidade e, caso o seu resultado, a lista de classes/categorias, seja vazia, usa-se a segunda LLM na ordenação e assim sucessivamente até ter uma lista de classes não vazia ou verificar todos os LLMs.

O método proposto é baseado em *Prompt Tuning*, em que as entradas são formatadas num *template* e fornecidas aos LLM's, para que estes se especializem na tarefa desejada. O treinamento ocorreu de forma supervisionada e utilizou técnicas de quantização junto ao adaptador LoRA [Hu et al. 2021], que adiciona parâmetros adaptáveis de baixo *rank* às camadas de atenção do modelo original, evitando a necessidade de retreinar todo o modelo, racionalizando o uso de recursos computacionais. Foram selecionados como alvos, os módulos q_proj e v_proj do adaptador, com objetivo de ajustar os LLM's para o formato de *prompt* e resultados esperados. Diferente de Ding et al. [2022], visando incentivar os LLM's preverem frases livres para descrever as entidades, o modelo proposto não limita os rótulos possíveis para os LLM's e não utiliza verbalizadores para expandir o conjunto de classes. A seguir são descritas cada uma das etapas do método proposto.

3.2 Projeto do prompt

O método proposto recebe uma sentença s com uma menção de entidade m destacada e a transforma em uma entrada secundária x' composta por um *prompt* fixo onde s e m são combinados com intruções contextuais. Em seguida, o *prompt* é alimentado nos LLM's, que retornam como saída preliminar y' o mesmo texto do *prompt* com os colchetes preenchidos com os tipos de entidade previstos para a menção. O modelo de *prompt* utilizado é mostrado na Fig 1, que ilustra um exemplo de preenchimento dentro do fluxo da tarefa completo. Os LLM's utilizados foram treinados em bases de dados da lingua inglesa, o que justifica a redação do *prompt* no mesmo idioma.

3.3 Pós-processamento

Ao submeter uma consulta a partir do *prompt*, os LLM's retornam uma saída y' textual idêntica ao texto de entrada, com os colchetes preenchidos com a classificação das entidades. A resposta final do

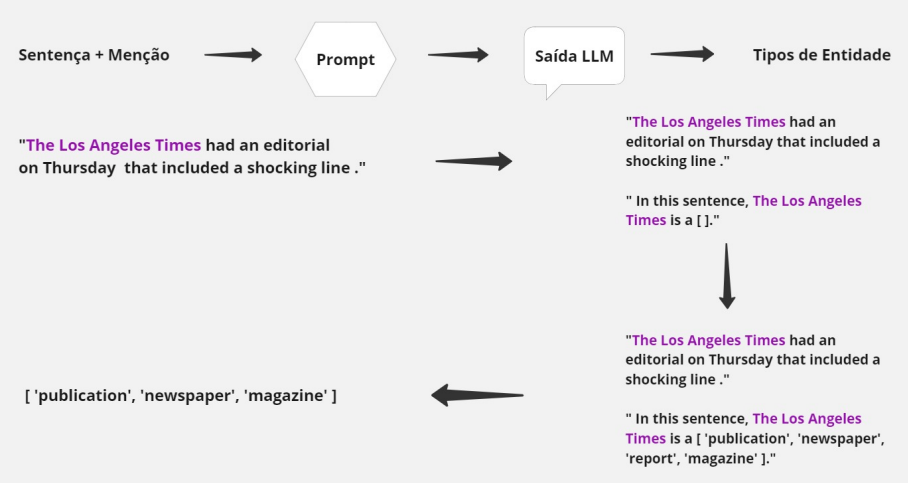

Fig. 1. Fluxo da tarefa NER com *prompt learning*

modelo, y, é derivada de y' extraindo-se o conteúdo localizado entre os colchetes.

3.4 Estratégia de consulta às LLMs

Após o ajuste fino, uma consulta é realizada a cada LLM, seguindo a ordenação definida durante o ajuste fino. Ou seja, usa-se o modelo que teve a melhor performance durante o treinamento e, caso sua resposta seja vazia, usa-se a saída do modelo com segunda melhor performance, e assim, suscessivamente, até que todos os LLM's tenham sido consultados ou obtenha um resultado não vazio. O objetivo é garantir que a consulta tenha uma saída não vazia, priorizando os modelos que tiveram melhores resultados, individualmente. A submissão a cada LLM é ilustrada na Fig. 1.

4. AVALIAÇÃO EXPERIMENTAL

4.1 Configuração dos Experimentos

Conjunto de Dados. O conjunto de dados *Open Entity* foi utilizado nos experimentos. Ele foi o único conjunto encontrado atendendo aos critérios do nível ultrafino. O conjunto possui aproximadamente 6.000 exemplos rotulados, divididos em três categorias diferentes para avaliar o desempenho dos modelos preditivos em cada nível. Ele possui 9 tipos gerais, 121 tipos finos e 10.201 tipos ultra-finos. O conjunto é dividido em treinamento, desenvolvimento e teste, com 1998 instâncias cada. Nele, cada exemplo contém uma sentença com um menção de entidade destacada e seus respectivos rótulos. Em média, cada instância possui 5 rótulos e, dentre aqueles classificados como ultrafinos, foram encontrados 2.300 tipos distintos. Neste trabalho foram utilizados apenas os dados pertencentes a parcela *crowd* do *dataset*.

LLM's Avaliadas. Foram utilizadas as menores versões dos modelos de linguagens Qwen [Bai et al. 2023], Llama 3 [AI@Meta 2024], Mistral [Jiang et al. 2023] e a versão 7B do Gemma [Gemma Team et al. 2024] (para efeito de comparação). Visando a otimização dos resultados, em cada experimento foi aplicado o *tokenizador* correspondente de cada modelo. A Tabela I fornece a identificação dos modelos e suas respectivas quantidades de parâmetros totais e treináveis nos experimentos e, abaixo, há uma descrição detalhada sobre cada deles.

O ajuste de parâmetros foi feito em uma época, usando o conjunto de exemplos de treinamento fornecido no conjunto de dados *crowd*, e batch de 2. Após o ajuste, usou-se, como dito anteriormente,

Modelo	Id	Total de Parâmetros	Parâmetros Treináveis
Gemma	google/gemma-7b	7B	0,78B
Qwen	Qwen/Qwen1.5-7B-Chat	7B	1,24B
Llama 3	meta-llama/Meta-Llama-3-8B	8B	1,05B
Mistral	mistralai/Mistral-7B-v0.3	7B	0,2B

Table I. Caracterização dos Modelos de Linguagem

Modelo	Precisão	Recall	F1-Score
Gemma	41,9	39,1	**40,5**
Llama 3	**54,3**	20,4	29,6
Qwen	26,3	**43,4**	32,7
Mistral	43,9	32,9	37,6

Table II. Resultados individuais dos LLM's sobre o conjunto de exemplos de teste.

Modelo	Precisão	Recall	F1-Score
Mistral	43,9	32,9	37,6
Mistral + Gemma	53,1	42	46,9
Mistral + Gemma + Llama 3	53,8	42,4	47,4
Mistral + Gemma + Llama 3 + Qwen	**54,1**	**42,7**	**47,7**

Table III. Resultados por *ensemble* avaliados no método.

a métrica de desempenho para definir a prioridade de uso dos LLM's. Assim, a ordem de utilização dos LLM's é Mistral, Gemma, Llama 3 e Qwen.

Métricas de Avaliação. Seguindo a escolha dos *baselines* [Ling and Weld 2012], os resultados foram avaliados a partir das métricas macro *precision*, *recall* e $F1$.

Baselines. Para avaliar, foram comparados os resultados da proposta aos relatados nos trabalhos [Choi et al. 2018] e [Ding et al. 2022]. Estes trabalhos estão descritos na Seção 2.

4.2 Análise dos resultados

Inicialmente, cada LLM teve seus parâmetros ajustados, retornando a função de perda de cada um. Neste momento, optou-se por verificar o desempenho isolado de cada LLM sobre os dados de teste. A Tabela II traz esses resultados. Ressalta-se que aplicou-se o teste estatístico Wilcoxon nestes resultados com 95% de confiança. Os resultados de precisão obtidos mostram modelos Llama e Mistral foram os mais precisos, estando estatísticamente empatados. Quanto ao *recall*, o modelo Qwen teve o melhor resultado dentre todos, seguido do Gemma e Mistral. Finalmente, com relação ao F1-score, Gemma forneceu o melhor resultado.

Qwen foi o modelo capaz de gerar o maior número de rótulos únicos na base de teste, com 2210. Em seguida vêm Gemma, Mistral e Llama 3, com 1777, 1455 e 932, respectivamente. De modo geral, todos os métodos conseguiram cumprir a totalidade ou quase nos níveis geral e fino e foram capazes de classificar um alto número de tipos no ultra-fino, com Qwen, Mistral, Gemma e Llama 3 tendo, respectivamente 1301, 945, 913 e 588 rótulos em comum com a base *Open Entity*. Os modelos geraram um número de etiquetas relativamente elevado para o *ultra-fine* quando comparado com as bases de dados disponíveis. Para exemplificar, a base HYENA e FIGER contam com 505 e 112 rótulos, respectivamente.

A Tabela III mostra o resultado obtido por *ensemble*, segundo a ordem estabelecida para o preenchimento das respostas. Usando apenas o Mistral, embora ele tenha se destacado na precisão, apresentou o segundo pior resultado individual para o *recall* e foi o modelo com maior número de resultados vazios. Assim, quando aplicado individualmente, não incrementou o processo de classificação,

o que pode ser explicado pela quantidade de saídas vazias e pelo número de rótulos previstos que não estão presentes na base de teste. Por outro lado, ao ser associado ao Gemma, no próximo *ensemble*, observa-se uma melhora, em termos de F1, de aproximadamente nove pontos percentuais, superando, inclusive, os *baselines* adotados. Isso se deve à complementariedade dos modelos, que possuem poucas saídas vazias em comum. Os dois últimos *ensembles* melhoram os resultados em uma proporção menor, demonstrando a eficiência do segundo em classificar a maior parte exemplos de testes. Ainda assim, o melhor resultado é obtido quando houve a consulta dos quatro LLM's utilizados, quando o modelo Qwen conseguiu completar a classificação de toda a base de testes, rotulando exemplos que nenhum dos outros três modelos foram capazes. A Tabela IV mostra comparativamente os resultados obtidos pelos modelos de *baseline* e a proposta deste trabalho. O método proposto conseguiu superar consistentemente os modelos de referência, em termos de F1. Mas, teve resultado inferior em termos de precisão. Ao não restringir o conjuto de rótulos em que os LLM's devem classificar as entidades, contribui-se com a elevação de rótulos que o ensemble foi capaz de prever, sobretudo no nível ultra-fino, que não estão presentes na base de testes. Isso limita e acaba penalizando a precisão, como observado por Choi et al. [2018]. Por outro lado, a mesma escolha favorece o *recall*, quando não se limita o número de rótulos que devem ser previstos.

5. CONCLUSÃO

O processamento de linguagem natural é uma área dinâmica que apresentou acelerada evolução nos últimos anos, mas que ainda apresenta desafios e oportunidades a serem explorados. Os grandes de modelos de linguagem, incluindo os de domínio público, têm representado uma nova evolução para a NLP, como um recurso altamente capaz de revolucionar os modelos tradicionais. No caso da UFET, ampliar a cobertura de rótulos em equilíbrio com o desempenho dos métodos ainda é uma necessidade. Neste trabalho, foi proposto um método de *ensemble* de LLM's para UFET e foram alcançados resultados superiores aos métodos estado da arte tanto quanto ao desempenho quanto à expansão da quantidade rótulos previstos. Como trabalhos futuros, pretende-se avaliar outros LLM's e *prompts*, o incremento da quantidade de épocas no treinamento e outras estratégias para o *ensemble*.

AGRADECIMENTOS

Agradecemos o suporte da UFOP, FAPEMIG, CAPES e ANPq.

REFERENCES

AI@Meta. Llama 3 model card. *arXiv:2407.21783*, 2024.

Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., Hui, B., Ji, L., Li, M., Lin, J., Lin, R., Liu, D., Liu, G., Lu, C., Lu, K., Ma, J., Men, R., Ren, X., Ren, X., Tan, C., Tan, S., Tu, J., Wang, P., Wang, S., Wang, W., Wu, S., Xu, B., Xu, J., Yang, A., Yang, H., Yang, J., Yang, S., Yao, Y., Yu, B., Yuan, H., Yuan, Z., Zhang, J., Zhang, X., Zhang, Y., Zhang, Z., Zhou, C., Zhou, J., Zhou, X., and Zhu, T. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Choi, E., Levy, O., Choi, Y., and Zettlemoyer, L. Ultra-Fine Entity Typing, 2018. arXiv:1807.04905 [cs].

Dai, H., Song, Y., and Wang, H. Ultra-Fine Entity Typing with Weak Supervision from a Masked Language Model. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, Online, pp. 1790–1799, 2021.

Modelo	Precisão	Recall	F1-Score
Choi et al. [2018]	47,1	24,2	32,0
PLET(T_3)	59,2	36,6	45,2
PLET(T_4)	**61,4**	36,9	46,1
Mistral + Gemma + Llama 3 + Qwen	54,1	**42,7**	**47,7**

Table IV. Resultados do melhor método usando o conjunto de dados de teste. A parte superior se refere aos *baseline* e seus resultados foram retirados de seus respectivos artigos.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2019. arXiv:1810.04805 [cs].

Ding, N., Hu, S., Zhao, W., Chen, Y., Liu, Z., Zheng, H., and Sun, M. OpenPrompt: An Open-source Framework for Prompt-learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, V. Basile, Z. Kozareva, and S. Stajner (Eds.). Association for Computational Linguistics, Dublin, Ireland, pp. 105–113, 2022.

Feng, Y., Pratapa, A., and Mortensen, D. R. Calibrated Seq2seq Models for Efficient and Generalizable Ultra-fine Entity Typing, 2023. arXiv:2311.00835 [cs].

Gemma Team, T. M., Hardin, C., Dadashi, R., Bhupatiraju, S., Sifre, L., Rivière, M., Kale, M. S., Love, J., Tafti, P., Hussenot, L., and et al. Gemma: Open models based on gemini research and technology, 2024.

Gillick, D., Lazic, N., Ganchev, K., Kirchner, J., and Huynh, D. Context-Dependent Fine-Grained Entity Type Tagging, 2016. arXiv:1412.1820 [cs] version: 2.

Grishman, R. and Sundheim, B. Message Understanding Conference- 6: A Brief History. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*, 1996.

Hearst, M. A. Automatic acquisition of hyponyms from large text corpora. In *COLING 1992 Volume 2: The 14th International Conference on Computational Linguistics*, 1992.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. LoRA: Low-Rank Adaptation of Large Language Models, 2021. arXiv:2106.09685 [cs].

Jehangir, B., Radhakrishnan, S., and Agarwal, R. A survey on Named Entity Recognition — datasets, tools, and methodologies. *Natural Language Processing Journal* vol. 3, pp. 100017, June, 2023.

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., de las Casas, D., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., and Sayed, W. E. Mistral 7b, 2023.

Li, J., Sun, A., Han, J., and Li, C. A Survey on Deep Learning for Named Entity Recognition. *IEEE Trans. Knowl. Data Eng.* 34 (1): 50–70, 2020. arXiv:1812.09449 [cs].

Li, Z., Li, X., Liu, Y., Xie, H., Li, J., Wang, F.-l., Li, Q., and Zhong, X. Label Supervised LLaMA Finetuning, 2023. arXiv:2310.01208 [cs].

Ling, X. and Weld, D. Fine-Grained Entity Recognition. *AAAI* 26 (1): 94–100, 2012.

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., and Neubig, G. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Comput. Surv.* 55 (9): 1–35, Sept., 2023.

Mai, K., Pham, T.-H., Nguyen, M. T., Nguyen, T. D., Bollegala, D., Sasano, R., and Sekine, S. An Empirical Study on Fine-Grained Named Entity Recognition. In *Proceedings of the 27th International Conference on Computational Linguistics*. Association for Computational Linguistics, Santa Fe, New Mexico, USA, pp. 711–722, 2018.

Qian, J., Liu, Y., Liu, L., Li, Y., Jiang, H., Zhang, H., and Shi, S. Fine-grained Entity Typing without Knowledge Base. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, pp. 5309–5319, 2021.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* 21 (140): 1–67, 2020.

Sekine, S. Extended Named Entity Ontology with Attribute Information. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, and D. Tapias (Eds.). European Language Resources Association (ELRA), Marrakech, Morocco, 2008.

Wang, Y., Xin, X., and Guo, P. An Empirical Study of Pre-trained Embedding on Ultra-Fine Entity Typing. In *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, Toronto, ON, Canada, pp. 2656–2661, 2020.

Xiong, W., Wu, J., Lei, D., Yu, M., Chang, S., Guo, X., and Wang, W. Y. Imposing Label-Relational Inductive Bias for Extremely Fine-Grained Entity Typing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Minneapolis, Minnesota, pp. 773–784, 2019.

Yadav, V. and Bethard, S. A Survey on Recent Advances in Named Entity Recognition from Deep Learning models. In *Proceedings of the 27th International Conference on Computational Linguistics*, E. M. Bender, L. Derczynski, and P. Isabelle (Eds.). Association for Computational Linguistics, Santa Fe, New Mexico, USA, pp. 2145–2158, 2018.

Yosef, M. A., Bauer, S., Hoffart, J., Spaniol, M., and Weikum, G. HYENA: Hierarchical Type Classification for Entity Names. In *Proceedings of COLING 2012: Posters*. The COLING 2012 Organizing Committee, Mumbai, India, pp. 1361–1370, 2012.