PREAnoTe: Uma abordagem de anotação de corpus para o ajuste fino de *Large Language Model* pré-treinado

Jones O. Avelino[1,4], Giselle F. Rosa[2], Gustavo R. Danon[3], Kelli F. Cordeiro[5], Maria Cláudia Cavalcanti[1,2,3]

[1]Pós-graduação em Engenharia de Defesa, [2]Pós-graduação em Sistemas e Computação e [3]Graduação em Engenharia da Computação – IME – Rio de Janeiro – RJ – Brasil

[4]Centro de Análise de Sistemas Navais – Rio de Janeiro – RJ – Brasil

[5]Subchefia de Comando e Controle – Ministério da Defesa – Brasília – DF – Brasil

{jones.avelino,giselle.farias,gustavo.danon,kelli,yoko}@ime.eb.br

Abstract. *Fine-tuning a Language Model (LM) requires a large, categorized and annotated corpus. However, corpora are scarce, and manual annotation is costly. As an alternative, the Distant Supervision approach has emerged, which can use Semantic Resources (SR). Nevertheless, there are gaps in using SR to minimize the annotation cost. This article proposes PREAnoTe, an approach that supports annotation using regular expression rules, guided by a metamodel and SR. The experiments showed promising results, achieving 95% accuracy for entities and 76% for relations, culminating in an adjusted LM with 86% precision and coverage.*

Resumo. *O ajuste fino de um Modelo de Linguagem (ML) necessita de corpus volumoso, categorizado e anotado. Contudo, corpora são escassos e a anotação manual é custosa. Como alternativa, surgiu a abordagem Distant Supervision que pode usar Recursos Semânticos (RS). Entretanto, há lacunas na utilização de RS para minimizar o custo da anotação. Este artigo propõe PREAnoTe, uma abordagem capaz de apoiar a anotação, utilizando regras de expressão regular, orientado por um metamodelo e RS. Os experimentos mostraram resultados promissores, alcançando uma precisão de 95% nas entidades e 76% nas relações, culminando em um ML ajustado com 86% de precisão e cobertura.*

1. Introdução

Nos últimos anos, houve uma busca por obtenção de conhecimento em textos através do uso de ML, em que se destacou o BERT models for Brazilian Portuguese (BERTimbau) [Souza et al. 2020]. Entretanto, um ML pode ser mais útil quando ajustado ao contexto do domínio, o que impõe o desafio de obter um corpus volumoso, anotado e categorizado. No domínio biomédico, há corpora anotados e com ML treinados, como o BioBERT[1]. Por tais recursos serem escassos, há autores que defendem o uso de textos ou documentos doutrinários[2] para compor um corpus, dado o seu caráter pedagógico [Liu et al. 2023].

Ainda assim, a anotação manual de um corpus com exemplos do domínio é custosa e demanda o envolvimento de especialistas. Como alternativa, há autores que adotam

[1]https://github.com/naver/biobert-pretrained

[2]Conjunto de princípios, conceitos e procedimentos expostos de forma integrada [BRASIL 2018].

métodos de *Distant Supervision (DS)*, aproveitando bases externas [Mintz et al. 2009]. Contudo, é necessária uma abordagem flexível em que os construtos não se restrinjam a domínios específicos e seja apta a explorar o conhecimento extraído de bases externas a partir de expressões regulares. Neste trabalho propomos PREAnoTe, uma abordagem mais flexível que apoia o processo de criação de um corpus anotado a partir de textos doutrinários, baseado em regras de expressão regular, orientado por um metamodelo de construtos genéricos e recursos semânticos (glossários ou ontologias). Para apoiar a avaliação da abordagem, foi desenvolvido o protótipo PREAnoTeTool em Python.

2. Conceitos básicos

O Processamento de Linguagem Natural (PLN) é um campo de pesquisa ligado à área da Inteligência Artificial (IA) que investiga problemas computacionais relacionados à linguagem humana, propondo métodos e soluções. Um desses problemas é a construção de corpus anotados, como o HAREM[3], em função de envolver dados linguísticos ou exemplos obtidos de conhecimentos variados. Além disso, a anotação abrange a definição de classes ou categorias e a sua interpretação. Dada a semântica do domínio envolve especialistas para refinar o treinamento do ML ou LLM, do inglês *Large Language Model (LLM)*. Por isso, o uso de RS, como as ontologias, pode apoiar a anotação de um corpus, servindo como fontes de relações semânticas a serem anotadas nos textos [Caseli and Nunes 2023].

A Extração de informação (EI), do inglês *Information Extraction*, tem como objetivo obter informação estruturada a partir de dados reais não estruturados. Nela, há tarefas de Reconhecimento de Entidades Nomeadas (NER), do inglês *Named Entity Recognition*, que identifica categorias de entidades, como *pessoa* e *organização*, e de Extração de relação (RE), do inglês *Relation Extraction*, que expressa a semântica entre as entidades, como *trabalha-em*, que denota o local onde a pessoa trabalha [Caseli and Nunes 2023].

A estrutura dos dados das tarefas de NER e RE é representada por um conjunto de triplas (arg_1, rel, arg_2), onde arg_1 e arg_2 são entidades e rel, a relação entre elas, a qual forma a base de conhecimento do ML [Collovini et al. 2020]. Os ML são complexos, porém seu conhecimento pode ser estruturado através de um grafo *Resource Description Framework (RDF)*[4], que também é formado por triplas, $T(m) = \{e_{1i}, r_k, e_{2i}\}$, Além disso, o RDF é propício para realização de consultas e inferências [Hogan et al. 2021].

3. Trabalhos relacionados

Os trabalhos de [Fries et al. 2021], [Zhou et al. 2022] e [Liu et al. 2023] implementam métodos de DS que buscam minimizar a anotação manual, utilizando RS a fim de recuperar o conhecimento. Entretanto, as categorias de entidades e relações são restritas, dificultando a sua aplicação em outros domínios. Além disso, esses trabalhos focam no uso de ontologias, restringindo as regras preestabelecidas. Por outro lado, os glossários por serem textuais favorecem o uso de expressões regulares e seus textos podem ser incorporados ao corpus. Por fim, somente [Fries et al. 2021] avalia a acurácia da anotação, mesclando ontologias com regras adicionais através de métricas qualitativas.

Diferente dos demais trabalhos, este artigo propõe uma abordagem flexível que adota regras de expressão regular combinando RS e um metamodelo, abstraindo metada-

[3]https://www.linguateca.pt/HAREM/

[4]https://www.w3.org/RDF/

dos utilizados, i.e., não é restrito a domínios específicos. Além disso, métricas quantitativas são utilizadas para avaliar a pré-anotação em comparação com a anotação manual.

4. PREAnoTe

A abordagem PREAnoTe tem como objetivo apoiar a anotação de corpus de textos para realizar o ajuste fino, ou *fine-tuning*, de ML pré-treinados, nas tarefas de NER e RE (Figura 1). O conhecimento do ML pode ser extraído em um grafo RDF para os usuários realizarem consultas utilizando SPARQL Protocol and RDF Query Language (SPARQL).

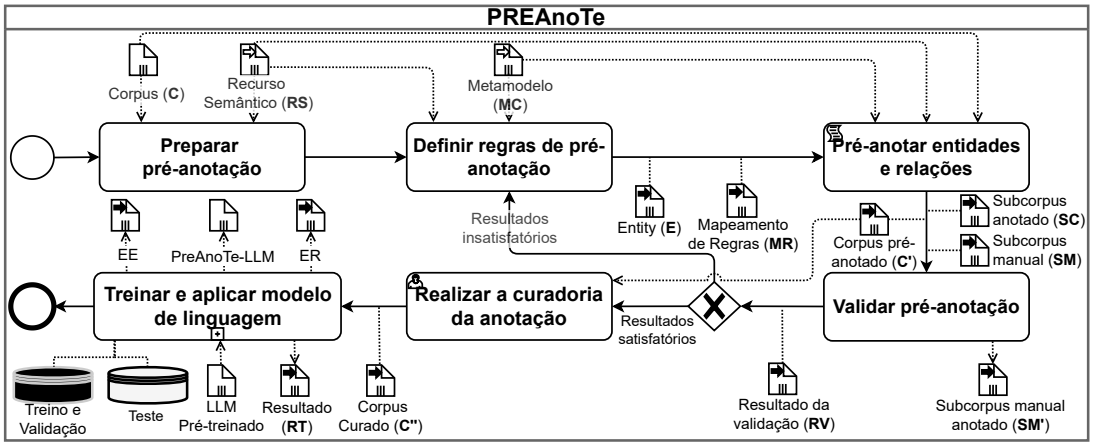

Figura 1. Processo da abordagem PREAnoTe.

Inicialmente, a atividade **Preparar pré-anotação** obtém e faz o pré-processamento do corpus, $C = \{s_1, s_2, \ldots, s_n\}$, e do RS. Considera-se que o RS pode ser um glossário ou ontologia com termos e suas definições. Assume-se que tais termos constituem um conjunto de entidades a serem anotadas em C. Em seguida, a atividade **Definir regras de pré-anotação** obtém um metamodelo conceitual, MC, ilustrado na Figura 2 [Avelino. et al. 2024], constituído de construtos[5] de alto nível de abstração, *Entity* (E), *Relation* (R) e suas especializações ($R_1, R_2, ...R_n$), como metacategorias para anotar C. Por exemplo, a metacategoria *responsible_for* é pré-definida no MC. Nessa atividade os termos do RS passam a ser instâncias distintas de *Entity* ($E = \{e_1, e_2, ..., e_n\}$). Com base nas definições dos termos de RS e no metamodelo são elaboradas expressões regulares para EI a fim de gerar o mapeamento de regras, $MR = \{R_k, t_{k,p}\}$, onde R_k é uma especialização de R, e $t_{k,p}$ é uma expressão regular associada à R_k. Por exemplo, ao analisar o termo COMANDANTE DE AERONAVE em RS, encontram-se expressões como (“responsável*”, “executado p*”, “cumprimento*”) que darão origem a uma expressão regular que será usada como regra para anotar a relação “*responsible_for*”.

A atividade **Pré-anotar entidades e relações**, representada no Algoritmo 1, gera o corpus pré-anotado C', a partir das entradas: C, E, MR. Para cada $s_i \in C$, e para cada par $e_j, e_m \in E$, verifica se tal par está em s_i e se a expressão regular $t_{k,p}$ referente a R_k $(R_k, t_{k,p}) \in MR$ está entre as posições (l_1, l_2) de e_j, e_m em s_i. Ao encontrar, é criada a tripla, e_j, R_k, e_m, que pré-anota s_i em C'. Além disso, são gerados os subcorpora SC e SM com as mesmas sentenças de textos para validar a pré-anotação, onde $SC \subset C'$, contém textos pré-anotados, e $SM \subset C$, contém textos sem anotação.

[5]Um modelo é um sistema básico de construtos usado na descrição da realidade [Kent 2012].

Em **Validar pré-anotação**, avaliam-se as pré-anotações de C' a partir de SC e SM. Para tal, o especialista anota SM manualmente, apoiado por ferramenta automatizada, e gera SM'. Em seguida, as anotações de SC e SM' são comparadas. No exemplo da Figura 2, "AERONAVE" pode ser anotada em ambos os subcorpora como *entity*, representando um *True Positive* (TP) $(SC \cap SM')$. Contudo, o especialista pode divergir das pré-anotações e anotar manualmente, p. ex., "TRIPULAÇÃO" como *entity*. Nesse caso, a divergência é representada como *False Negative* (FN) $(SM' - SC)$. Assim como, o contrário pode ocorrer, representado como *False Positive* (FP) $(SC - SM')$. Os valores de TP, FP e FN são utilizados para calcular as métricas *precision* e *recall*, representadas por RV. Similarmente, a mesma avaliação é feita para as relações, porém de duas formas diferentes. Inicialmente, considera-se uma ocorrência FP quando SC tem anotada uma relação distinta da rotulada por SM' (avaliação chamada de "categorizada"). Em um segundo momento, considera-se que quando SC encontra uma relação, independente do rótulo atribuído, essa é uma ocorrência TP (avaliação chamada "não categorizada"). Caso RV seja satisfatório, segue-se para **Realizar a curadoria da anotação** de C' a fim de gerar o corpus curado C''. Caso contrário, retorna-se para **Definir regras de pré-anotação**.

Por fim, na atividade **Treinar e aplicar modelo de linguagem**, são obtidos C'' e o LLM pré-treinado. Cada s_i de C'' é ordenada aleatoriamente, *tokenizada* e C'' é dividido em 80%, para treino e validação, e 20% para teste. Em seguida, ocorre o ajuste fino e PREAnoTe-LLM é gerado. RT representa as métricas *precision*, *recall* e *F1-Score* que servem para avaliar PREAnoTe-LLM. Além disso, podem ser extraídas de PREAnoTe-LLM as entidades, $EE = \{e_1, e_2, ...e_n\}$, onde e_1 é "comandante de aeronave" e e_2 é "aeronave". Assim como, as triplas de entidades e relações, $ER = \{(e_i, r_j, e_k) \mid e_i, e_k \in EE$ estão semanticamente relacionados por $r_j \in R$, em PREAnoTe-LLM$\}$, representado pela tripla $\{e_1, r_8, e_2\}$, onde r_8 é *responsible_for*. Nesse caso, os recursos vinculados a EE representam os nós, bem como as triplas ER representam as arestas no grafo RDF.

Algoritmo 1: Pré-anotar entidades e relações

Entrada: $C = \{s_i\}, i = 1, \ldots, |C|$; $E = \{e_j\}, j = 1, \ldots, |E|\}$;
$MR = \{(R_k, t_{k,p})\}, k = 1, \ldots, |R| \wedge p = 1, \ldots, z$;
Saída: $C' = \{(s_i, e_j, R_k, e_m)$

```
para cada (s_i) ∈ C faça
    para cada e_j, e_m ∈ E, e_j ≠ e_m faça
        se (((l_1, l_2) ← search_entities(s_i, e_j, e_m))∧
        (search_rule(s_i, l_1, l_2, t_k,p))) então
            C' ← pre_annotation_t(s_i, e_j, R_k, e_m)
        fim
    fim
fim
retorna C';
```

5. Estudo de caso

No contexto militar, as operações demandam das Forças Armadas (FA) esforços para manter seu efetivo capacitado e pronto para o emprego. O acervo de Doutrinas Militares (DM)[6] fornece informações para capacitar o pessoal. Contudo, essas DM são textuais e

[6]https://bdex.eb.mil.br/jspui/

um dos desafios é extrair delas conhecimento útil e disponibilizar o seu acesso ao pessoal. Assim, este trabalho explora o estudo de caso a partir de um experimento, através do protótipo **PREAnoTeTool**[7] desenvolvido em Python, que utiliza trechos de DM para fazer o ajuste fino de um ML, permitindo a estruturação através de um grafo RDF.

Na Figura 2, é ilustrada a aplicação de PREAnoTeTool a partir da amostra de RS e C, contendo 3 mil textos de DM e do Glossário [BRASIL 2018]. MC e MR representam o metamodelo e o mapeamento das regras. Além disso, um trecho de C é destacado com as entidades sublinhadas e relações de C', pré-anotadas, e de C'', curadas. Note que em C'' as anotações *tripulação* e *vôo*, e_5 e e_6, assim como as relações, *composed_of* e *applied_to*, foram feitas manualmente pelo especialista, contudo as demais pré-anotadas.

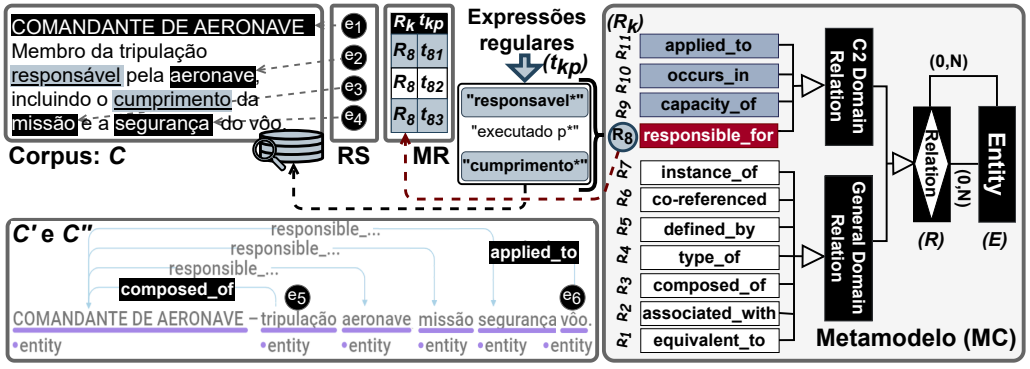

Figura 2. Preparação, definição regras, pré-anotação e curadoria.

Na Tabela 1, são apresentados os resultados da validação de C' a partir da seleção de 100 textos para compor SM' e SC. Com base no *F1-score*, deduz-se que o uso de RS com a pré-anotação mostrou-se como uma boa estratégia para as entidades. Ademais, considerando que um especialista gastou 10 horas para anotar SM', deduz-se que seriam gastos 30 vezes mais horas para anotar manualmente todo C', indicando que PREAnoTe pode contribuir com a redução do esforço e favorecer a construção do corpus. Por outro lado, o resultado das relações indica que o especialista deve atuar com maior atenção.

Tabela 1. Pré-anotação baseada em Regras (SC) x Anotação manual (SM')

		TP	FP	FN	*Precision*	*Recall*	*F1-Score*
Entidades		441	22	307	95%	59%	72%
Relações	**não categorizadas**	275	88	398	76%	41%	53%
	categorizadas	163	200	511	45%	24%	31%

Na Tabela 2, são apresentados os resultados do ajuste fino de PREAnoTe-LLM a partir de C'' e do LLM pré-treinado BERTimbau [Souza et al. 2020]. Além disso, os *pipelines* do Spacy[8], aplicados nas tarefas NER e RE, foram parametrizados, respectivamente, com os valores: *Batch_size* de 128 e 500; *Max_Length* de 4096 e 250; e *Threshold* de 0,5 para RE. Em destaque, o *pipeline* **pt_core_news_sm** obteve os melhores resultados em ambas as tarefas. Ao analisar o número de épocas, o *Batch_size* e o número de textos para treino, deduz-se que 304 épocas na tarefa NER indicam um bom resultado. Por outro

[7]https://github.com/jonesavelino/preanotetool
[8]https://spacy.io/models/pt

lado, na tarefa RE, apesar de alta precisão e cobertura, o número de 66 épocas deve ser investigado, demandando novas rodadas de treinamento com ajustes no *Dropout*.

Tabela 2. Resultados do ajuste fino de PREAnoTe-LLM

	Pré-treinado	Pipeline	Épocas	*Precision*	*Recall*	*F1-Score*
NER	BERTimbau Base	**pt_core_news_sm**	**304**	**86,56%**	**86,48%**	**86,51%**
		pt_core_news_md	159	84,42%	79,07%	81,66%
		pt_core_news_lg	323	85,44%	80,24%	82,76%
RE		**pt_core_news_sm**	**66**	**98,06%**	**98,37%**	**98,21%**
		pt_core_news_md	49	91,93%	81,28%	86,28%
		pt_core_news_lg	46	91,97%	87,24%	86,54%

Na Figura 3, é ilustrado o resultado da consulta SPARQL a partir dos recursos do grafo RDF obtidos de EE e ER extraídos através do ajuste fino que gerou PREAnoTe-LLM. Sendo assim, a consulta retorna os recursos diretamente vinculados ao nó *comandante_de_aeronave* através de *responsible_for*, como *aeronave*, *missao* e *seguranca*.

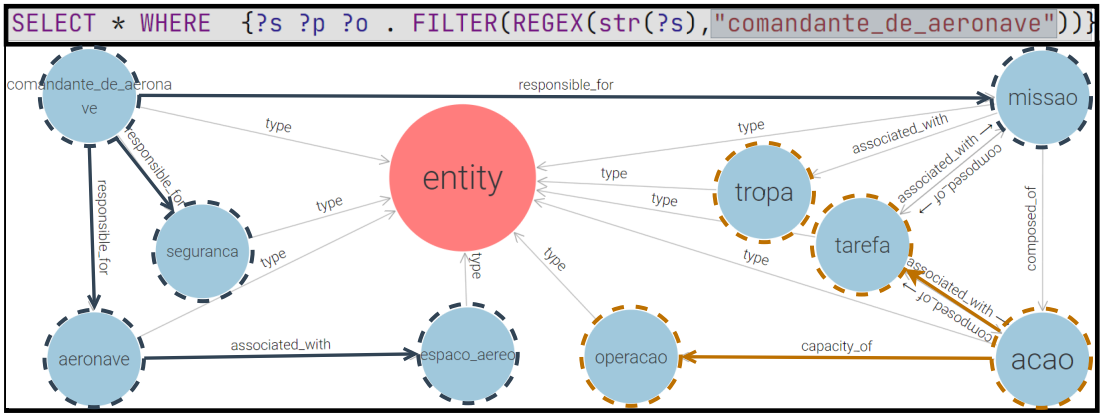

Figura 3. Grafo RDF destacando o recurso *comandante_de_aeronave* e relações.

6. Considerações finais

Este artigo apresentou PREAnoTe, uma abordagem para apoiar a criação de um corpus anotado a fim de realizar o ajuste fino do LLM pré-treinado. A abordagem foi implementada (PREAnoTeTool) e mostrou-se capaz de extrair o conhecimento de PREAnoTe-LLM, estruturando-o em um grafo RDF disponível para consultas. O protótipo PREAnoTeTool foi submetido a um estudo de caso e os resultados mostraram-se promissores, evidenciando a utilidade e a viabilidade da abordagem proposta. Trabalhos futuros incluem: a implementação com outros Recursos Semânticos de modo a explorar a abordagem e comparar os resultados; e a exploração de inferências no grafo indiretamente vinculadas ao nó *comandante_de_aeronave*, como destacado em *tropa*, *acao* e *tarefa* na Figura 3.

Agradecimentos

Agradecemos ao CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico, e ao IME - Instituto Militar de Engenharia, pelo apoio através de bolsa de iniciação científica (PIBIC Edital 2023/2024), e à FINEP/DCT/FAPEB (nº 2904/20-01.20.0272.00), pelo apoio ao Projeto ‘Sistemas de Sistemas de Comando e Controle”.

Referências

Avelino., J., Rosa., G., Danon., G., Cordeiro., K., and C. Cavalcanti., M. (2024). Knowledge Graph generation from text using Supervised Approach supported by a Relation Metamodel: An application in C2 domain. In *Proceedings of the 26th International Conference on Enterprise Information Systems - Volume 1: ICEIS*, pages 281–288. INSTICC, SciTePress.

BRASIL (2018). Glossário de termos e expressões para uso no Exército. *Exército. Estado-Maior*.

Caseli, H. M. and Nunes, M. G. V., editors (2023). *Processamento de Linguagem Natural: Conceitos, Técnicas e Aplicações em Português*. BPLN. `https://brasileiraspln.com/livro-pln`.

Collovini, S., Gonçalves, P. N., Cavalheiro, G., Santos, J., and Vieira, R. (2020). Relation Extraction for Competitive Intelligence. In *International Conference on Computational Processing of the Portuguese Language*, pages 249–258. Springer.

Fries, J. A., Steinberg, E., Khattar, S., Fleming, S. L., Posada, J., Callahan, A., and Shah, N. H. (2021). Ontology-driven weak supervision for clinical entity classification in electronic health records. *Nature communications*, 12(1):2017.

Hogan, A., Blomqvist, E., Cochez, M., D'amato, C., Melo, G. D., Gutierrez, C., Kirrane, S., Gayo, J. E. L., Navigli, R., Neumaier, S., Ngomo, A.-C. N., Polleres, A., Rashid, S. M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., and Zimmermann, A. (2021). Knowledge Graphs. *ACM Computing Surveys*, 54(4).

Kent, W. (2012). *Data and reality: a timeless perspective on perceiving and managing information*. Technics publications.

Liu, P., Qian, L., Zhao, X., and Tao, B. (2023). The construction of Knowledge Graphs in the Aviation Assembly Domain Based on a Joint Knowledge Extraction Model. *IEEE Access*, 11:26483–26495.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant Supervision for Relation Extraction without Labeled Data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, Suntec, Singapore. Association for Computational Linguistics.

Souza, F., Nogueira, R., and Lotufo, R. (2020). BERTimbau: Pretrained BERT Models for Brazilian Portuguese. In Cerri, R. and Prati, R. C., editors, *Intelligent Systems*, pages 403–417, Cham. Springer International Publishing.

Zhou, J., Li, X., Wang, S., and Song, X. (2022). NER-based Military Simulation Scenario development process. *The Journal of Defense Modeling and Simulation*, 20(4):563–575.