MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA
INSTITUTO MILITAR DE ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE DEFESA

JONES DE OLIVEIRA AVELINO

IDEA-C2: UMA ABORDAGEM HÍBRIDA DE MODELAGEM CONCEITUAL APOIADA POR UM MODELO DE LINGUAGEM E UM METAMODELO DE DADOS NO CONTEXTO DE COMANDO E CONTROLE

RIO DE JANEIRO
2026

JONES DE OLIVEIRA AVELINO

IDEA-C2: UMA ABORDAGEM HÍBRIDA DE MODELAGEM CONCEITUAL APOIADA POR UM MODELO DE LINGUAGEM E UM METAMODELO DE DADOS NO CONTEXTO DE COMANDO E CONTROLE

Tese apresentada ao Programa de Pós-graduação em Engenharia de Defesa do Instituto Militar de Engenharia, como requisito parcial para a obtenção do título de Doutor em Ciências em Engenharia de Defesa.

Orientador(es): Maria Cláudia Reis Cavalcanti, D.Sc.
Kelli de Faria Cordeiro, D.Sc.

Rio de Janeiro

2026

INSTITUTO MILITAR DE ENGENHARIA
Praça General Tibúrcio, 80 – Praia Vermelha
Rio de Janeiro – RJ CEP: 22290-270

Avelino, Jones de Oliveira.
IDEA-C2: uma abordagem híbrida de modelagem conceitual apoiada por um modelo de linguagem e um metamodelo de dados no contexto de Comando e Controle / Jones de Oliveira Avelino. – Rio de Janeiro, 2026.
177 f.

Orientador(es): Maria Cláudia Reis Cavalcanti e Kelli de Faria Cordeiro.

Tese (doutorado) – Instituto Militar de Engenharia, Engenharia de Defesa, 2026.

1. grafo de conhecimento; modelo de linguagem; modelo de domínio; data-driven; theory-driven. i. Reis Cavalcanti, Maria Cláudia (orient.) ii. Faria Cordeiro, Kelli de (orient.) iii. Título

JONES DE OLIVEIRA AVELINO

IDEA-C2: uma abordagem híbrida de modelagem conceitual apoiada por um modelo de linguagem e um metamodelo de dados no contexto de Comando e Controle

Tese apresentada ao Programa de Pós-graduação em Engenharia de Defesa do Instituto Militar de Engenharia, como requisito parcial para a obtenção do título de Doutor em Ciências em Engenharia de Defesa.

Orientador(es): Maria Cláudia Reis Cavalcanti e Kelli de Faria Cordeiro.

Aprovada em 09 de fevereiro de 2026, pela seguinte banca examinadora:

Prof[a]. **Maria Cláudia Reis Cavalcanti** - D.Sc. do IME - Presidente

Prof[a]. **Kelli de Faria Cordeiro** - D.Sc. do IME

Prof. **Ronaldo Ribeiro Goldschmidt** - D.Sc. do IME

Prof. **Julio Cesar Duarte** - D.Sc. do IME

Prof. **João Luiz Rebelo Moreira** - Ph.D. da Universidade de Twente

Prof. **Sérgio Manuel Serra da Cruz** - D.Sc. da UFRJ

Rio de Janeiro

2026

Este trabalho é dedicado a todos que acreditam na educação
e na pesquisa científica como fator impulsionador de uma nação.

AGRADECIMENTOS

Às professoras e orientadoras, Maria Cláudia e CMG(T) Kelli, por terem me apoiado nessa caminhada. Durante todo esse período a colaboração, a paciência e, acima de tudo, a confiança que depositaram em mim foi fundamental para o meu desenvolvimento.

À Marinha do Brasil por ter acreditado, concedido tempo e proporcionado a realização de um objetivo de vida.

Aos meus colegas de IME TC Marcus Albert, Maj Giselle, Gustavo Danon, Flávio Ferreira, André Demori, Júlio Tesolin e Josinaldo por terem me ajudado, dividido experiências e colaborado durante o curso. Um agradecimento especial à minha amiga Maj Giselle de Farias Rosa que muito colaborou com este trabalho e me ajudou com todo o seu conhecimento e apoio incansável. Outro agradecimento especial ao Gustavo Danon pelo empenho e disponibilidade para colaborar com a pesquisa.

Aos meus colegas da MB, Cláudio Coreixas, Carlos Eduardo Barbosa, Leandro Ouriques, Bruno Hansen, Mônica Fraga, Eduardo André, Fernando Muradas, Lucimar Lial, Marta Rigaud, Gláucia Botelho, Victor Bastos e Thaisa pelo apoio e incentivo antes e durante o curso.

À minha família pelo suporte, entendimento nas ausências e colaboração nessa caminhada. Em especial à minha Mãe, um exemplo de perseverança e por tudo que fez por mim. Além disso, à minha esposa que tanto me ajudou e deu o suporte durante o curso.

Por fim, a Cristo por sua infinita misericórdia e por permitir que eu esteja aqui.

"Seja você quem for, seja qual for a posição social que você tenha na vida,
a mais alta ou a mais baixa, tenha sempre como meta muita força,
muita determinação e sempre faça tudo com muito amor e
com muita fé em Deus, porque um dia você chega lá.
De alguma maneira você chega lá."
(Ayrton Senna)

RESUMO

A obtenção de conhecimento a partir de dados textuais foi impulsionada pelo avanço dos modelos de linguagem, cujo desempenho pode ser aprimorado por meio do ajuste fino em domínios específicos. Contudo, abordagens orientadas por dados (*data-driven*) geram modelos subsimbólicos que apresentam limitações, como falta de explicabilidade e vieses. Em contraste, abordagens orientadas por teoria (*theory-driven*) baseiam-se em conceituações formais para a construção de modelos de domínio simbólicos, embora enfrentem desafios na extração de classes e relações relevantes a partir de textos. Nesse contexto, esta tese propõe o IDEA-C2, uma abordagem supervisionada híbrida que combina textos doutrinários, recursos semânticos e um metamodelo de alto nível para anotar corpora e ajustar modelos de linguagem pré-treinados em língua portuguesa. A abordagem emprega técnicas de pré-anotação heurística e permite a geração de knowledge graphs (KG) flexíveis, viabilizando consultas exploratórias e inferências, a fim de apoiar o desenvolvimento de modelos de domínio (DM). Avaliada em seis experimentos distintos, a abordagem apresentou resultados promissores. Na pré-anotação do corpus, IDEA-C2 alcançou uma precisão de 95% nas entidades e 76% nas relações, culminando em um Modelo de Linguagem (ML) ajustado ao contexto com uma precisão e cobertura acima de 85%. Em outro experimento mais amplo, envolvendo 28 participantes, ao aplicar o ML ajustado combinado com o KG no apoio à construção de um DM, os resultados do IDEA-C2 mostraram que 40% das classes e relações do KG foram similares às dos DM construídos de maneira tradicional. Esses resultados demonstram a utilidade e viabilidade da abordagem IDEA-C2 tanto na geração de artefatos essenciais ao ajuste de um ML e geração de KG quanto na sua aplicação na construção de um DM.

Palavras-chave: grafo de conhecimento; modelo de linguagem; modelo de domínio; data-driven; theory-driven.

ABSTRACT

The extraction of knowledge from textual data has been driven by advances in language models (LM), whose performance can be improved through fine-tuning in specific domains. However, data-driven approaches generate sub-symbolic models that have limitations, such as lack of explainability and biases. In contrast, theory-driven approaches rely on formal conceptualizations to construct symbolic domain models, although they face challenges in extracting relevant classes and relations from texts. In this context, this thesis proposes IDEA-C2, a hybrid supervised approach that combines doctrinal texts, semantic resources, and a high-level metamodel to annotate corpora and fine-tune pre-trained language models in Portuguese. The approach employs heuristic pre-annotation techniques and allows the generation of flexible knowledge graphs (KG), enabling exploratory queries and inferences to support the development of domain models (DM). Evaluated in six separate experiments, the approach showed promising results. In the corpus pre-annotation phase, IDEA-C2 achieved 95% precision in entities and 76% in relations, resulting in a context-aware LM with precision and recall above 85%. In another larger experiment involving 28 participants, when applying the fine-tuned LM model combined with the KG to support the construction of a DM, the results from IDEA-C2 showed that 40% of the KG's classes and relations were similar to those of traditionally constructed DMs. These results demonstrate the usefulness and viability of the IDEA-C2 approach both in generating artifacts essential to adjusting a LM and generating KG, and in its application in constructing a DM.

Keywords: knowledge graph; language model; domain model; data-driven; theory-driven.

LISTA DE ILUSTRAÇÕES

LISTA DE QUADROS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

BERT	Bidirectional Encoder Representations from Transformers
BPMN	Business Process Model and Notation
C2	Comando e Controle
CF	Constituição Federal
DL	Deep Learning
EMCFA	Estado-Maior Conjunto das Forças Armadas
EBIA	Estratégia Brasileira de Inteligência Artificial
EB	Exército Brasileiro
F Cte	Força Componente
FA	Forças Armadas
FAB	Força Aérea Brasileira
IA	Inteligência Artificial
KG	Knowledge Graph
LOD	Linked Open Data
MCTI	Ministério da Ciência, Tecnologia e Inovações
MB	Marinha do Brasil
MD	Ministério da Defesa
OWL	Web Ontology Language
PND	Política Nacional de Defesa
PLN	Processamento de Linguagem Natural
RDF	Remote Description Framework
S2C2	Sistemas de Sistemas de Comando e Controle
UFO	Unified Foundational Ontology
UML	Unified Modeling Language

SUMÁRIO

1 INTRODUÇÃO

Nas últimas décadas, com o aumento de dados textuais houve uma alta demanda de consumo e serviços sobre esses dados. Em 2015, um estudo divulgado pela revista Forbes, com base em relatórios da *International Data Corporation* (IDC), apontou que o volume de dados produzido naquele biênio foi o maior da história (6). Mesmo não havendo um número exato e apesar da Abordagem Relacional (7) existir a mais de cinco décadas, estima-se que de 80% a 90% dos dados produzidos no mundo ainda são não-estruturados (8). Além disso, algumas contribuições se destacaram no período com avanços significativos na técnica de Extração de informação (EI) em Processamento de Linguagem Natural (PLN). Esses avanços incrementaram a obtenção de conhecimento a partir da submissão de textos (estruturados, semiestruturados e não-estruturados) a Modelos de Linguagem (ML), explorando, por exemplo, tarefas de Reconhecimento de Entidades Nomeadas (NER) e Extração de relação (RE) (9).

Com a ascensão dos Large Language Models (LLMs), a extração de informações de modo interativo ganha contornos jamais vistos (10). Em 2023, por exemplo, o chatGPT atingiu 100 milhões de usuários em apenas dois meses, mesmo com a ferramenta apresentando alguns resultados ruidosos (11). Desde então, as tarefas de reconhecimento de entidades nomeadas e identificação de relações semânticas ligadas à extração de informações são facilitadas em função dos LLMs serem treinados com conjuntos de dados, ou corpora, anotados e volumosos. Na realidade, o treinamento é composto de textos com exemplos do mundo real extraídos de diversas fontes, por exemplo da Wikipedia[1], ou através de corpus de livros, como a BookCorpus, ou obtidos através de *Web crawls* de textos da internet. Esses textos são pré-processados, tratados, anotados e utilizados para realizar o ajuste dos pesos do vetor de *embeddings* (12, 13). Contudo, a anotação é custosa e os investimentos nessa área são vultosos. Um exemplo foi a aquisição pela Meta, em 2025, da Scale IA, uma empresa especializada em lidar com dados brutos para treinamento de LLMs (14).

Mesmo com o sucesso repentino, há críticas de alguns autores dada a natureza subsimbólica (*bottom-up*) ou *Data-Driven (DD)* dos LLMs. Isto é, eles obtêm o conhecimento através dos ajustes dos pesos dos próprios dados, sem considerar a conceituação formal do domínio (15). Além disso, há autores que afirmam que os LLMs possuem uma "cegueira conceitual", reproduzindo padrões estatísticos coerentes, em função de sua aleatoriedade estocástica. Isso pode acarretar problemas de falta de explicabilidade em algumas respostas, ou alucinações com possibilidade de geração de respostas falsas (16).

Em linha com a crítica sobre os LLMs, há autores renomados, como Yann LeCun,

[1] https://en.wikipedia.org/wiki/Main_Page

que reforçam esse posicionamento, afirmando que a eficiência estatística dos LLMs conflita com as representações conceituais de cognição humana (17). Por exemplo, supondo as sentenças, s_1: "O presidente autorizou a operação Acolhida." e s_2:"O presidente quebrou a perna durante a operação Acolhida." Ao analisar ambas as sentenças, o LLM é capaz de interpretar corretamente que a autorização partiu do presidente dado o padrão de combinação dos termos: "presidente", "autorizou" e "operação". Porém, o LLM não distingue a pessoa física do presidente com o seu papel institucional. Logo, ele interpreta erroneamente que o presidente quebrou a perna na operação, algo que um ser humano conseguiria diferenciar sem maiores dificuldades.

O processo de obtenção do conhecimento de um LLM é uma caixa-preta, dado que ele é composto por bilhões de parâmetros que são associações probabilísticas distribuídas por pesos na rede e não possui regras explícitas de suas decisões (subsimbólico) (15). O trabalho de Yang, Han e Poon(18) destaca o uso de Knowledge Graph (KG) para complementar os LLMs, atuando como um componente simbólico e explicável. Além disso, dada a sua flexibilidade de representação e utilização de mecanismos de inferências, o KG pode permitir a análise de interligações implícitas (19). Ademais, a exploração das interligações causais nos conjuntos de dados do KG pode minimizar as alucinações por permitir raciocínio fora do contexto literal dos dados (18).

Em parte, as atribuições dadas aos KG vão ao encontro das conclusões do trabalho de Saba(16), quando o autor defende o uso de uma arquitetura híbrida, i.e., composta por elementos subsimbólicos e simbólicos, ou *Theory-Driven (TD)*. Outros trabalhos reforçam a proposta híbrida, combinando tanto a parte estatística e generalizável dos LLMs quanto o simbolismo para o entendimento e raciocínio semântico (15). Não é por acaso que as *Big Techs* (e.g. Google, Amazon, IBM, eBay) vêm, desde de 2012, investindo massivamente em KGs por serem uma representação que favorece a organização e transmissão de conhecimento sobre fatos diversos (20).

Portanto, a obtenção de conhecimento através de um ML é caracterizada por inferências subsimbólicas baseadas em probabilidades estatísticas. Além disso, é influenciada pelo conteúdo do corpus, pela estratégia de anotação e curadoria, principalmente por ser uma atividade custosa e complexa. Porém, os estudos indicam que as arquiteturas híbridas podem obter melhores resultados, incluindo um simbolismo sobre os textos extraídos que podem ser representados em um KG. Dentro desse contexto, surge uma oportunidade de construir uma abordagem flexível, combinando aspectos DD e TD, que permita utilizar as melhores características de ambos: a parte generalizável e refinada de um ML capaz de inferir textos submetidos em um domínio, bem como a exploração aprimorada através de inferências sobre os recursos de dados em um KG a partir de suas representações conceituais.

1.1 Contextualização e Motivação

Com o intuito de definir o escopo de atuação do trabalho, nesta tese são considerados dois aspectos motivadores, descritos em detalhes nas subseções 1.1.1 e 1.1.2. O primeiro aspecto envolve a relevância acadêmica da pesquisa em função do contexto de extração de informações, utilizando ML e a estruturação dos dados extraídos através de um KG. O segundo aspecto destaca a aplicação de abordagens com potenciais que explorem tecnologias disruptivas aplicadas no contexto militar, observando as principais oportunidades de uso em relação às demandas das Forças Armadas (FA).

1.1.1 Relevância acadêmica no contexto de extração de informações

No que diz respeito à relevância da pesquisa, é inegável que os últimos avanços na técnica de Extração de informação (EI) são notáveis. Contudo, há estudos de casos que indicam que os ML pré-treinados podem ser mais úteis quando ajustados ao contexto específico e à tarefa que se pretende apoiar, também conhecido como *Fine-tuning* (12). Uma alternativa ao *fine-tuning*, é aplicar ML ajustados (e.g. SciERC(9), BioBERT(21), MatSciBERT(22) e TForMIX(23)) em um determinado domínio. Porém, ML ajustados são escassos e na maioria das vezes não abrangem os contextos de uso em função do seu processo de ajuste ser direcionado a um conjunto de categorias predeterminado. Por exemplo, o *Scientific Entity and Relation Corpus* (SciERC) é voltado ao domínio de Ciência da Computação e foi ajustado a partir de seis categorias de entidades (e.g. *task*, *method*, etc.) e sete categorias de relações (e.g. *used-for*, *feature-of*, etc.) (9). Apesar de ser um ML promissor, o SciERC é limitado a esse conjunto de categorias, não havendo a possibilidade, por exemplo, de analisar a autoria e revisão dos métodos ou, simplesmente, as análises temporais de quando um método foi publicado.

Nesse sentido, quando os ML ajustados em um domínio não atendem às demandas do contexto de uso, a alternativa viável é realizar o *fine-tuning*. Porém, há alguns desafios para realizar o ajuste fino de um ML. Primeiramente, é necessário obter um corpus de textos volumoso que represente uma porção significativa de um domínio específico. Além disso, o corpus deve estar anotado com categorias expressivas do domínio e, preferencialmente, curado e validado por seres humanos. Contudo, esses corpora são escassos e muitas vezes nem estão disponíveis, como exemplo no domínio militar. Uma alternativa que alguns autores propõem é compor o corpus através de textos de artigos científicos, como o próprio SciERC que é formado por mais de 500 resumos da área de científica (9) ou *Material Science* que é composto de 800 resumos anotados manualmente da área de Ciência de Materiais (24).

Como os resultados dos ML são influenciados tanto pelo conteúdo do corpus quanto por sua curadoria (15), alguns autores defendem como alternativa o uso de textos de

livros didáticos ou documentos doutrinários[2] para compor um corpus, dado o seu caráter pedagógico (25, 26). Esse tipo de abordagem não é nova e foi usada por outros autores para obter, organizar e representar o conhecimento (27). Essencialmente, isso ocorre em função desses tipos de documentos serem informativos e organizados logicamente.

Mesmo assim, outro desafio para construir um corpus é o custo de anotação, principalmente quando se lida com textos brutos. Além dos investimentos em metodologia e expertise, como exemplificado no caso da aquisição da Scale IA pela Meta, há também discussões éticas envolvidas que podem atingir a imagem das organizações. Um caso de dilema ético ganhou destaque em meados de 2024, envolvendo a Meta e a Google. Na matéria veiculada, tornou-se público que ambas as empresas contrataram profissionais de países de baixa renda, subvalorizando o custo de mão de obra, para atuarem como rotuladores, ou *data taggers*. De acordo com a publicação, os profissionais são contratados para enriquecerem os conjuntos de dados de treinamento, atuando na curadoria e avaliando as respostas dos LLMs (28).

Deixando essa polêmica em segundo plano, uma alternativa para minimizar os custos de anotação proposta por alguns autores (29, 30) é a utilização de bases de dados externas (e.g. ontologias, vocabulários ou glossários[3]) apoiado por métodos supervisionados à distância (31). Embora esses trabalhos propostos possuam resultados promissores, eles ainda são limitados a domínios específicos e restritos a padrões de triplas ontológicas (sujeito, predicado e objeto) predeterminadas.

Outra discussão pertinente quando se trata de *fine-tuning*, envolve a utilização de ML de domínio amplo (genérico) e pré-treinados (específico). Em ML de domínio amplo (e.g. DistilBERT (32), RoBERTa (33), etc.), as empresas (e.g. Meta, Facebook, Hugging Face, etc.) instanciam o Bidirectional Encoder Representations from Transformers (BERT)(12) tradicional, submetem um corpus multidomínio e utilizam hiperparâmetros distintos para treinar o ML. No caso dos ML pré-treinados, geralmente o processo é similar, porém os ML são ajustados (*fine-tuning*) em alguns domínios específicos, como exemplo o BioBERT (21) na área biomédica e o SciBERT (34) na área científica.

Alguns estudos evidenciam que os ML pré-treinados possuem resultados superiores aos ML de domínio amplo, quando aplicados no contexto em que foram ajustados (35). Além disso, há trabalhos que defendem o uso de ML pré-treinados no idioma em que são aplicados em virtude deles possuírem resultados superiores aos ML multilinguais. Um exemplo disso é o próprio BERTimbau, que utiliza textos em língua portuguesa com base na Wikipedia (36). De forma semelhante, o FinBERT-PT-BR, que ajustou o BERTimbau, utilizando textos de notícias econômicas (37) e o LegalBERT-pt que utilizou textos do domínio jurídico brasileiro (38).

[2] Conjunto de princípios, conceitos e procedimentos expostos de forma integrada (1).
[3] Ambos também são classificados como Recursos Semânticos (RS).

Portanto, os ML, sejam eles de domínio amplo, pré-treinados ou ajustados a um domínio específico, são limitados ao conjunto de textos do corpus utilizado em seu treinamento, i.e., eles reconhecem entidades nomeadas previamente categorizadas e anotadas. Contudo, os seus dados podem ser estruturados em um KG, permitindo que os usuários realizem consultas, inferências e até enriquecimento de *datasets* (19).

1.1.2 Aplicação de tecnologias disruptivas no contexto militar

No contexto militar, a complexidade e a robustez das operações favorecem a implementação de tecnologias disruptivas para dar apoio às atividades das Forças Armadas (FAs), que abrangem dentro e fora do país. Ao redor do mundo, as FAs vêm investindo cada vez mais em Inteligência Artificial (IA) para apoiar a realização de suas atividades. No mundo ocidental, com destaque aos Estados Unidos da América (EUA), por exemplo, o Departamento de Defesa Americano (DoD) possui um histórico de inovações na utilização da IA no contexto da guerra.

Um deles é o Projeto Maven que há alguns anos se concentra em Visão Computacional para detecção de alvos e objetos de interesse através de imagens (39). Outro exemplo empreendido pelo DoD, em parceria com a empresa Scale IA, foi a adoção do Defense Llama[4], uma IA generativa baseada no Llama da Meta ajustada ao domínio militar, para apoiar missões de segurança nacional no planejamento de operações militares e de inteligência. Dentre essas iniciativas, ainda há outras como o protótipo QuantaIQ, que atua no apoio ao treinamento de pessoal, e o Hermes LLM[5] com foco no apoio ao planejamento militar.

No mundo oriental, as FAs chinesas apresentaram o Ernie Bot[6], em parceria com a Baidu Research, que atua na geração de previsões do comportamento humano no combate a fim de subsidiar decisões de seus comandantes (40). Além disso, os chineses já haviam apresentado o famoso chatBIT, da empresa Beijing Institute of Technology (BIT)[7], uma IA generativa que foi ajustada ao contexto militar a partir do chatGPT, utilizando dados da Enciclopédia Chinesa do Conhecimento e do Instituto de Tecnologia de Pequim, para atuar nos serviços de inteligência militar (41).

Em contrapartida, no Brasil, a utilização de IA no contexto da Defesa ainda é modesta. Em um estudo recente que avaliou o nível de produção científica de IA aplicada ao contexto da Defesa, no quinquênio de 2019 a 2024, o resultado demonstrou que o Brasil ocupa a vigésima colocação do ranking, alcançando 0,82% na produção científica mundial. Para efeitos de comparação, no mesmo ranking, juntos China e EUA lideram com 36%, representando, respectivamente, 19,25% e 16,66%, e são seguidos por Índia com

[4] https://scale.com/donovan/defense-llm
[5] https://nousresearch.com/hermes3/
[6] https://research.baidu.com/Blog/index-view?id=183
[7] https://english.bit.edu.cn/index.html

incríveis 10,38% e Reino Unido com 5% (42). Apesar da incômoda posição, nos últimos anos, o governo brasileiro publicou a Estratégia Brasileira de Inteligência Artificial (EBIA), instituído e alterado pelas Portarias números 4.617 e 4.979 de 2021(43, 44), com o objetivo de estabelecer ações estratégicas de implantação de IA no país (45). Em uma outra frente de trabalho do Ministério da Defesa (MD), é importante destacar o projeto Sistemas de Sistemas de Comando e Controle (S2C2)[8], financiado através do convênio FINEP-FAPEB 01.20.0272.00, com o objetivo de desenvolver um Sistema de Sistemas de Comando e Controle (S2C2) para atender às demandas de interoperabilidade da Família de Aplicativos de Comando e Controle da Força Terrestre (FAC2FTer).

No momento, as FAs possuem por volta de 350 mil pessoas atuando na ativa, incluindo militares e servidores civis. É o segundo maior contingente de pessoal no Governo Federal, ficando atrás somente do Ministério da Educação, com cerca de 455 mil servidores (46, 47). Segundo dados do Relatório de Gestão do MD, em 2024, as FAs possuíam um orçamento de 121 bilhões de reais, o qual é distribuído entre o próprio MD e as três FAs (Marinha, Exército e Aeronáutica). A princípio, esse orçamento parece ser vultoso, mas ao analisar a distribuição dos gastos no relatório, nota-se que 78,3% é destinado a pagamento de pessoal e 13,3% a custeio, sobrando somente 6,7% a investimentos (46). Pode até parecer controverso, contudo com esse percentual destinado a investimentos, realmente é difícil alavancar o desenvolvimento de IA no ambiente militar. Entretanto, ao analisar esse cenário sob outra perspectiva, surgem diversas oportunidades de realizar experimentos de pequeno porte através de projetos essenciais ao desenvolvimento das Forças Armadas (FAs).

Como as FAs são instituições permanentes de Estado, elas atuam na defesa da pátria e segurança nacional por meio de exercícios e operações militares, sejam eles singulares, conjuntos ou combinados com outros entes de dentro ou fora da Federação. Só no ano de 2024, foram mobilizados mais de 34 mil militares em operações que resultaram em 280 mil ações e apreensões de drogas, armamentos, explosivos, embarcações, veículos, aeronaves, minérios e dinheiro, além da detenção dos envolvidos (46). Como exemplo, em destaque a Operação Ágata[9], que se destina ao combate de crimes nas regiões de fronteira, e Operação Acolhida[10], voltada à assistência emergencial a refugiados e imigrantes venezuelanos. Embora haja um alcance nacional com as operações e exercícios, existe um problema relacionado ao pessoal da Força. Particularmente, as FAs são compostas por uma força de trabalho híbrida (militares, servidores civis e empregados públicos celetistas), distribuída por diversas carreiras e corpos de trabalho, sendo um dos desafios a manutenção de todo esse pessoal treinado e apto a desenvolver suas atividades.

[8] https://fapeb.com.br/s2c2-convenio-finep-fapeb-01-20-0272-00/

[9] https://www.gov.br/defesa/pt-br/assuntos/exercicios-e-operacoes/operacoes-conjuntas/operacao-agata

[10] https://www.gov.br/mds/pt-br/acoes-e-programas/operacao-acolhida

Um dos pilares das FAs é a capacidade de treinar o seu pessoal através de um conjunto robusto de Doutrinas Militares (DML), detalhada no Capítulo 2. Essas DML são textuais e distribuídas em diversos repositórios, como no acervo online do MD[11] e nas bibliotecas de cada Força Armada (FA)[12,13,14]. Mesmo assim, no contexto militar, um dos desafios de lidar com dados textuais para extrair conhecimento envolve a dificuldade de existir corpus ou conjuntos de dados voltados ao contexto, principalmente se os esforços forem voltados à constituição de um Large Language Model (LLM), treinado no domínio militar e que apoie as diversas atividades dentro da Força.

Portanto, no ambiente militar há inúmeras aplicações possíveis de LLMs, inclusive sobre dados reais de operações e exercícios. Mesmo sendo um caso de difícil acesso em virtude de políticas de sigilo e preservação estratégica (48, 29). Em contrapartida, é oportuno explorar o conteúdo das Doutrinas Militares (DML), dada a sua facilidade de acesso, estruturação e classificação de sigilo. Mesmo assim, ainda é necessária uma abordagem que seja capaz de obter conhecimento, explorando os textos dessas DML.

1.2 Caracterização do Problema

Nesta tese, o problema é exposto em dois níveis. O primeiro é de caráter geral e engloba características do cenário de aplicação, as quais são relevantes para definir as estratégias e avaliar as oportunidades durante o desenvolvimento do trabalho. Por sua vez, o segundo nível é específico e mais voltado à área de pesquisa, no caso, aos problemas de obtenção de conhecimento por meio do processamento de textos.

No domínio militar, o adestramento da tropa é condição necessária para o efetivo emprego da missão. No geral, com o avanço das tecnologias e serviços, cada vez mais é exigido um ciclo de aperfeiçoamento mais curto, com custos menores e maior abrangência. Entretanto, alguns fatores intrínsecos da atividade militar dificultam o cumprimento dessa missão. Como mencionado, um deles está associado à rotatividade de pessoal, inerente à atribuição do militar, e outro, um pouco mais complexo, se refere à redução de pessoal. Esse último, é uma realidade em função da Lei nº 13.954, de 16 de dezembro de 2019, que dispõe sobre a carreira dos militares e, por consequência, a redução do efetivo em um horizonte de curto prazo (49).

No que tange à pesquisa, os desafios de extrair conhecimento a partir de textos, envolvem algumas considerações. Uma delas é voltada à escassez de corpus anotados e ML ajustados ao contexto militar, principalmente em Língua Portuguesa. Outra consideração

[11] https://www.gov.br/defesa/pt-br/assuntos/estado-maior-conjunto-das-forcas-armadas/doutrina-militar
[12] https://www.marinha.mil.br/bibliotecadamarinha/
[13] https://bdex.eb.mil.br/jspui/
[14] https://www2.fab.mil.br/bibliotecaunifa/

envolve a limitação das abordagens de ajuste fino de ML nas tarefas de NER e RE em função das categorias de entidades e relações serem limitadas a um contexto específico. No sentido de enfrentar os desafios mencionados, a construção de um corpus deve conter categorias abrangentes que se baseiam em recursos semânticos do domínio. Além disso, como a anotação manual é custosa, a redução da intervenção humana pode agilizar a anotação e colaborar com a qualidade do corpus.

Além dessas considerações, a utilização de tarefas de PLN para reconhecimento de entidades nomeadas e suas possíveis relações dentro de um determinado domínio, mostra-se um caminho viável para obter conhecimento através do aprendizado realizado, buscando estabelecer correlações entre conceitos a fim de realizar inferências. Porém, como os textos utilizados nos ML são restritos, é possível estruturá-los em um KG a partir das interações com esses modelos. Com isso, é possível expandir a exploração das correlações, buscando novas inferências implícitas. Apesar dos esforços para contornar os desafios impostos, os resultados alcançados são limitados, restritos a alguns domínios de negócio e, na maioria dos casos, exploram um conjunto de entidades e relações restrito, como discutido nos trabalhos apresentados no Capítulo 3.

Portanto, caracterizados ambos os problemas, geral e específico, como gerar Knowledge Graph (KG) a partir de documentos doutrinários apoiados por um Modelo de Linguagem (ML) ajustado, cujas aplicações podem ser realizadas no contexto militar, permitindo que ambos os artefatos deem suporte à construção de Modelos de Domínio (DM)?

1.3 Questões de Pesquisa

Na seção anterior, os problemas levantados envolvem considerações acerca dos desafios da obtenção de conhecimento. Apesar de alguns caminhos vislumbrados, há algumas lacunas que podem ser melhor elucidadas e relevantes ao contexto da pesquisa. Nesse aspecto, é importante determinar um conjunto de questões de pesquisa, buscando dar materialidade ao objeto em estudo. Em suma, diante dos problemas expostos, são levantadas as seguintes questões de pesquisa:

- **Q1**: Como gerar um KG a partir de textos doutrinários no contexto militar utilizando um ML?
- **Q2**: Como gerar modelos de domínio a partir de textos no contexto militar, combinando o uso de artefatos das abordagens Data-Driven (DD) e Theory-Driven (TD)?

As questões de pesquisa são alinhadas aos problemas e detalham as indagações que o trabalho deve responder. Cada questão de pesquisa é respondida por meio de uma

ou mais hipóteses, observando o problema, o escopo e as limitações impostas, como será explorada na seção 1.4.

1.4 Hipótese

Nesta tese, são formuladas hipóteses que buscam responder às questões de pesquisa apresentadas na seção anterior que envolvem desde a criação de corpora anotados a partir de textos doutrinários no contexto militar, incluindo o ajuste fino de um ML, a geração do KG e suas aplicações. Dada a multiplicidade das questões de pesquisa, a seguir são apresentadas as hipóteses pormenorizadas:

- **H1**: Um método baseado em um metamodelo e com apoio de um ML ajustado no contexto militar, é capaz de gerar um KG a partir de textos doutrinários.
- **H2**: Um metamodelo que permite metacategorizar as entidades e relações pode flexibilizar a anotação de um corpus para o ajuste fino de um ML nas tarefas NER e RE.
- **H3**: Um metamodelo combinado com uma pré-anotação heurística e Recursos Semânticos (RS) aplicado a um corpus, gera ML ajustados cujas métricas de avaliação são equiparáveis ao estado da arte.
- **H4**: Uma abordagem híbrida que explore os dados em um KG, gerados a partir da submissão de textos a um ML ajustado no contexto militar, apoia a construção de modelos de domínio similares aos gerados por uma abordagem orientada à teoria.

1.5 Objetivo

Esta tese tem como objetivo geral especificar uma abordagem híbrida para apoiar a construção de um Modelo de Domínio (DM). Essa abordagem é apoiada por um Knowledge Graph (KG) gerado através de um Modelo de Linguagem (ML) ajustado, em especial, nas tarefas de NER e RE. O corpus utilizado no ajuste do ML tem como base os dados textuais não-estruturados de documentos doutrinários do contexto militar. Esse objetivo geral é dividido nos seguintes objetivos específicos:

- **O1**: Especificar um metamodelo flexível com metacategorias genéricas para apoiar o processo de ajuste fino de um ML no contexto militar.
- **O2**: Construir um corpus anotado no contexto militar de acordo com os construtos definidos no metamodelo ou combinando com outros Recursos Semânticos (RS).

- **O3**: Especificar um processo de ajuste fino de um ML, voltado às tarefas NER e RE, utilizando o corpus anotado no contexto militar.
- **O4**: Especificar um processo de triplificação para gerar o KG, baseado em um grafo RDF, que permita a realização de operações sobre os seus recursos.
- **O5**: Especificar um mecanismo para pré-anotar um corpus a partir de regras de expressão regular, explorando heurísticas baseadas em padrões textuais do contexto militar.
- **O6**: Especificar um mecanismo de apoio à elaboração de modelos de domínio a partir de textos combinando o ML ajustado com a exploração do KG.
- **O7**: Desenvolver um protótipo baseado no cenário de aplicação para realização de experimentos que permita validar e avaliar a utilidade da abordagem.

1.6 Justificativa

Nesta seção, são apresentados alguns pontos destacados nos documentos referenciais dos órgãos de Governo que se alinham, ou melhor, que inspiram o desenvolvimento desta tese. Além disso, são mencionadas algumas especificidades do cenário de aplicação que colaboram com a justificativa do trabalho.

A Política Nacional de Defesa (PND) estabelece oito Objetivos Nacionais de Defesa (OND), descrevendo-os com base na análise dos ambientes internacional e nacional, além da concepção política dos países (50). Alinhados a este trabalho, destacam-se os OND: a) Assegurar a capacidade de Defesa para o cumprimento das missões constitucionais das FAs; e b) Promover a autonomia tecnológica e produtiva na área de Defesa. O primeiro evidencia as capacidades necessárias que as FA necessitam para realizar suas funções amparadas na Constituição Federal (CF). E o segundo estimula a pesquisa e a busca do desenvolvimento de tecnologias autóctones nas mais críticas áreas de Defesa. Dessa forma, busca-se com este trabalho colaborar com ambos os OND no que tange ao desenvolvimento de uma abordagem para contribuir com o incremento do adestramento da tropa, utilizando soluções disruptivas aplicadas no contexto militar.

O Ministério da Ciência, Tecnologia e Inovações (MCTI), por intermédio da Portaria Nº 1.112 de 19 de março de 2020, em seu artigo 4º, fomenta o uso de IA como área habilitadora em projetos de pesquisa, de desenvolvimento de tecnologias e inovações no Governo Federal, principalmente por contribuir com o processo de inovação e conhecimento científico e tecnológico (51). Aliada à portaria, o MCTI publica a Estratégia Brasileira de Inteligência Artificial (EBIA), voltada a projetos de pesquisa, de desenvolvimento de tecnologias disruptivas e inovações, principalmente entre entes do Estado, buscando

interoperabilidade entre diversos conceitos, dados e aplicações (52). Sendo assim, este trabalho se alinha aos normativos, já que propõe o desenvolvimento de uma abordagem que utiliza ML com ênfase em soluções flexíveis e disruptivas.

Adicionalmente, este trabalho também observa alguns pontos mencionados na seção 1.2, principalmente associados às especificidades do cenário de aplicação, que justificam a necessidade de desenvolvimento. O primeiro desafio é a rotatividade de pessoal que afeta as FAs. Apesar de inerente à atividade militar, é necessário que os treinamentos sejam contínuos e efetivos. Outro ponto é que a redução de pessoal imposta pela Lei nº 13.954, de 16 de dezembro de 2019, pode impactar a continuidade do serviço no médio e longo prazo. Finalmente, também podemos mencionar a ampla dimensão de documentos doutrinários e a comunidade de militares aptos para treinamentos. Portanto, este trabalho visa contribuir para a geração de um KG a partir de um ML ajustado ao contexto militar. Esses grafos podem ser armazenados em repositórios de dados e interligados a novos domínios. Além disso, eles podem ser consumidos por aplicações que apoiam as atividades de treinamento de pessoal, tornando-as mais simples e com menor dependência de intervenção humana. Inclusive, podem apoiar analistas na construção de modelos de domínio tendo em vista a sua capacidade semântica e de inferência dos recursos explorados no KG.

No âmbito das FAs, este trabalho se alinha com as ações estratégicas tanto da Marinha do Brasil (MB), principalmente por aplicar IA no contexto militar. Na MB, por exemplo, o Planejamento Estratégico (PEM 2040), estabelece a Ação Estratégica Naval (AEN) – Defesa 2: "Implantar a Defesa Proativa da Amazônia Azul que define a implantação de um sistema proativo de Comando e Controle (Comando e Controle (C2)), incorporando inovações militares – cinéticas e não cinéticas – além de novas tecnologias de Ciência de Dados, como Inteligência Artificial" (53).

No contexto da pesquisa científica, em PLN, a execução de tarefas de NER e RE necessita de conjuntos de dados anotados para obter êxito em seus resultados. Nos casos em que não há conjuntos de dados anotados, buscam-se corpus de textos para realizar a anotação. Nesse meio tempo, diversas abordagens e métodos foram propostos com objetivo de reduzir a anotação manual. Entretanto essas abordagens compartilham da limitação de predefinir as categorias ou rótulos de classes de anotação para domínios específicos (9, 54, 55, 56). Por outro lado, algumas pesquisas apontam a necessidade de haver categorias de propósito geral que possam ser aplicadas em outros domínios (57). Entretanto, as pesquisas mais recentes indicam que as abordagens de classes de propósito geral ainda podem ser evoluídas e se tornarem mais abrangentes (18, 58).

Por fim, esta tese se alinha também ao projeto Sistemas de Sistemas de Comando e Controle (S2C2)[15] tendo em vista que o processo de geração de KG proposto por este trabalho pode ser a base para apoiar a construção de modelos de domínio, sugerindo

[15] https://fapeb.com.br/s2c2/

perspectivas de entidades e relações que podem enriquecer ontologias, incrementando a interoperabilidade semântica entre os sistemas de C2. Além disso, o ML ajustado pode ser utilizado em aplicações e sistemas a fim de comparar os resultados com outros ML. Com isso, pode-se contribuir com o objetivo do projeto Sistemas de Sistemas de Comando e Controle (S2C2) que visa desenvolver pesquisas para o aprimoramento da interoperabilidade semântica de Comando e Controle das Forças Armadas Brasileiras. Esse projeto possui convênio via financiamento FINEP-FAPEB 01.20.0272.00 – Sistema de Sistemas de Comando e Controle.

1.7 Metodologia

Como apontado na seção 1.5, o objetivo desta tese é gerar um KG de acordo com algumas especificidades. Para alcançar este objetivo, é necessário estabelecer uma metodologia que deve adotar o paradigma iterativo e incremental (59). Dessa forma, os requisitos não são estabelecidos todos de uma única vez. Então, a cada questão de pesquisa, os requisitos estabelecidos são investigados. A medida que avança o entendimento do negócio, novos requisitos podem ser incluídos ou alterados.

Em seguida, as hipóteses são desenvolvidas e a busca de trabalhos relacionados situam como as abordagens do estado da arte investigaram aquela questão de pesquisa. A depender do tamanho da hipótese, o paradigma de divisão e conquista (60) é adotado com o intuito de fragmentar em hipóteses menores e favorecer a sua implementação e gerenciamento. Assim, alinhada às hipóteses, pequenos experimentos são implementados, gerando um conjunto de artefatos (e.g. modelos, código-fonte, conjuntos de dados, etc.) e seus resultados são avaliados, nos casos em que são comparáveis, de acordo com os trabalhos relacionados.

Os artefatos gerados nos experimentos são testados e organizados em um repositório público com o passo a passo de sua implementação em conjunto com os testes e os resultados finais. Assim, a metodologia deve contemplar processos que geram artefatos baseados em técnicas estabelecidas. Além disso, cada produto ou artefato deve ser submetido à avaliação para análise dos seus resultados.

Na primeira etapa do trabalho, são realizados levantamentos de corpora de textos do contexto de C2, anotados ou não, para servirem de insumos no treinamento do ML. Além disso, serão utilizados documentos doutrinários e Recursos Semânticos (RS) do domínio disponíveis como fontes de dados para composição do corpus. Na segunda etapa, concomitante à primeira, serão levantados os trabalhos do estado da arte na geração de KG. Preferencialmente, esses trabalhos devem envolver o uso de técnicas de PLN, com ênfase nas tarefas de NER e RE. Ademais, serão levantadas abordagens de extração de relações e ML que deem suporte a essas tarefas.

No que concerne à implementação, são levantados ambientes de desenvolvimento, linguagens, ferramentas de armazenamento, versionamento de código-fonte, bibliotecas de software, dentre outros. Preferencialmente, serão utilizadas ferramentas *open source* ou que tenham custos razoáveis. Além disso, essas ferramentas devem atuar de forma integradas na nuvem ou em repositórios na Web. Dessa forma, serão implementados algoritmos por meio de experimentos. Cada experimento visa identificar e testar arquiteturas que contribuam com a pesquisa. Para lidar com a extração de textos dos documentos, por exemplo, serão levantadas bibliotecas destinadas a lidar com textos de diferentes formatos.

A terceira etapa destina-se a especificar o processo da abordagem proposta. Nele, devem conter atores, fluxos de processos e atividades, fontes de dados de entrada e saída e conexões com outros processos. Alguns experimentos devem ser desenvolvidos para fazer pequenas validações do processo proposto. É previsto que nesses experimentos incluam proposições utilizadas em outros trabalhos para efeito de comparação direta. Além disso, é previsto um experimento envolvendo pessoas com o intuito de coletar um número razoável da validações do processo da abordagem proposta.

Por fim, a quarta etapa envolve a especificação de métodos de validação dos artefatos produzidos. Desse modo, a validação deve cobrir três aspectos essenciais: i) preferencialmente devem ser utilizados dados do contexto de C2; ii) artefatos produzidos devem estar disponíveis em plataformas de acesso público; iii) principais resultados obtidos devem ser comparáveis aos resultados de outros trabalhos relacionados por meio de métricas qualitativas e quantitativas.

1.8 Organização do trabalho

Este trabalho está dividido em sete capítulos. No Capítulo 2, é apresentada a fundamentação teórica com ênfase na extração de informações a partir de textos, modelo de linguagem, modelagem de dados conceitual, metamodelagem, grafos de conhecimento e abordagens DDs e TDs. O Capítulo 3 contém os trabalhos relacionados. No Capítulo 4, é apresentada a especificação da abordagem IDEA-C2. O Capítulo 5 descreve a implementação da abordagem IDEA-C2. No Capítulo 6, encontram-se as descrições dos experimentos realizados e seus resultados. Finalizando, o Capítulo 7 contém algumas considerações finais e indicações de trabalhos futuros.

2 CONCEITOS BÁSICOS

Com a finalidade de situar os objetivos da pesquisa, foi conduzida uma revisão de diferentes áreas do conhecimento vinculadas ao tema, visando à fundamentação de conceitos e à identificação de características e capacidades comuns.

Ao lidar com textos não-estruturados há alguns desafios, principalmente quando é necessário extrair informações úteis. Em Processamento de Linguagem Natural (PLN), existe a técnica de Extração de informação (EI) que permite obter informações estruturadas a partir desses textos (seção 2.1). Além disso, a depender do tipo de informação a ser extraída, há um conjunto de tarefas específicas que são utilizadas de acordo com a análise textual pretendida (léxica, sintática e semântica). Na análise léxica, pequenas porções do texto podem ser separadas em palavras ou sentenças (tokenização). Elas também podem ser reduzidas (lematização), ou terem identificadas as classes gramaticais (*pos tagging*) ou as suas entidades nomeadas reconhecidas (NER). A análise sintática foca na estrutura gramatical através da tarefa de *parsing* para análise de dependências, ou *chunking* para sintagma nominal, dentre outras. Finalmente, a análise semântica visa entender o significado contido no texto através de tarefas, como Extração de relação (RE), análise de sentimentos, dentre outras (61). Neste trabalho, serão detalhadas as tarefas de NER e RE, respectivamente, nas subseções 2.1.1 e 2.1.4.

Considerando que as tarefas NER e RE são aplicadas sobre textos, é importante que haja corpora anotados ou também conhecido como *dataset*. Na ausência de corpora, é necessário preparar uma infraestrutura que pode envolver conhecimentos distintos, incluindo pessoas, aqui representadas através de papéis (e.g. anotadores e curadores) e um conjunto de dados relevante ao negócio, como detalhado na subseção 2.1.2. Entretanto, como a anotação é custosa, é possível gerar corpora de forma semi-automática, explorando bases de dados externas, através da técnica de supervisão à distância (31), detalhado na subseção 2.1.3.

Os Modelos de Linguagem (ML) ou Large Language Models (LLMs) de domínio amplo ou pré-treinados são caracterizados por serem treinados com corpora volumosos e com conteúdo abrangente. O LLaMA-65B[1] é um exemplo de LLM de domínio amplo que foi pré-treinado com 1.4 trilhão de palavras (62). Outro exemplo de LLM de domínio amplo é o Bidirectional Encoder Representations from Transformers (BERT), que vem se destacando nos últimos anos e alcançando resultados promissores. Os ML podem obter respostas mais úteis quando aplicados em um contexto específico através do fine-tuning. Contudo, o fine-tuning requer um corpus anotado e categorizado com textos do domínio

[1] https://huggingface.co/huggyllama/llama-65b

em que ele será aplicado a fim de executar tarefas de NER e RE, como detalhado na seção 2.2.

Apesar dos LLMs ajustados ao contexto expressarem a semântica de um domínio, eles são caixa-pretas e encapsulam o raciocínio de suas respostas. Contudo, toda a massa de dados utilizada no ajuste fino, assim como as inferências obtidas através das tarefas de NER e RE podem ser armazenadas em um repositório de dados. No entanto, o repositório deve ser constituído de uma estrutura prévia para comportar os dados. A modelagem conceitual é um mecanismo apropriado para estruturar os dados através de entidades e relacionamentos (63), como detalhado na seção 2.3. Embora haja diferenças entre os tipos de entidades e relacionamentos obtidos por meio da extração de informações e aqueles definidos na modelagem conceitual, ainda é possível empregar um modelo capaz de prover a estrutura necessária para o armazenamento desses dados.

Em textos expressos em linguagem natural, há dificuldade em antecipar a forma como o conteúdo será formulado, o que pode resultar no desenvolvimento de diferentes formas de estruturação, em função das variadas representações dos dados. Ao utilizar um metamodelo, por exemplo, é possível definir uma estrutura robusta e de alto nível capaz de lidar com as diversas formas de representação de dados textuais, como detalhado na seção 2.4. Entretanto, um metamodelo requer uma abordagem de representação de conhecimento que consiga explicitar seus níveis de abstração (metaesquema, esquema e instância), como por exemplo um Knowledge Graph (KG).

Um KG permite a representação de um metamodelo utilizando as mesmas estruturas que representam o modelo e as suas respectivas instâncias em uma só visão. Dado o seu grau de flexibilidade, as ontologias também podem ser incorporadas, permitindo o enriquecendo semântico dos recursos contidos no KG, como detalhado na seção 2.5. Finalmente, o trabalho também explorou os conceitos do ambiente militar, em especial, ao Comando e Controle (C2) por ser o domínio de aplicação desta tese, como detalhado na seção 2.6.

2.1 Extração de Informação a partir de dados textuais não-estruturados

Como mencionado no Capítulo 1, os ML para realizar tarefas de PLN se baseiam em compreensão da linguagem natural e foram inspirações de pesquisas das áreas da Filosofia e Linguística com o objetivo de representar o conhecimento a partir de um raciocínio. O PLN é uma das subáreas de IA de grande complexidade em virtude da compreensão da linguagem natural exigir uma investigação empírica por meio da prática de experimentos do comportamento humano (64). Inicialmente, o PLN era utilizado para lidar com problemas que envolvem perguntas e respostas, tradução automática, elaboração de resumos, análise sintática e de sentimentos (65). Porém, mais tarefas foram inseridas

no contexto de PLN através da técnica de Extração de informação (EI), como: avaliação de similaridade semântica e classificação de documentos (13).

A EI (*Information Extraction*) é uma técnica que permite extrair dados de textos (estruturados, semiestruturados e não estruturados) com base na busca de ocorrências de uma classe específica de objetos. Tarefas associadas à EI são capazes de identificar e reconhecer entidades nomeadas ou conceitos (NER do inglês *Named Entity Recognition*) e relações, inclusive semânticas, entre essas entidades, denominada extração de relações (RE do inglês *Relation Extraction*). Comumente, são exemplos de NER: pessoas, locais, nomes de organizações, drogas, doenças, conceitos, etc. Enquanto que a RE representa "quem fez algo" ou "qual sintoma é causado por uma determinada doença", dentre outros (61). A Figura 1 exemplifica a identificação de entidades nomeadas e relações a partir de um texto retirado da Doutrina Militar Terrestre, a qual discorre sobre o conceito de ambiente operacional (66). Essa figura é explorada em detalhes nas subseções 2.1.1, 2.1.2 e 2.1.4.

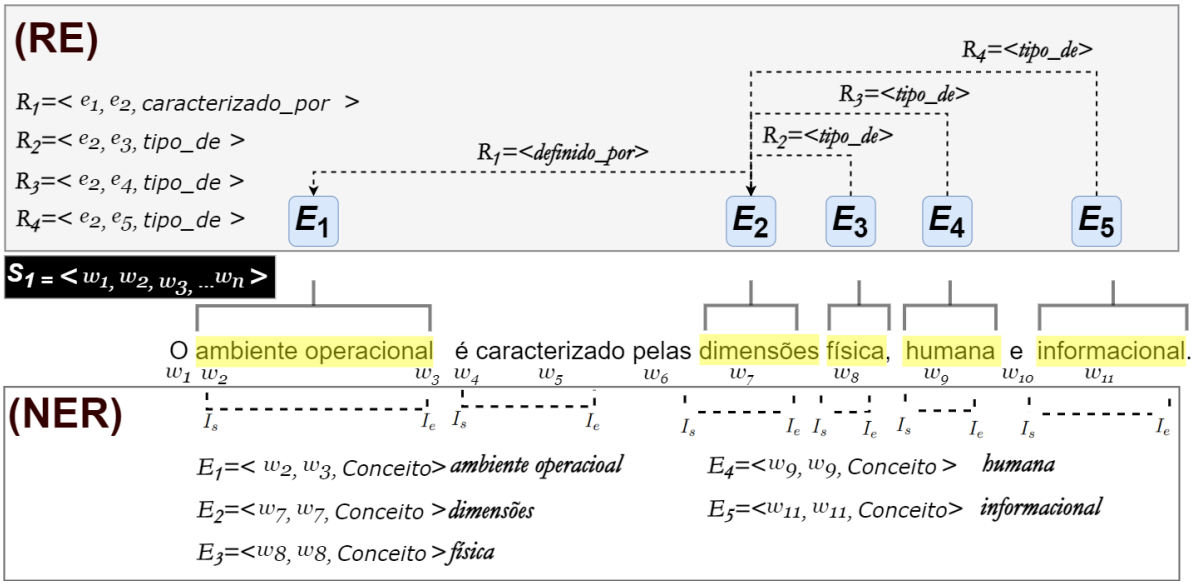


Figura 1 – Exemplificação de NER e RE. Adaptado de Li et al.(67)

2.1.1 Entidades Nomeadas

As entidades nomeadas são formadas por uma sequência de *tokens*, definidas por $s = \langle w_1, w_2, ..., w_N \rangle$, com o objetivo de gerar um conjunto de tuplas que representa uma entidade nomeada em s, definido por $\langle I_s, I_e, E_j \rangle$, onde $I_s \in [1, N]$ e $I_e \in [1, N]$, que são os índices inicial e final de uma entidade nomeada, e E_j representa o tipo de entidade genérico identificado na sentença. Cabe ressaltar que o tipo de entidade também pode ser definido por um conjunto específico de categorias, por exemplo: pessoa, lugar, etc. As RE expressam a semântica da relação entre as entidades nomeadas, permitindo a identificação de suas ações (67). A partir da exploração das RE, por exemplo, é possível construir grafos de conhecimento (68), como será abordado adiante.

A Figura 1 exemplifica a identificação tanto de entidades nomeadas quanto de relações em uma sentença. No caso da NER, as cinco entidades nomeadas foram reconhecidas na sentença, identificadas através de onze *tokens* (palavras), onde w representa cada *token*. Os rótulos utilizados para identificar as relações nesse exemplo, foram concebidos a partir do identificação das entidades nomeadas. Apesar de ser possível conduzir as tarefas de NER e de RE por meio de ML independentes, a segunda pode ser dependente do produto da primeira, já que as relações são expressas em termos das entidades que dela participam como sujeito e objeto, como observado no exemplo da figura.

Desde a década de 1970, as técnicas de EI sobre dados textuais vêm sendo estudadas, explorando métodos baseados em regras gramaticais formais (61). Porém, esses métodos limitam mudanças de domínio ou até o estilo do texto. Assim, a cada alteração é necessária a mudança da regra (69, 70). Mais tarde, surgiram trabalhos baseados em aprendizado supervisionado. Nesse tipo de abordagem, um conjunto de textos é anotado, principalmente por especialistas do domínio, com o objetivo de criar exemplos rotulados a serem utilizados no treinamento de modelos de linguagem. A anotação do texto tem como objetivo identificar qual a classe de entidade um determinado termo na sentença representa (71, 64). Há também abordagens que utilizam parte de conjunto de entrada anotada e outra não anotada. Esse tipo de abordagem é denominada semi-supervisionada. Por fim, há abordagens de aprendizado não-supervisionado que são caracterizadas por não utilizar bases não anotadas. Nesse caso, não há o dispêndio da anotação nem da criação de regras por utilizar relações sem definir um domínio específico (71, 64).

2.1.2 Anotação de Texto

Como vimos, as EI supervisionadas necessitam de exemplos anotados para o treinamento do ML que abrange processos de anotação, análise e aprovação ou desaprovação, envolvendo papéis especializados para anotar e realizar curadoria. Os curadores são responsáveis por aprovar ou não o texto anotado, geralmente exercido por um especialista no domínio. A anotação é apoiada por ferramentas especializadas que atuam a partir da submissão de um conjunto de textos. O objetivo dessas ferramentas é produzir um conjunto de dados rotulado para treinamento de ML. Algumas ferramentas são *open-source* e outras requerem licenciamento. Elas oferecem um conjunto de funcionalidades que permitem aos usuários definirem as categorias ou rótulos e relações identificadas nas sentenças de textos a serem anotados. São exemplos de anotadores de texto: Brat Rapid Annotation Tool (BRAT) (72), UBIAI Text Annotation Tool (UBIAI) (73), Label Studio: Data Labeling Software (74), Doccano (75) e Prodigy (76).

Na Figura 2, na parte superior é ilustrada a anotação de um texto de C2. Enquanto que na parte de inferior, é ilustrada a saída gerada pela ferramenta no formato JSON Lines (JSONL). Nela, são apresentadas as categorias definidas das entidades e relações. Para

Figura 2 – Anotação de texto no Doccano e exportação no formato JSONL. Imagem do Autor.

anotar as entidades são utilizadas cinco categorias, a saber: “ação_tática”, “tipo_de_ação”, “local”, “meio_naval” e “tipo_de_meio”. No caso das relações, foram utilizadas quatro categorias, a saber: “tipo_de”, “realizada_em”, “executado_por” e “usado_para”. Algumas categorias de relação tem caráter genérico, como no caso de “tipo_de” que faz alusão à relação de hiperonímia e hiponímia[2]. As categorias de entidades, geralmente, são restritas ao domínio, como nos casos de “ação_tática” e “meio_naval”. Por outro lado, pode haver categorias de entidades genéricas, como no caso de “local”.

Finalmente, em relação à saída, a ferramenta gera um arquivo no formato JSON Lines (JSONL) (77) que armazena os dados de modo estruturado e cada linha do arquivo representa um objeto JSON com exemplo anotado. A ferramenta especifica cada texto anotado de acordo com a sua delimitação posicional, definido por ***start_offset***, destacado em verde, que marca o início, e ***end_offset***, destacado em amarelo, que marca o fim. Esse tipo de anotação se assemelha ao formato IOB (*Inside-Outside-Beginning*), em que o **B** representa o início (*begin*) de uma entidade, o **I** (*in*) a continuação dela, e o **O** (*out*) representa que a palavra não pertence à entidade (61). Além disso, a ferramenta divide a geração do arquivo em três seções principais. A primeira é a ***text***, que retorna o texto completo, destacado em laranja. A segunda é a ***entities***, que corresponde às entidades anotadas, destacado em azul. Por fim, a ***relations*** que carateriza a relação binária entre as entidades, destacado em vermelho.

[2] Hiperonímia e hiponímia são relações semânticas hierárquicas entre palavras.

2.1.3 Abordagem supervisionada à distância

Como mencionado, as abordagens de Aprendizado de Máquina (AM) sejam elas, supervisionadas ou não, utilizam ML para realizar tarefas variadas que podem ser preditivas ou descritivas. Nas tarefas preditivas ou supervisionadas, o objetivo é detectar um modelo ou hipótese a partir dos dados de treinamento. Nesse caso, é comum a aplicação de algoritmos de classificação e regressão. Por outro lado, as tarefas descritivas ou não supervisionadas são caracterizadas por explorar conjuntos de dados. Por exemplo através de algoritmos de *agrupamento* de acordo com a similaridade dos atributos. Além disso, as tarefas descritivas podem utilizar a *associação* para encontrar padrões frequentes entre atributos ou a *sumarização* para compactar dados (71, 64).

No contexto da abordagem supervisionada, os conjuntos de dados geralmente são divididos aleatoriamente em três subconjuntos distintos: treinamento, validação e testes (71). Os conjuntos de treinamento e validação são utilizados na predição do modelo. O conjunto de teste é utilizado para avaliar o modelo gerado. As tarefas de treinamento do modelo de linguagem recebem como entrada um conjunto de treinamento com N pares de exemplos de entrada e saída $(x_1, y_1), (x_2, y_2), ...(x_n, y_n)$, onde cada x_n pode ser uma sentença de documento e y_n é gerado por meio de uma função ou modelo definido por $y = f(x)$, isto é, y representa o valor da predição. O objetivo é descobrir uma função h, que é uma hipótese possível que se aproxime da função f, a partir da avaliação dos dados do conjunto de testes, distinto do conjunto de treinamento. Dessa forma, consegue-se avaliar a precisão do modelo, ou seja, se ele conseguiu aprender ou não (64).

Como vimos, em PLN é comum utilizar a abordagem supervisionada em função da experiência do curador no domínio da aplicação para informar ou avaliar exemplos do negócio que sejam úteis para o treinamento do ML. Além disso, a abordagem supervisionada requer um corpus de textos anotado e volumoso com categorias ou rótulos de entidades e relações previamente definidas, as quais expressam a semântica das relações entre as entidades nomeadas no texto. Contudo, corpora abrangentes e volumosos são escassos para certos domínios, em alguns casos até inexistentes. Outro fato relevante é que a atividade de anotação manual de textos é onerosa tanto em função do tempo gasto quanto o custo financeiro de sua produção (78).

Há décadas, são produzidas coleções de dados em organizações que foram construídas a partir de modelos de dados, ontologias, glossários e taxonomias identificadas com o negócio. Esses conjuntos de dados são variados e possuem estruturas de representação distintas, cujos dados podem ser usados para a anotação de textos, e assim colaborar com o treinamento do modelo. Dessa forma, a supervisão à distância é uma abordagem que busca utilizar esses conjuntos de dados de modo a enriquecer o treinamento do ML. A partir dessa abordagem, é possível criar estratégias de extração baseadas, por exemplo, em regras heurísticas. A elaboração dessas regras tem como objetivo fornecer exemplos

variados de textos anotados ao treinamento. Além disso, caracteriza-se em uma abordagem semi-supervisionada, já que o papel do especialista do domínio na atividade de anotação é amenizado, propiciando avaliações mais ricas e ajustes finos dos exemplos com maior qualidade e volume (31).

As técnicas de supervisão à distância (*Distance Supervision*) foram incorporadas ao estudo de AM com o objetivo de gerar categorias de treinamento baseada em heurísticas calculadas de acordo com os dados de uma base de dados externa. Essa técnica surgiu através da extensão dos trabalhos de extração de relações entre pares de entidades nomeadas em sentenças de texto, explorando as relações entre hiperônimos do tipo "é-um" (*is-a*) (78). No trabalho de Mintz et al.(31), é explorado um exemplo resumido, considerando a relação **localização-contém** a partir dos pares de instâncias expostos no texto: "Virgínia" e "Richmondi", onde é possível extrair características contidas nas sentenças de textos entre esses pares, como exemplo "Richmond, a capital da Virgínia". Além disso, recentemente, bibliotecas de larga escala, como o Snorkel (79) e SPEAR (80), são exemplos de implementação da técnica de supervisão à distância.

Basicamente, as implementações baseadas na abordagem supervisionada à distância permitem que desenvolvedores de software codifiquem funções de rotulagem, *@labeling_function()* ou simplesmente Labeling Function (LF), arbitrando heurísticas com base em termos associados aos pares de entidades, ilustrado na Figura 3 (79, 80). Como pode haver um número elevado de sentenças de um par de entidades e, por consequência, extrair um grande número de características, há possibilidade de ruídos ou falsos-positivos (31). O Snorkel, uma biblioteca desenvolvida em Python de código aberto, que a partir dos pesos modela probabilidades de classes de acordo com as LF. Nele, para contornar o ruído, pode ser implementado um classificador de regressão logística a partir das classes de cada LF, atribuindo um peso para cada regra. De acordo com os autores, em comparação com a anotação manual, estudos apontam que ao utilizar a supervisão à distância é possível construir modelos de treinamento 2,8 vezes mais rápido, com quase 50% de aumento de desempenho na predição (79).

Na Figura 3, é ilustrado o exemplo de implementação de uma LF, utilizando a biblioteca do Snorkel, através de dois *frames*. No *frame 1*, é destacado o *decorator @labeling_function()* que cada função de rotulagem deve utilizar. O *decorator* pode ser aplicado a qualquer função python que retorne um rótulo para um único ponto de dados. A LF **lf_keyword_acao_definido_por(x)** implementa uma regra baseada em um conjunto de palavras, comumente utilizadas em definições de termos, com objetivo de encontrar essa regra no texto fornecido. No *frame 2*, é ilustrado o resultado após a execução de um conjunto de LF implementadas. Em destaque está o resultado da regra **lf_keyword_acao_definido_por(x)**, onde pode ser observado que houve três ocorrências, conforme registrado na coluna ***j***.

1
```python
@labeling_function()
def lf_keyword_lookup_relacao_definido_por(x):
    keywords = ["é a", "é o", "são as", "são os", "com o propósito de", "é um"]
    return OK if any(word in x.sentences.lower() for word in keywords) else ABSTAIN
```

2

	j	Polarity	Coverage	Overlaps	Conflicts
lf_keyword_lookup_uma_entidade	0	[1]	0.125	0.000	0.0
lf_keyword_lookup_entidades	1	[1]	0.625	0.125	0.0
lf_regex_contains_resposavel	2	[]	0.000	0.000	0.0
lf_keyword_lookup_relacao_definido_por	3	[1]	0.125	0.125	0.0

Figura 3 – Exemplo de *@labeling_function()* utilizando o Snorkel. Imagem do autor.

Outros trabalhos exploram a abordagem supervisionada à distância por meio de fontes de dados baseadas em ontologias, como nos casos de Fries et al.(30) e Dong et al.(81). As ontologias possuem vocabulários ricos e pré-definidos que representam o conhecimento de um domínio particular. Esse conhecimento e sua interpretação são incorporados às ontologias de modo explícito, fornecendo regras para interpretar os conceitos. Inclusive, as ontologias podem ser combinadas a outras ontologias através de alinhamento (82).

As ontologias podem ser representadas em RDF através de triplas ou recursos, denotada por $T(i) = (e_1, r_i, e_2)$, formadas por sujeito, predicado e objeto. As entidades de informação são representadas pelo sujeito e objeto, aqui expressos respectivamente por e_1 e e_2. A semântica existente entre as entidades é expressa por meio da relação ou predicado, denotado por r_i. Assim, ao considerarmos um corpus qualquer não rotulado, $C = (s_1, s_2...s_n)$, constituído por n sentenças, pode-se utilizar as triplas para anotação a partir da identificação das entidades e as relações nas sentenças sn heuristicamente.

Embora a abordagem supervisionada à distância ofereça inúmeras vantagens sobre a rotulação manual, devem ser levados em consideração a questão do ruído, o custo de processamento das LF e a necessidade do especialista do domínio fornecer regras que são implementadas por cada LF. Cada biblioteca derivada da abordagem supervisionada à distância busca implementar rotinas de tratamento específicas para lidar com o ruído (80). Contudo, não é objetivo deste trabalho discutir as estratégias de tratamento de ruído tampouco aprofundar sobre o uso dessas bibliotecas.

2.1.4 Extração de Relações

Os estudos acerca da tarefa NER são amplamente difundidos e baseados em textos anotados. A atividade de anotação segue alguns paradigmas e determina a forma com que os textos são identificados. Porém, como a extração de relações envolve aspectos semânticos, os desafios são maiores e demandam pesquisas robustas (67). Como abordado

na seção 1.1, a arquitetura *Transformer* ganhou notoriedade através do mecanismo de atenção, que foi introduzido pelo artigo "*Attention is all you need*" (83). Essa publicação revolucionou as tarefas de PLN, permitindo a identificação da existência de relações entre entidades distantes no texto (84).

Desde então, houve alguns avanços relevantes na tarefa de RE, envolvendo métodos e estratégias de implementação. Em publicação recente, no trabalho de Ali et al.(58), uma revisão abrangente foi proposta acerca de extração de relações, principalmente no contexto multilingual, comparando desde métodos clássicos baseados em *features* até técnicas avançadas implementadas com LLMs. Entretanto, os autores destacam o papel da técnica de supervisão à distância combinado com técnicas de *multi-instance learning*, *translation projection* e ajuste fino em LLMs para reduzir os problemas de ruídos. Além disso, como os estudos acerca de tarefas de RE são extensos, algumas abordagens são brevemente apresentadas a seguir, com destaque à: *Sentence-Level RE*(85), *Document-Level RE*(86), *Few-Shot RE*(54), *Zero-Shot RE* (87) e *Definition Extraction*(88).

No trabalho de Han et al.(85), é apresentando o Sentence-Level Relation Extraction ou *Sentence-Level RE*, uma tarefa de identificação e classificação de relação semântica entre pares de entidades. Essa tarefa é constituída das etapas de identificação das entidades nomeadas, extração do contexto e a classificação do tipo de relação. Além disso, ela exige que cada frase seja previamente anotada manualmente com duas menções de entidades para posteriormente ser submetida ao modelo de linguagem de IA para realizar a aprendizagem. Por exemplo, na sentença: "A GLO20XY ocorrerá no Rio de Janeiro". Assumem-se os pares de entidades "GLO20XY", como operação militar, e "Rio de Janeiro", como local, e a relação expressa entre elas é descrita por "ocorre" que é rotulada por "local_operacao".

Contudo, a *Sentence-Level RE* é restrita a uma única sentença e sensível a ruído em função de nem todos os pares de entidades expressarem de fato a mesma relação. Uma evolução natural a *Sentence-Level RE* é a tarefa *Bag-Level RE* que visa agregar as sentenças que mencionam o mesmo par de entidades em um *bag* de pares de entidades, reduzindo a necessidade de anotações de textos. Assim, a *Bag-Level RE* trata os problemas de ruído da tarefa anterior e minimiza a necessidade de anotação (85).

A abordagem *Document-Level RE* difere de *Sentence-Level RE* e *Bag-Level RE* em função dela lidar com múltiplas entidades em um documento e tratar dessas relações, como no caso de DocRED (86). Por outro lado, a abordagem *Zero-Shot RE* (87) utiliza *templates* de questionários de treinamento a partir de pares de relação nomeadas, $R(x, y)$, onde R define a relação, no caso *educado_em*(x,y), x representa o termo questionado e y a sua resposta. Enquanto que a abordagem *Few-Shot RE* utiliza poucas instâncias de treinamento e consegue lidar com a escassez de dados anotados, principalmente por seu aprendizado ser inspirado em aprendizado meta (*meta learning*) (54).

Por fim, a abordagem *Definition Extraction*, ou extração de definições, que originou

o *DeftEval*, é uma tarefa compartilhada com o SemEval Task 6 (89) que difere das demais abordagens por utilizar pares de termo-definição no contexto de frases (75). Nesse sentido, ela utiliza o *Corpus* DEFT[3] que foi aplicado no cenário de aprendizagem através da extração de conteúdo de Livros didáticos e Termos de contrato (88).

2.2 Modelo de linguagem

No Capítulo 1, abordamos o conceito de Modelos de Linguagem (ML), com ênfase ao BERT, destacando a sua aplicação em PLN. Introduzimos a questão dos ajustes de pesos para aprendizado do contexto através do conceito de ajuste fino ou *fine-tuning*, permitindo a incorporação semântica a um ML generalista. Nesta seção, vamos apresentar questões associadas aos ML em detalhes, distinguindo suas definições, características, representação vetorial, arquitetura *Transformers* e, por fim, apresentar, resumidamente, o Bidirectional Encoder Representations from Transformers (BERT) (12).

Mecanismos, modelos e algoritmos focados em realizar atividades de treinamento em IA evolvem, por exemplo, modelos de Redes Neurais (REN). Esses modelos são baseados em Deep Learning (DL) e dispõem de um método de aprendizagem que utiliza uma série de camadas sucessivas, a qual permite descobrir recursos ocultos automaticamente. Essas camadas representam neurônios matemáticos em rede que possuem uma camada de entrada (*input*), uma de saída (*output*) e outras camadas ocultas (*hidden*) (90).

Há diversas arquiteturas e modelos que combinam diferentes técnicas que são aplicadas em REN (67). Em PLN, uma das formas de se obter as relações entre NER é utilizar REN. Desde a publicação do trabalho de LeCun, Bengio e al(90), os métodos baseados em aprendizagem profunda vem se tornando cada vez mais populares. A princípio, as Redes Neurais Recorrentes/recursivas (RNN) e as Redes Neurais Convolucionais (CNN) se destacaram como os dois principais métodos baseados em REN.

As RNN servem para analisar dados de séries temporais e realizar previsões futuras. Elas são úteis para tarefas de PLN, principalmente para tradução automática de voz para texto. Há algumas variações de modelos de RNN que se destacam devido as capacidades de memorizar os dados de entrada na rede neural, dentre eles: Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (BiLSTM) e Conditional Random Field (CRF), também chamada de BiLSTM-CRF. Já as CNN foram inspiradas no estudo do córtex visual do cérebro e são utilizadas para tarefas de reconhecimento de imagens (65). A princípio, ambas as REN foram desenvolvidas para representar imagens, contudo estudos confirmam que elas são propícias também para representar estruturas de texto, i.e., sequências de palavras e árvores de dependências (91).

[3] https://github.com/adobe-research/deft_corpus

Como os modelos de DL utilizam vetores numéricos é necessário converter o texto de entrada em números a fim de utilizar o modelo. Esse processo de conversão é também conhecido como vetorização de texto. Inicialmente, foi apresentada a estratégia de vetorização *one-hot* em que o texto de entrada é representado por um vetor binário (0 e 1), caracterizando uma representação esparsa (65). Um exemplo de modelo Esparso é o **Bag-of-Words** (BoW), proposto por Salton, Wong e Yang(92), uma abordagem baseada em cálculos de densidade para escolha de um vocabulário de indexação de uma coleção de documentos. Cada documento é representado como um vetor e seus elementos são associados a um termo do vocabulário. Ao longo dos anos, houve proposições de métodos e técnicas que utilizam modelos de redes neurais não lineares. Esses modelos não lineares se baseiam em vetores densos como entrada, em substituição aos modelos lineares que consistiam em entradas esparsas, permitindo que os algoritmos de DL capturem não somente a similaridade entre palavras, mas também as representações distribuídas de palavras (também conhecidas como *word embeddings*) (93).

A *word embedding* consiste em uma estratégia de vetorização de texto de alta densidade e baixa dimensão, onde um vetor de números representa cada palavra no vocabulário, fornecendo uma forma de representação densa em que palavras similares são codificadas de modo semelhantes, como exemplo o **word2vec** proposto por Mikolov et al.(94) (65). O **word2vec** visa capturar representações sintáticas e semânticas com base em duas arquiteturas contínuas de modelos para calcular representações vetoriais, o Continuous Bag-of-Words Model (CBOW) e o Skip-gram Model (Skip-gram) (94). As palavras podem estar associadas ou não ao contexto em que estão representadas. Essas representações contextuais podem ainda ser unidirecionais ou bidirecionais. Em PLN, para tarefas de classificação de frases é necessário incorporar contextos bidirecionais (esquerda-direita e vice-versa) tanto para as representações unidirecionais independentes do contexto quanto as sensíveis ao contexto (83, 95). Por isso, há necessidade de arquiteturas mais robustas, como no caso do BERT (12).

Uma evolução na estratégia de vetorização é o BERT que utiliza o treinamento bidirecional do *Transformer* (12). O *Transformer* é um modelo baseado na arquitetura *encoder-decoder* que utiliza o mecanismo de autoatenção ou *self-attention*. Esse mecanismo permite aprender relações contextuais entre palavras dentro do texto, relacionando diferentes posições, ao invés de analisar a sequência de modo unidirecional, como visto nas outras estratégias (83). Além disso, o BERT consiste em um modelo para pré-treinamento de representação bidirecional (esquerda e direita) profunda de texto não rotulado em todas as camadas. Ele permite ajustes com apenas uma camada de saída adicional e tem como objetivo a geração de um modelo de linguagem. Basicamente, ele é composto por duas arquiteturas que se diferenciam pelo número de camadas. O BERT Base que contém 12 camadas *Transformer* de 12 núcleos. E BERT Large com 24 camadas *Transformer* de 16 núcleos. Assim, ele utiliza apenas codificadores empilhados, correspondendo ao estado da

arte de modelo de PLN (12).

O treinamento do BERT consiste em duas etapas: pré-treinamento (*feature-based*) e *fine-tuning*, ilustrado na Figura 4. Na epata de pré-treinamento, o modelo é treinado em dados não rotulados para otimizar conjuntamente a realização de duas tarefas: a) *Masked Language Model* (MLM) e; b) *Next Sentence Prediction* (NSP). Nessa etapa, ao BERT é incorporado diversos conceitos do aprendizado sem especificar um contexto alvo, i.e., deixando o ML com característica generalista.

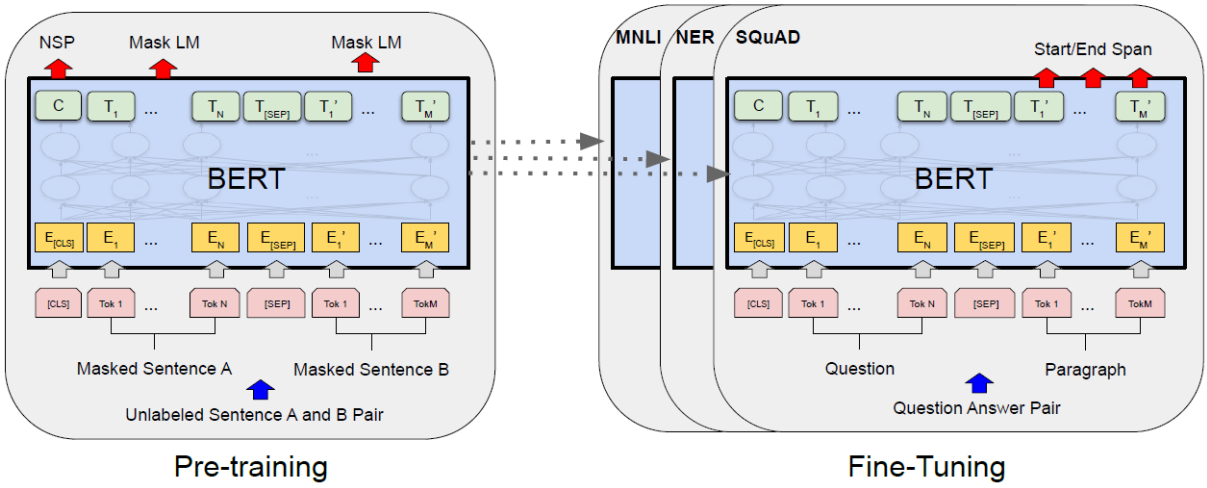


Figura 4 – Ilustração das etapas de pré-treinamento e *fine-tuning*. Adaptado de Devlin et al.(12)

A tarefa MLM tem como objetivo mascarar 15% das palavras, utilizando alguns *tokens* (MASK), e prever o ***id*** do vocabulário original, levando em conta o contexto das palavras não ocultadas. Por sua vez, o NSP, dado como pares de sentenças (A e B), tem como objetivo o pré-treinamento do conjunto das representações de pares de texto, identificando se as sentenças são subsequentes ou não. Enquanto que a etapa *fine-tuning* ocorre após o modelo ter sido pré-treinado. Ela consiste em uma técnica de *transfer learning* que utiliza um modelo pré-treinado com base nos dados da Wikipedia e realiza um treinamento adicional com o conjunto de dados. Cabe ressaltar que há variações do BERT treinados com outros conjuntos de dados, como exemplos: DistilBERT (32) e RoBERTa (33). Além disso, há extensões pré-treinadas em outros idiomas, como exemplo o BERTimbau(36) em língua portuguesa. Há também outros modelos ajustados a contextos específicos através de *fine-tuning*, como o BioBERT(21), SciERC(9), como abordado no Capítulo 1.

No Quadro 1, são ilustradas as representações vetoriais a partir do exemplo, "A GLO20XY ocorrerá no Rio de Janeiro", apresentado na subseção 2.1.4. Inicialmente, o texto é dividido em sete *tokens* de palavras, ["A", "GLO20XY", "ocorrerá", "no", "Rio", "de", "Janeiro"], numeradas de 1 a 7 na coluna *T*. Na coluna *One-hot*, é representado cada

Quadro 1 – Tipos de representações vetoriais.

T	**One-hot**	**Word Embedding**	**Vetor com BERTimbau**
1	[1, 0, 0, 0, 0, 0, 0]	[0.12, 0.04, 0.09, 0.03]	[-4.7082 -4.4419 -5.8756 ... -2.6283]
2	[0, 1, 0, 0, 0, 0, 0]	[0.77, 0.15, -0.30, 0.42]	[0. 0. 0. 0. 0.]
3	[0, 0, 1, 0, 0, 0, 0]	[0.50, 0.62, -0.10, 0.48]	[-0.35232 1.082 -3.0444 ... -2.5344]
4	[0, 0, 0, 1, 0, 0, 0]	[0.18, 0.25, 0.08, 0.07]	[-3.9856 -0.20308 0.55546 ... -1.6248]
5	[0, 0, 0, 0, 1, 0, 0]	[0.70, 0.55, 0.22, 0.60]	[-0.14677 0.059509 -2.4065 ... 3.1655]
6	[0, 0, 0, 0, 0, 1, 0]	[0.10, 0.20, 0.04, 0.06]	[3.9609 4.5764 -2.5355 ... -7.2996]
7	[0, 0, 0, 0, 0, 0, 1]	[0.72, 0.57, 0.20, 0.62]	[0.78665 1.5896 -1.8703 ... 0.57558]

token através de um vetor binário em que o número "1" indica a posição da palavra no vocabulário. Como abordado, o *one-hot* produz vetores esparsos e sem relação semântica. Por outro lado, na coluna *Word Embedding*, são apresentados vetores densos de 4 dimensões, representando relações semânticas e sintáticas entre as palavras. Ao analisar os *tokens*, 5 e 7, respectivamente, ["Rio"] e ["Janeiro"], nota-se que seus valores são muito próximos, presumindo que o modelo de linguagem inferiu que eles são semanticamente relacionados. De modo semelhante, ocorreu com os *tokens*, 4 e 6, respectivamente, ["no"] e ["de"], em função de serem preposições. Diferentemente do que ocorreu com o *token* 2, ["GLO20XY"], provavelmente por ser uma entidade distante em relação aos outros *tokens*. Por fim, na coluna Vetor com BERTimbau, contém representações vetoriais com 300 dimensões densas, geradas por meio do código-fonte[4], utilizando o modelo BERTimbau (pt_core_news_md[5]). Ao analisar os resultados, nota-se que os *tokens*, 5 e 7, ["Rio"] e ["Janeiro"], possuem uma alta similaridade de 94% em função de sua proximidade semântica, análogo ao Word embedding. Em contrapartida, o *token* 2, ["GLO20XY"], ficou com o valor zero em função de não ter sido reconhecido pelo modelo.

Independente do tipo de representação, os vetores de *embeddings* constituem os parâmetros aprendidos do ML, já que os pesos neles atribuídos são aprendidos através dos dados de treinamento. Os valores dos pesos são ajustados para capturar as características semânticas e sintáticas dos *tokens*, como explorado no exemplo entre os *tokens*, 5 e 6, que representam as palavras "Rio" e "Janeiro". Esses parâmetros influenciam a qualidade final dos ML em conjunto com os hiperparâmetros. Os hiperparâmetros definem um conjunto prévio de valores externos para parametrizar o processo de treinamento ou ajuste fino de um ML (61). Há diversos hiperparâmetros disponíveis, porém a seguir são listados alguns comumente utilizados:

- *Batch size*: define o número de amostras de treinamento utilizadas antes da atualização dos pesos do ML.

[4] <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/deliverables/idea-c2/exemplos/exemplo1.ipynb>

[5] https://github.com/explosion/spacy-models/releases/tag/pt_core_news_md-3.8.0

- *Epochs*: é o número total de vezes que todos os exemplos utilizados passam no treinamento do ML. Além disso, em conjunto com o *dropout*, ele está intimamente ligado aos problemas de sobreajuste (*overfitting*) e subajuste (*underfitting*). Por exemplo, quando o número de *epochs* é muito alto, o ML reconhece em demasia os dados de treinamento e perde a capacidade de generalização, caracterizando o *overfitting*. Porém, quando o número de *epochs* é muito baixo faz com que o ML não reconheça os dados de treinamento, caracterizando o *underfitting*.

- *Dropout*: é uma técnica de regularização utilizada para controlar os problemas de *overfitting*, evitando que o ML perca a capacidade de generalização através de uma taxa fixa que desativa os neurônios da rede neural. Por exemplo, ao definir um *dropout* de 0,1, cerca de 10% dos neurônios da rede são desativados.

Como os hiperparâmetros atuam na etapa que antecede o treinamento do ML, após o treinamento a qualidade é avaliada através das métricas de desempenho. Há diversas métricas no contexto de PLN para avaliar a performance do modelo, como veremos em detalhes no Capítulo 4. As variáveis a seguir descrevem os valores de retorno de cada métrica e são utilizadas como parâmetros nas fórmulas: *1) TP (True Positive); 2) FP (False Positive); e 3) FN (False Negative)*. As fórmulas correspondem aos cálculos realizados para se obter os valores de cada métrica. Nesse sentido, são utilizadas as métricas: *a) Precision* que se refere à porcentagem de acertos, dada pela equação: $P = \left(\frac{TP}{TP+FP}\right)$; *b) Recall* que se refere à porcentagem de entidades reconhecidas, dada pela equação: $R = \left(\frac{TP}{TP+FN}\right)$; *c) F1-Score* que se refere à média harmônica entre *Precision* (P) e *Recall* (R), dada pela equação: $F1 - Score = 2 * \left(\frac{P*R}{P+R}\right)$ (61). Cabe destacar que essas métricas não se limitam a avaliar a performance do modelo nas tarefas de treinamento ou *fine-tuning*, elas também podem ser utilizadas na avaliação de predição de modelos, como explorado no exemplo a seguir.

Um exemplo prático[6] para demonstrar a avaliação dos resultados das métricas (*Precision*, *Recall* e *F1-Score*), é aplicado a partir do exemplo, "A GLO20XY ocorrerá no Rio de Janeiro", apresentado na subseção 2.1.4. O objetivo é identificar a capacidade de inferência do modelo BERTimbau Large pré-treinado (pt_core_news_lg[7]) na tarefa NER. É importante relembrar que as entidades nomeadas "GLO20XY", se referem à operação militar (OP_MIL) e "Rio de Janeiro" ao local (LOC). Na preparação, foi adicionada artificialmente a categoria "no_ent" tendo em vista que pt_core_news_lg não foi treinada com OP_MIL. Esse artifício vai facilitar a análise da matriz de confusão. Ao submeter o texto do exemplo, somente a entidade nomeada "Rio de Janeiro" foi reconhecida como LOC. Logo, os valores das três métricas obtiveram resultados de 100%.

[6] <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/deliverables/idea-c2/exemplos/exemplo2.ipynb>

[7] https://github.com/explosion/spacy-models/releases/tag/pt_core_news_lg-3.8.0

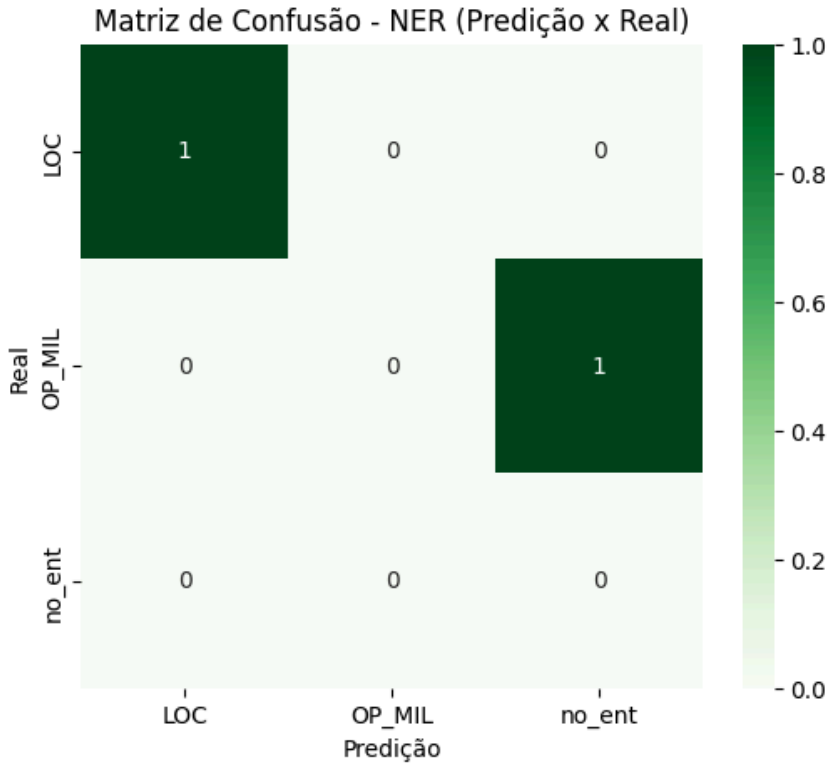


Figura 5 – Avaliação da predição de **BERTimbau** na tarefa NER. Imagem do Autor.

Enquanto que a entidade "GLO20XY", como já esperado, não foi reconhecida pelo modelo, assim, obtendo o valor zero em todas as métricas. A análise desses valores é mais explícita na matriz de confusão (Figura 5). Espera-se o valor "1" na interseção entre as mesmas categorias. Isso somente ocorreu entre LOC (real) versus LOC (predição), pois o modelo somente reconheceu "Rio de Janeiro" como LOC. Diferentemente do que ocorreu na interseção entre OP_MIL (Real) e no_ent (Predição), representando que a predição não corresponde ao valor real.

Nesta seção, foi explorado em detalhes os aspectos que envolvem modelos de linguagem, em destaque ao BERT, por meio de exemplos para explicitar os conceitos apresentados. Contudo, os ML são caixas-pretas e seus dados, bem como a estrutura de organização, são limitadas aos conjuntos de dados de treino, como podemos observar no último exemplo quando BERTimbau não reconheceu "GLO20XY" tendo em vista que não foi treinado com a categoria "OP_MIL", necessitando neste caso de *fine-tuning*. Nesse sentido, é possível explicitar esses dados através de uma estrutura de representação mais rica e robusta, como veremos na seção 2.3.

2.3 Modelagem Conceitual de Dados

No contexto de dados textuais não-estruturados, ao identificar candidatos a entidades nomeadas e relações, é possível obter um conjunto de termos que expressam informações (e.g. conceitos e relações) de um domínio específico (61). Contudo, apesar dos documentos serem ricos, a falta de estrutura de representação ou a ausência de um esquema formal, dificulta a extração de esquemas conceituais ou modelagens conceituais que representem o domínio. Isso é o que ocorre com os ML dada a sua natureza *Data-Driven (DD)*, como mencionado. Os ML inferem as categorias, por exemplo, a partir de uma função extensional que é aprendida através dos dados (instâncias), i.e., de forma *bottom-up* (15). Em outras palavras, supondo que o ML foi treinado com uma lista qualquer de livros do autor hipotético "XYZ ABC", ele vai utilizar essa estatística contextual para inferir que "XYZ ABC da Silva" é também autor dos mesmos livros dada a alta probabilidade resultante.

Por outro lado, os modelos de dados conceituais, como o modelo de dados ER e outros, são ferramentas úteis que viabilizam estruturar o conhecimento que se pode encontrar nos textos (63). Eles têm como finalidade representar de forma não ambígua os objetos, características, relacionamentos do mundo real, permitindo a elaboração de modelos de domínio (Modelo de Domínio (DM)) (96, 97). Assim, os DM são construídos seguindo a abordagem *Theory-Driven (TD)*, i.e., a partir da observação do domínio é definida conceitualmente a sintaxe e a semântica de armazenamento. Nessa abordagem, os dados ou instâncias são utilizados para validar o modelo, permitindo que os DM sejam explicáveis pois a sua construção é baseada em uma teoria do mundo real (15).

O modelo ER oferece os construtos: entidade, atributo e relacionamento. As entidades representam os objetos, como exemplo no domínio de C2 um *comandante* ou uma *força singular*. Por sua vez, os atributos descrevem as propriedades das entidades, como o nome de um comandante ou de uma força singular. O relacionamento representa a relação entre pares de entidades e distingue o tipo de relação, como a que descreve a responsabilidade de um comandante sobre uma força singular (63, 98). Esses construtos podem ser expressos por níveis de abstração, demonstrando a visão do negócio em que cada objeto pode ser representado em um domínio. Além disso, os DMs, geralmente, permitem representações visuais, em que conceitos e relações entre eles podem ser ilustrados por meio de diagrama de classes (99) ou Modelo de Entidade-Relacionamento (MER) (98), como ilustrado na Figura 6.

A abstração de dados refere-se ao mecanismo que permite encapsular os detalhes de armazenamento, assim como a organização dos recursos essenciais para entender os dados (63). Neste trabalho, estende-se o conceito de abstração de dados como algo que determina se um objeto está representado como **esquema** ou **instância** em um domínio (100). O esquema é representado pelas entidades, atributos e relacionamentos por descreverem

o objeto e suas características, no caso os metadados, como por exemplo a entidade "comandante" e seus atributos "nome", "unidade", etc. As instâncias ou os dados são ocorrências associadas ao esquema, como exemplo, "**João da Silva**", que representa o nome do comandante (63). Além disso, a abstração em níveis é significativa à medida que pode influenciar operações complexas de interligação em conjuntos de dados, principalmente aquelas que podem envolver metadado e dado, como veremos na seção 2.5 (101, 102).

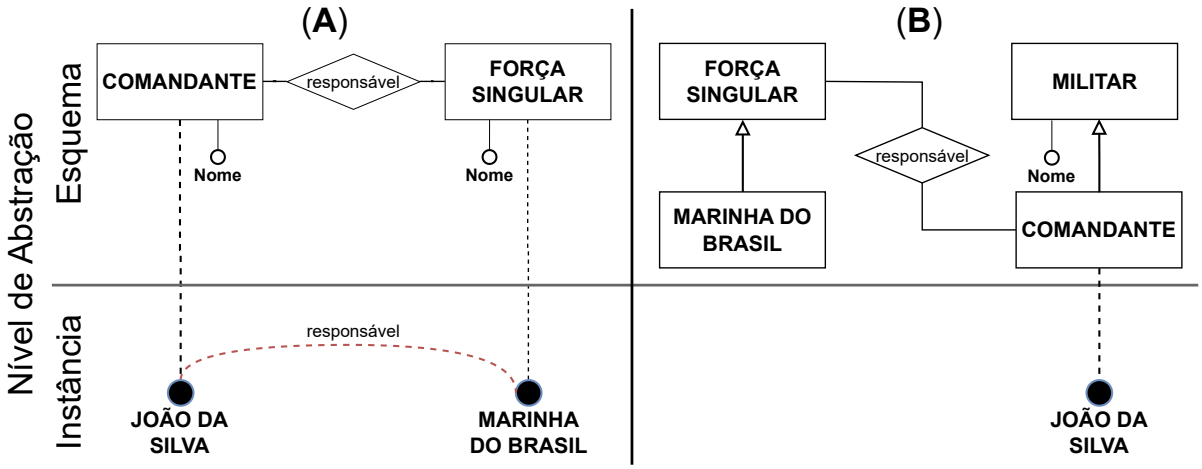


Figura 6 – Modelos conceituais com visões distintas da entidade Marinha do Brasil. Adaptado de Kent(103).

A Figura 6 ilustra dois quadros, **A** e **B**, que representam visões distintas de esquemas de dados conceituais, representados por níveis de abstração, evolvendo as entidades **comandante** e **força singular** e seu relacionamento. No quadro A, ambas as entidades descrevem os objetos comandante e força singular, assim como as ocorrências de cada uma delas através das instâncias, respectivamente, "JOÃO DA SILVA" e "MARINHA DO BRASIL". Por outro lado, no quadro B, a única diferença é que "MARINHA DO BRASIL" é representada como uma entidade especializada de "FORÇA SINGULAR". Assim, é possível que modelos de dados conceituais apresentem percepções distintas do mundo real, em função dos envolvidos no levantamento de um domínio possuírem visões diferentes sobre o mesmo fato, entidade, relação ou instância (104).

Há tempos que existem discussões sobre as formas distintas de se modelar um mesmo fato dentro de um domínio. Isso pode influenciar a forma e o modo como os fatos e as relações podem ser expressos (103). Ao lidar com dados textuais não-estruturados, isso pode ser agravado, como no caso dos ML. Além disso, especificar um modelo conceitual prevendo todas as entidades e relações parece é algo desafiador. Entretanto, uma alternativa para contornar essa questão, talvez seja estabelecer construtos de alto nível mais expressivos. Por exemplo, através uma metamodelagem capaz de comportar modelos distintos, como detalhado na próxima seção.

2.4 Metamodelagem

A metamodelagem tem como objetivo estruturar objetos complexos e genéricos do mundo real a partir de construções de abstração de alto nível, permitindo alterações estruturais aos seus objetos e preservando a sua representação original (105). Na visão de Brambilla, Cabot e Wimmer(106), um metamodelo representa uma abstração do modelo que destaca as suas propriedades, constituindo a definição de uma linguagem de modelagem em função de ela descrever as classes dos modelos. Há autores que definem metamodelo como uma técnica de modelagem utilizada para descrever a semântica do próprio modelo (97). Por outro lado, há autores que descrevem o uso de metamodelos como uma solução para problemas conhecidos, análogo aos padrões de projeto de Gamma et al.(107), dada a capacidade de abstração e aplicação em diferentes contextos e domínios (105).

Há exemplos de aplicação de metamodelos de diversos modos e contextos. Um exemplo aplicado na Engenharia de Software é o *Meta Object Facility* (MOF)[8] da *Object Management Group* (OMG) (106). O MOF é um metamodelo criado para definir outros modelos, inclusive linguagens, como exemplo a *Unified Modeling Language* (UML) (108). Outro exemplo de uso de metamodelo aplicado na modelagem conceitual em Banco de Dados, utilizando a notação de Chen(98), é o *Enhanced Entity-Relationship MetaModel* (EERMM) (109). O EERMM é um metamodelo que utiliza três construtos (*node*, *link* e *schema*) para modelagem conceitual *Enhanced Entity-Relationship* (EER) (98, 63). Nele, o construto *node* são as entidades, associações, atributos, relacionamentos e heranças. Enquanto o *link* representa as relações dos construtos. (110, 109, 111).

Como os metamodelos representam abstrações de alto nível dos próprios modelos é possível definir níveis infinitos de metamodelagem. Entretanto, a OMG(108) adotou a arquitetura de quatro níveis, que se tornou referência de abstração na construção de metamodelos, como ilustrado na Figura 7. Nela, destacada à esquerda, mostra-se que um modelo se adapta ao seu metamodelo (*conformsTo*) quando todos os seus elementos podem ser expressos como instâncias (*instanceOf*) das metaclasses correspondentes do metamodelo. Além disso, à direita, é ilustrado um exemplo resumido de um modelo do filme Casablanca[9], explicitando os quatro níveis de abstração, de **M0** a **M3**. O nível **M0** é a instância do mundo real, i.e., próprio filme. O **M1** representa o modelo que descreve o objeto do filme, no caso a classe Vídeo e seu atributo. Enquanto **M2** é o metamodelo que representa os construtos **Classe** e **Atributo** instanciados em **M1**. Por fim, **M3** é o nível meta-metamodelo com o construto **Classe** que representa o maior nível de abstração da arquitetura de metamodelagem (106).

Como vimos, há definições variadas acerca de metamodelos (97, 105, 106), assim como implementações com notações e contextos distintos (106, 109). Neste trabalho, vamos

[8] https://www.omg.org/mof/
[9] https://pt.wikipedia.org/wiki/Casablanca_(filme)

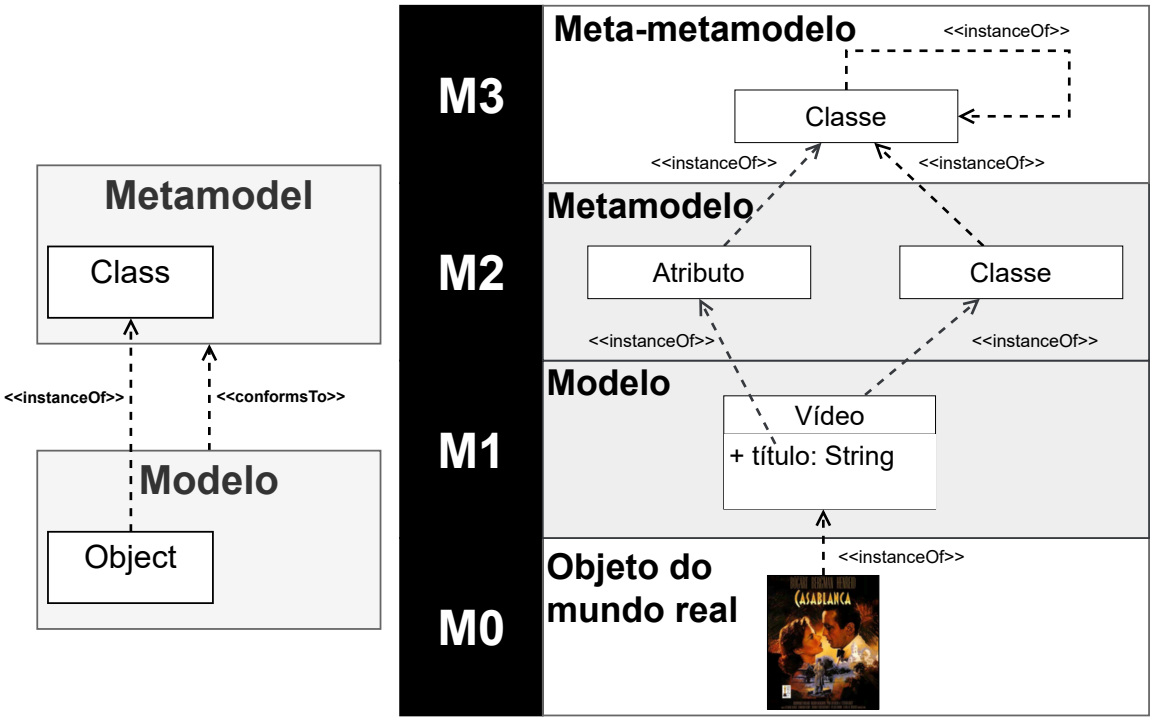


Figura 7 – Exemplo de representação de metamodelo em níveis do filme Casablanca. Adaptado de Brambilla, Cabot e Wimmer(106).

assumir que os metamodelos são abstrações que utilizam construtos de alto nível para descrever modelos conceituais por níveis de abstração, permitindo que modelagens distintas, nos casos possíveis, consigam conviver com o mesmo metamodelo.

Na seção anterior, foi apresentado um exemplo, na Figura 6, com duas formas distintas de modelagens de entidades e atributos do domínio de C2, envolvendo *comandante*, *força singular*, além do relacionamento de responsabilidade e suas instâncias. Na Figura 8, adotamos um metamodelo que atende as duas modelagens conceituais, permitindo que ambas as soluções convivam em uma única modelagem, preservando todas entidades, atributos e relacionamentos dos modelos conceituais originais. Além disso, são ilustrados, por níveis de abstração, os objetos do mundo real, em M0 Instância, os modelos de dados conceituais, nos *frames A* e *B*, em M1 Modelo, o metamodelo com os construtos *entidade*, *relação* e *atributo*, em M2, e a *Entity* que representa o nível meta metamodelo no EER, em M3 (106).

Como apresentado nesta seção, a metamodelagem é capaz de gerar ou até mesmo fazer com que modelos conceituais distintos consigam conviver em um modelo. Entretanto, a sua utilização requer uma abordagem de persistência com características flexíveis, que permita coexistir representações de dados em diferentes níveis de abstração de modo transparente a seus usuários. Um KG baseado no grafo RDF possui essas características e

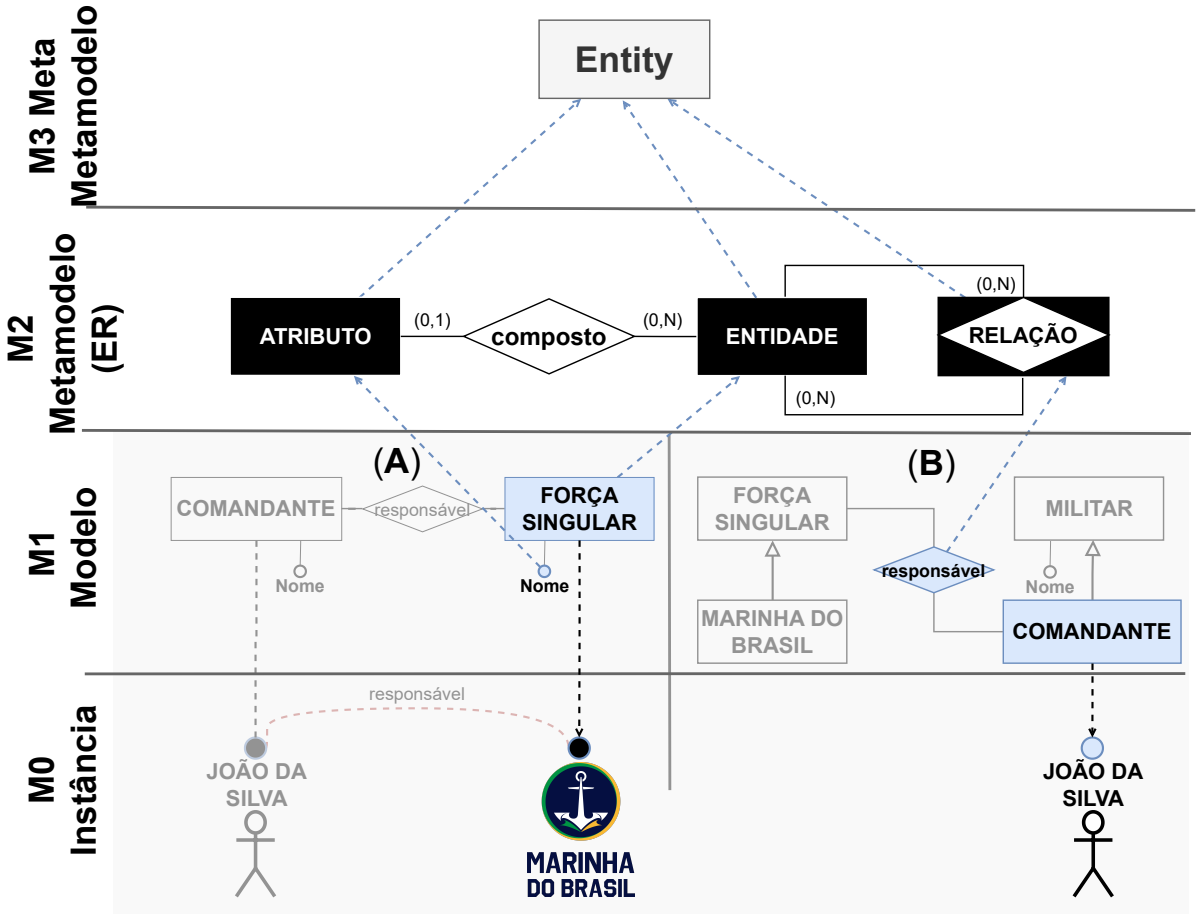


Figura 8 – Metamodelo simplificado aplicado nos modelos conceituais da Figura 6. Imagem do autor.

possibilita a realização de inferências, como veremos na próxima seção.

2.5 Grafos de Conhecimento

Basicamente, um Grafo de Conhecimento (GC) (Knowledge Graph (KG)) é uma representação expressa em grafo com o objetivo de acumular e transmitir conhecimento. Ele é baseado no modelo de dados em grafo, caracterizado por sua flexibilidade de representação, permitindo que os dados armazenados não possuam, obrigatoriamente, um esquema previamente definido (19). O KG é composto por três construtos básicos: *nós* (que representam os vértices do grafo); *relacionamentos* (que representam as arestas); e as *propriedades* dos nós e relacionamentos. A principal característica da abordagem em grafo é permitir que uma determinada aplicação execute consultas que atravessem uma rede de nós e arestas e analise os relacionamentos entre as entidades (112).

O conhecimento expresso no KG pode ser explorado por meio de inferências de modo indutivo ou dedutivo. A indução generaliza padrões a partir de fatos conhecidos e gera novas conexões no grafo. Com base nisso, novas inferências podem ser realizadas

e consequentemente novas conclusões são obtidas, assemelhado aos algoritmos de AM. Como exemplos de inferência de modo indutivo têm-se, a análise de centralidade, detecção de comunidade, dentre outras operações (19). Por outro lado, a dedução é baseada em um conjunto de premissas que implica em formalizar uma consequência lógica (19). Na dedução, é utilizada a Lógica de Primeira Ordem, como por exemplo a dedução de novas relações a partir de propriedades das relações (e.g. transitividade, simetria, etc.) ou de axiomas lógicos pré-definidos. Há casos em que são usadas ontologias, que se baseiam em declarações descritivas precisas de representação do conhecimento (113).

Os grafos podem ser direcionados ou não, com arestas e nós rotulados ou não. Eles possuem propriedades em nós e arestas, incluindo ainda os hipergrafos e hipernodos. Dentre os tipos de grafo tem-se: *(i) Directed Edge-labelled Graph*, *(ii) Heterogeneous Graph*, *(iii) Property Graph* e *(iv) Graph Dataset* (114). Apesar de haver essa variedade, neste trabalho vamos focar no *Directed Edge-labelled Graph*. Para exemplificar, é ilustrado na Figura 9, um KG em que os nós representam algumas cidades Chilenas e as relações entre elas são estabelecidas pelo tipo de cobertura modal existente (aéreo ou ônibus). Os nós destacados em azul representam as cidades com cobertura no modal aéreo. Enquanto que os nós em verde, destacam as cidades que possuem cobertura somente pelo modal ônibus. Esse exemplo permite a realização de análises sobre esses dados para apoiar decisões no domínio de logística de transportes, como por exemplo escolher trajetórias entre duas cidades.

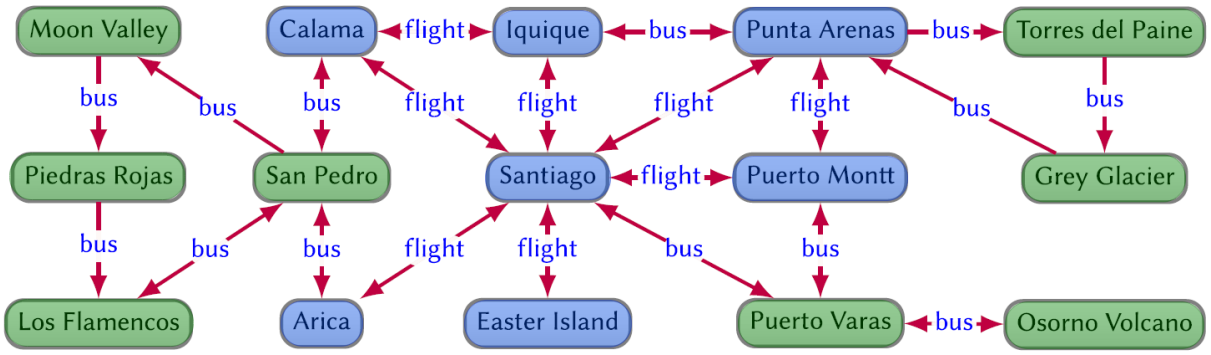


Figura 9 – Exemplo de KG de cidades do Chile cobertas por transportes nos modais aéreo e ônibus. Adaptado de Hogan et al.(19).

O *Directed Edge-labelled Graph* é um tipo de grafo rotulado e suas arestas são direcionadas, o que provê conhecimento de modo dedutivo, como por exemplo o grafo Resource Description Framework (RDF)[10] (19). O RDF é um *framework* recomendado pela W3C que representa dados e metadados interligados na Web (115). Os dados modelados em RDF formam um grafo de dados representado por triplas, compostas por *sujeito, predicado e objeto*. Cada elemento da tripla permite associar e reutilizar vocabulários controlados

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

ou ontologias, os quais são representados na Web de Dados por meio do RDF, como ilustrado na Figura 10. Além disso, os elementos da tripla são identificados por *Uniform Resource Identifier* (URI). Cada URI é um conjunto de caracteres, constituído por valores válidos que identifica recursos e propriedades. A utilização do URI evita ambiguidades e facilita o entendimento das máquinas (115). Além disso, vale destacar que o RDF permite representar dado e metadado de modo uniforme, acomodando em um mesmo grafo nós de vários níveis de abstração.

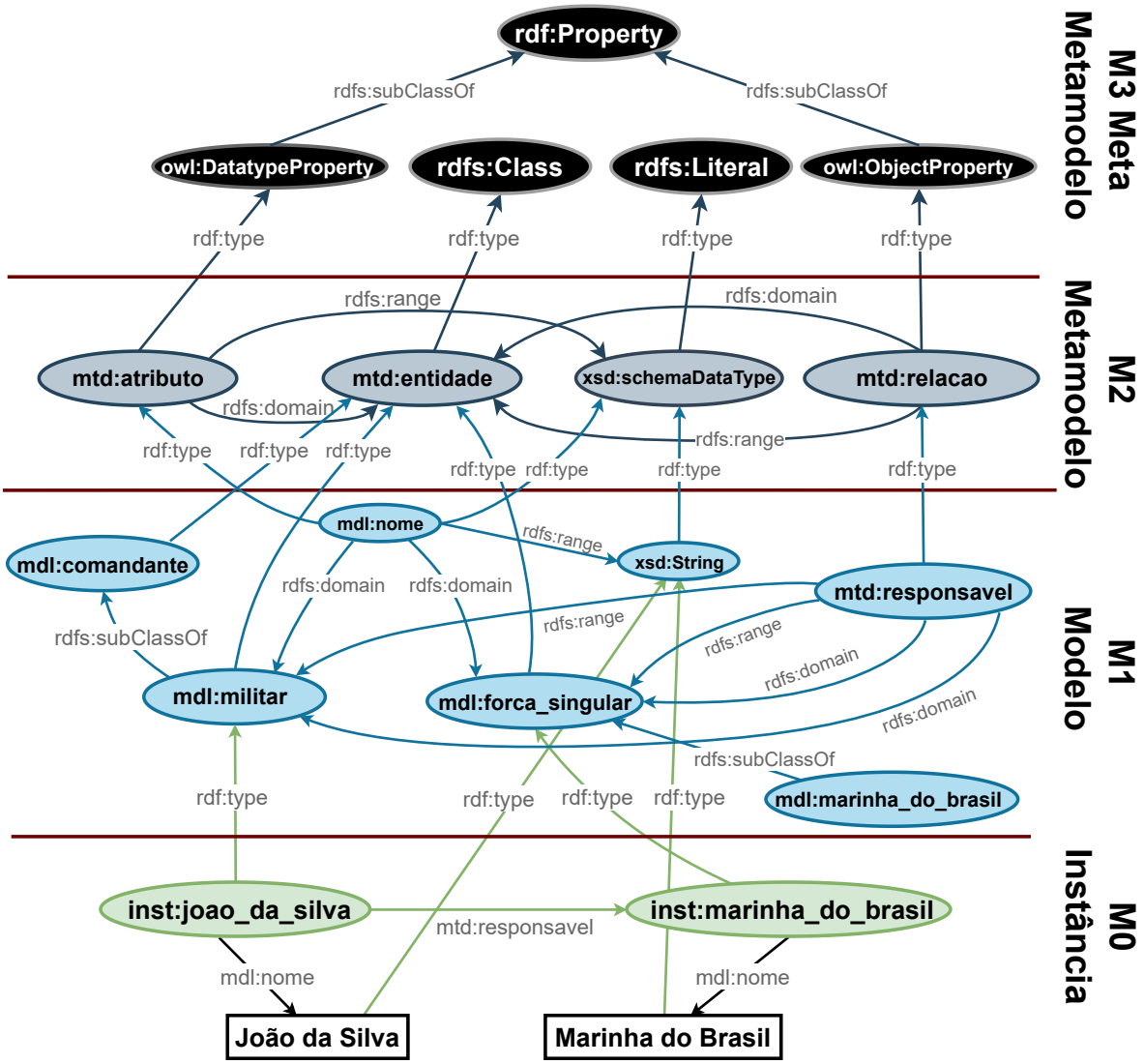


Figura 10 – Exemplo de grafo RDF aplicado no contexto de C2 em níveis de abstração. Imagem do autor.

Na Figura 10, é ilustrado o exemplo de um grafo RDF com base no mapeamento entre os diferentes níveis de abstração da modelagem ilustrada na Figura 8. No grafo, as triplas formadas por sujeito, predicado e objeto são compostas por recursos vinculados. Cada recurso é destacado por níveis de abstração, de **M0** a **M3**. Note que as instanciações

entre os recursos dispostos nos níveis se dá através da propriedade *rdf:type*.

No **M3**, na cor preta, são representadas as metaclasses do RDF, ***rdfs:Class*** e ***rdf:Property***. Enquanto que no **M2**, na cor cinza, os recursos ***mtd:entidade***, ***mtd:atributo***, ***mtd:relacao*** compõem o metamodelo, o qual descreve os recursos instanciados em **M1** do modelo. No **M1**, na cor azul, são representados os recursos, ***mdl:militar***, ***mdl:comandante***, ***mdl:forca_singular*** e suas especializações, responsáveis por descreverem as instâncias em **M0**. Por fim, no **M0**, na cor verde, são representadas as instâncias, as quais são compostas por triplas de recursos e literais, em destaque ao comandante ***inst:joao_da_silva*** e à força singular ***inst:marinha_do_brasil***.

Note que o grafo RDF é flexível e congrega todos os recursos de modo transparente, mesmo em casos como **Marinha do Brasil**, que possui duas visões distintas oriundas do modelo conceitual, representadas como **M1** e **M0**. Essa flexibilidade dá ao grafo inúmeras vantagens sobre as demais representações de dados (112). Além disso, é ainda possível realizar consultas, agregar vocabulários e ontologias, inferir novos fatos e buscar interligações a partir de recursos semelhantes, visando o enriquecimento dos dados e, com isso, ampliar o conhecimento do domínio.

Os grafos em RDF podem ser publicados na Web e disponibilizados para o consumo e demais serviços. O Linked Open Data (LOD)[11] surgiu para estimular a interligação entre conjuntos de dados na Web de dados. Ele representa uma nuvem abstrata de dados interligados, onde os dados são distribuídos em repositórios. Os dados publicados no LOD podem estar associados às ontologias e, assim, ajudar nas inferências. Baseado nas interligações, a Web Semântica (WS) fornece um conhecimento amplo de diversos domínios (115). Entretanto, algumas interligações não foram efetivas na extração de conhecimento dada sua precariedade semântica. Dessa forma, surgiram abordagens focadas na ampliação de conjuntos de dados a fim de enriquecer a semântica das relações (116). Neste trabalho, não será abordado sobre enriquecimento de dados.

Nesta seção, foi apresentado como o KG favorece a representação do conhecimento e possui uma estrutura flexível para lidar com dados textuais. Além disso, o KG permite a interligação de conjuntos de dados multidomínios e a inferência de recursos vinculados. Entretanto, apesar do KG suportar dados semânticos e atribuir raciocínio utilizando ontologias, não é trivial extrair conceitos fundamentados.

2.6 Comando e Controle

Por definição, Comando e Controle (C2) é a "Ciência e arte que trata do funcionamento de uma cadeia de comando" (117). Como nas FA, a cadeia de comando é exercida por meio da sequência hierárquica de comandantes, as atividades de C2 tratam

[11] https://lod-cloud.net/

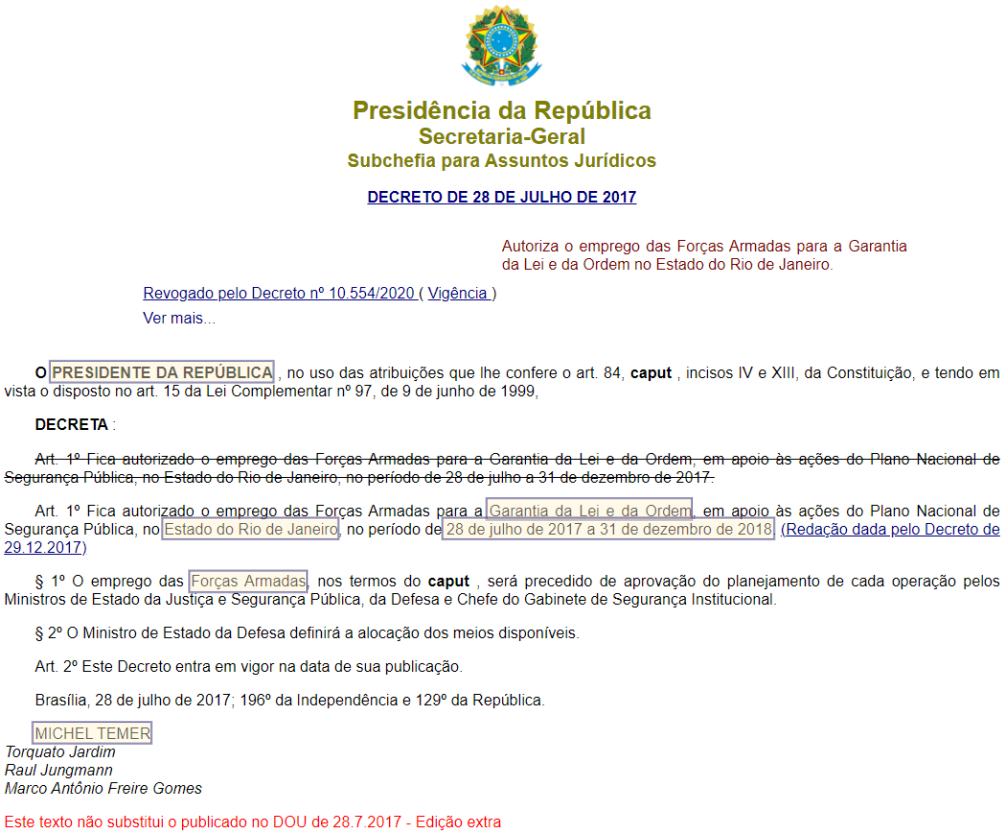
do funcionamento dessa cadeia de comando e são fundamentais para o êxito das operações militares. O C2 envolve três componentes: a autoridade, o processo decisório e a estrutura. A autoridade envolve as decisões para o exercício e controle do comando. Já o processo decisório possui base doutrinária para formular ordens e estabelecer o fluxo de informações e o seu cumprimento. Por fim, a estrutura inclui pessoas, equipamentos e tecnologias necessários ao exercício de C2 (118).

O MD e as FA são dotados de Doutrinas Militares (DML) que compreendem um conjunto harmônico e integrado de princípios, conceitos, normas e procedimentos, baseados na prática e experiência, que define, orienta as ações e atividades para o pleno emprego de seu pessoal nas operações e exercícios (118). As DML possuem abrangência com amplo espectro de atuação e são associadas às Capacidades Nacionais de Defesa (CND), as quais envolvem desde proteção, dissuasão, gestão da informação, logística até desenvolvimento tecnológico de defesa. Em busca livre nos repositórios públicos do MD e das FA foram listados um total de 912 documentos entre doutrinas e manuais, sendo 40 do MD (119), 80 da MB (120), 562 do EB (121) e 230 da FAB (122). Dessa forma, as DML estão presentes em toda estrutura da cadeia de comando desde o MD até as FA, compondo um rol de documentos normativos que seguem uma linguagem própria e são consubstanciados por leis, normas, glossários e abreviaturas. Há exemplos de Doutrina Militar (DML) para: Operações Conjuntas (123), Sistema Militar de Comando e Controle (124), Conceitos de Operações (125), Doutrina Militar Terrestre (DMT) (66), dentre outras.

As FA preparam continuamente seu efetivo para o planejamento e emprego em operações e exercícios singulares ou conjuntos (Op Cj) com objetivo de aprimorar seu pessoal, equipamentos e serviços para pronta-resposta da tropa de forma integrada em situação de guerra ou não-guerra. A coordenação desses exercícios e operações cabe ao Estado-Maior Conjunto das Forças Armadas (EMCFA), responsável pelo planejamento estratégico das FA (123, 117). Cada vez mais, os conflitos entre nações tendem a não ser declarados e com maior imprevisibilidade de duração. Nesse sentido, há uma necessidade constante das FA, com base em suas CND, de treinar seu pessoal para atuar de forma efetiva nessas operações. Uma das formas de treinamento pode ser implementada através de sistemas de informação que promovam maior interação através, por exemplo, de tarefas de PLN. No trabalho Mosafi et al.(126), é explorado um cenário em que um agente de IA apoia as comunicações através de um chat tático baseado em PLN, colaborando na identificação, priorização e distribuição de mensagens durante a operação.

Um exemplo de operação militar é a Garantia da Lei e da Ordem (GLO) que é instituída pelo Presidente da República. Essa operação é de caráter temporário ou episódico em que as FA atuam de modo coordenado com demais órgãos de segurança pública do Estado. Ela tem como propósito assegurar o funcionamento do estado democrático de direito, conforme expresso na Constituição Federal do Brasil (1). Na Figura 11, é ilustrado

o Decreto Presidencial que instituiu a operação de GLO, em 2017, no estado do Rio de Janeiro. É possível identificar no decreto que algumas informações destacadas contribuem para entender o contexto e as restrições de sua aplicação. As informações como o tipo de atuação, o local, o período e o emprego a ser realizado pelas FA definem o papel e a atuação pretendida pela autoridade máxima do país na operação (127). Essas informações, se bem tratadas, podem servir de insumos para ajustar ML na condução de extração de informações condizentes aos interesses das FA.



Presidência da República
Secretaria-Geral
Subchefia para Assuntos Jurídicos

DECRETO DE 28 DE JULHO DE 2017

Autoriza o emprego das Forças Armadas para a Garantia da Lei e da Ordem no Estado do Rio de Janeiro.

Revogado pelo Decreto nº 10.554/2020 (Vigência)
Ver mais...

O PRESIDENTE DA REPÚBLICA, no uso das atribuições que lhe confere o art. 84, **caput** , incisos IV e XIII, da Constituição, e tendo em vista o disposto no art. 15 da Lei Complementar nº 97, de 9 de junho de 1999,

DECRETA :

~~Art. 1º Fica autorizado o emprego das Forças Armadas para a Garantia da Lei e da Ordem, em apoio às ações do Plano Nacional de Segurança Pública, no Estado do Rio de Janeiro, no período de 28 de julho a 31 de dezembro de 2017.~~

Art. 1º Fica autorizado o emprego das Forças Armadas para a Garantia da Lei e da Ordem, em apoio às ações do Plano Nacional de Segurança Pública, no Estado do Rio de Janeiro, no período de 28 de julho de 2017 a 31 de dezembro de 2018. (Redação dada pelo Decreto de 29.12.2017)

§ 1º O emprego das Forças Armadas, nos termos do **caput** , será precedido de aprovação do planejamento de cada operação pelos Ministros de Estado da Justiça e Segurança Pública, da Defesa e Chefe do Gabinete de Segurança Institucional.

§ 2º O Ministro de Estado da Defesa definirá a alocação dos meios disponíveis.

Art. 2º Este Decreto entra em vigor na data de sua publicação.

Brasília, 28 de julho de 2017; 196º da Independência e 129º da República.

MICHEL TEMER
Torquato Jardim
Raul Jungmann
Marco Antônio Freire Gomes

Este texto não substitui o publicado no DOU de 28.7.2017 - Edição extra

Figura 11 – Decreto Presidencial que instituiu a operação de GLO em 2017. Adaptado de BRASIL(127).

Nos últimos anos, as FA vêm investindo em sistemas de simulação, nas modalidades construtiva, virtual e viva, para apoiar o adestramento da tropa (128). Além da simulação, alguns trabalhos apontam a utilização de ferramentas de ensino a distância (EAD) a fim de possibilitar a preparação intelectual do pessoal, destacando que tal medida é eficaz para redução de custos, integração do pessoal e compartilhamento de materiais (129). Nesse aspecto, um exemplo de trabalho que explora o ambiente de simulação baseado em realidade virtual é o trabalho de Doneda e Oliveira(130). Nesse trabalho, o objetivo é otimizar o adestramento combinando algoritmos aplicados em realidade virtual com métodos de reconhecimento de gestos através de AM (130), demonstrando, assim, a viabilidade e as oportunidades de emprego de soluções que caminhem nessa mesma direção.

3 TRABALHOS RELACIONADOS

Os estudos sobre obtenção de conhecimento a partir de dados textuais não-estruturados vêm sendo conduzidos ao longo dos anos e têm motivado diversos trabalhos na literatura. Em geral, as pesquisas vêm dando ênfase aos problemas relacionados aos mecanismos semânticos que podem ser utilizados sobre esses textos com o intuito de extrair informações úteis.

Alguns autores defendem o uso de algoritmos de Aprendizado de Máquina (AM), com ênfase nas técnicas de PLN. Como mencionado, com o uso de PLN é possível realizar, por exemplo, tarefas de reconhecimento de entidades nomeadas e extração de relações semânticas a partir de textos. Contudo, há abordagens que implementam formas distintas de se realizar essas tarefas de PLN para obtenção de conhecimento, como veremos adiante.

As questões ligadas à obtenção de conhecimento não ficam restritas ao uso de Aprendizado de Máquina (AM). O conhecimento autocontido adquirido em um domínio pode ser expandido através de mecanismos que favoreçam a sua interligação a outros contextos. Nesse sentido, alguns autores defendem o uso dos KG, não somente para interligar domínios, mas também para permitir a realização de inferências sobre esses dados (19). Cabe destacar que há uma diversidade de abordagens com propostas distintas que fazem uso de KG, como veremos adiante.

Neste capítulo, são apresentados os principais trabalhos relacionados à tese. Na seção 3.1, são detalhados alguns pontos acerca da estratégia de busca dos trabalhos e os critérios de seleção e descarte adotados. Na seção 3.2, são apresentadas as análises sobre os trabalhos relacionados, incluindo alguns comentários de pontos relevantes à pesquisa. Por fim, na seção 3.3, são apresentados os critérios de avaliação, destacando pontos em comum com esta tese.

3.1 Revisão da Literatura

As buscas por trabalhos relacionados na literatura foram realizadas nas bases de dados da Scopus, Google Scholar e Periódicos da Capes, compreendendo o período de 2018 a 2025[1], utilizando a combinação de termos relevantes para a pesquisa com base em seus títulos e palavras-chaves. As bases de dados utilizadas e as *strings* de busca estão apresentadas no Quadro 2, incluindo o quantitativo de trabalhos retornado. Na *string* de busca, foram destacados alguns termos, como "*natural language processing*" e "*knowledge graph*", que compõem fragmentos de um conjunto de conhecimentos amplos e alinhados

[1] Cabe ressaltar que antes do ano de 2018, as abordagens propostas não utilizavam ML baseados na arquitetura *Transformer*.

ao tema principal da tese. Além disso, foram incluídos outros fragmentos, como "*domain model*" e "*conceptual modeling*" para lidar com trabalhos que adotaram PLN e KG no apoio à geração de DM, e "*distance supervisor*" para buscar trabalhos que adotaram mecanismos para minimizar a anotação manual na realização do ajuste fino em um ML.

Quadro 2 – Estratégias de busca empregadas

Base de dados	**String principal de busca (Título, resumo, palavras-chave)**	**Filtros adicionais**	**Qtd.**
Scopus	(NLP OR "natural language processing" OR "text-based" OR "unstructured data" OR "text mining") AND ("knowledge graph" OR "knowledge graph" OR "domain model" OR "conceptual modeling" OR "distance supervisor")	Palavras-chave: "knowledge graph", "natural language processing"	132
Google Scholar			201
Periódicos Capes			41

Com base no retorno das pesquisas nas bases de dados, a seleção dos trabalhos foi conduzida a partir da leitura dos títulos e resumos. Como a utilização dos termos "extração de informação" e "grafos de conhecimento" é variada, inicialmente foram avaliados os trabalhos alinhados ao tema proposto.

A consulta baseada na *string* de busca retornou um total de 374 artigos. Além do critério de tempo de publicação, foi verificada a qualificação do artigo. A princípio, optou-se por restringir os trabalhos qualificados de acordo com o grau de Qualis, A ou B, definido pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), elencados na plataforma Sucupira[2]. Todavia, dado o caráter de inovação do tema, algumas exceções foram feitas.

Além da *string* de busca, foram adotados outros critérios com o objetivo de colaborar com o resultado da análise dos trabalhos. Nesse caso, foram descartados 108 trabalhos que não atendiam aos critérios de número de citações, nível de contribuição dos autores e fóruns de publicação. Em relação ao número de citações, é importante salientar que em trabalhos mais novos esse critério não foi adotado. Por outro lado, o número de citações é importante para avaliar a relevância e o impacto dos trabalhos na comunidade acadêmica. Um exemplo disso é o artigo "Bert: Pre-training of deep bidirectional transformers for language understanding" (12) que possui mais de 146 mil citações.

Após as análises, identificou-se que entre os 266 artigos, cerca de 40% eram repetidos, os quais resultaram em 106 trabalhos. Uma parte deles não estava alinhada ao tema de pesquisa ou não atendia aos critérios definidos. Um dos esforços para incluir os artigos, envolveu a busca por trabalhos que usaram alguma estratégia relevante de anotação

[2] https://sucupira.capes.gov.br/sucupira/

de textos, bem como a implementação de algumas estratégias que visam minimizar a necessidade de anotação manual, como explicitado nos trabalhos de Gao et al.(84) e Soares et al.(57). Porém, 23 trabalhos foram descartados por não apresentarem estratégias relevantes, restando 83 artigos.

Um ponto a ser observado é que alguns trabalhos abordam a geração de textos a partir de grafos de conhecimento. Apesar do presente trabalho estar direcionado à geração de KG a partir de textos doutrinários, durante as análises foram observados alguns trabalhos que fazem o contrário, como exemplo, o Wang et al.(131). Isso ocorre em função da dificuldade dos motores de busca entenderem que o foco é retornar somente a geração KG a partir de textos e não o contrário. Apesar de interessantes, os trabalhos que extraem textos a partir de KG foram descartados, restando 65 artigos.

Há trabalhos que propõem a geração de KG com apoio de ML e obtiveram resultados promissores. Entretanto, não foram relacionados ao trabalho proposto, tendo em vista que eles não exploram aspectos de representação do conhecimento de acordo com os objetivos deste trabalho, como por exemplo o uso de RDF. Deste modo, foram descartados 30 trabalhos. Um desses trabalhos é o Weston et al.(24), que, embora tenha sido utilizado em um experimento da abordagem IDEA-C2, como detalhado na seção 6.1, não foi selecionado como um dos trabalhos relacionados.

Finalmente, após a adoção da *string* de busca, dos critérios e filtros adicionais, a análise dos trabalhos relacionados concentrou-se em 24 trabalhos. Dada a multidisciplinaridade que trata esta tese, os esforços foram direcionados em abordagens alinhadas ao tema com o objetivo de obtenção de conhecimento, em especial a geração de KG, incluindo ML ajustados ao contexto e suas possíveis aplicações.

3.2 Análise dos trabalhos relacionados

Nesta seção, é apresentada a análise dos trabalhos relacionados ao tema desta tese. A análise se concentra nos trabalhos selecionados na revisão da literatura, combinando as perspectivas de análise com os critérios de comparação. Para organizar a forma de apresentação, os trabalhos são classificados de acordo com as perspectivas de análise, listadas na subseção 3.2.1. Enquanto que na subseção 3.2.2, são definidos os critérios para analisar individualmente cada trabalho. Finalmente, nas subseções 3.2.3, 3.2.4 e 3.2.5, cada trabalho é apresentado, destacando a perspectiva de análise e os critérios de comparação através de avaliações de sua aplicação e resultados alcançados.

3.2.1 Perspectivas de análise

As hipóteses de pesquisa são proposições que descrevem conjecturas que devem ser demonstradas em experimentos e validadas por meio de seus resultados, como apresentado

no Capítulo 1. Além disso, as hipóteses norteiam as perspectivas de análise utilizadas neste trabalho. Nesse sentido, foram definidas três perspectivas de análise, detalhadas a seguir, com o objetivo de facilitar a identificação do trabalho, organizar as ideias e a utilizar os critérios de comparação para avaliar cada trabalho.

A perspectiva **Geração de KG a partir de textos com apoio de ML** tem como objetivo apresentar os trabalhos que implementam abordagens de geração de KG a partir de corpora de textos, utilizando técnicas e tarefas de PLN. Além disso, essa perspectiva é relacionada às hipóteses H1, no que diz respeito à proposição de geração do KG, e H2, quando se refere às metacategorias utilizadas para descrever entidades e relações, as quais são estendidas como construtos do KG. Maiores detalhes são apresentados na subseção 3.2.3.

Por outro lado, a perspectiva de análise **Anotação de corpus para ajuste fino de ML** explora abordagens de anotação de corpus, destacando técnicas que minimizam a anotação manual, em especial a adoção de métodos supervisionados à distância (31). Essa perspectiva está relacionada à hipótese H3, no que diz respeito à aplicação do metamodelo combinado com regras heurísticas e Recursos Semânticos (RS) para pré-anotar um corpus. Maiores detalhes são apresentados na subseção 3.2.4.

Finalmente, a perspectiva de análise **Apoio na geração de modelos de domínio** está relacionada à hipótese H4 e apresenta abordagens que apoiam a construção de um Modelo de Domínio (DM) a partir de linguagem natural, com o suporte de um ML, discutindo as estratégias adotadas, os tipos de ML utilizados, os domínios de aplicação e seus resultados. Maiores detalhes são apresentados na subseção 3.2.5.

3.2.2 Critérios de comparação

Os critérios de comparação representam as características relevantes dos trabalhos relacionados e favorecem a avaliação de cada abordagem. Os critérios foram estabelecidos por meio de observação crítica dos trabalhos. Assim, o objetivo é definir um critério individualizado que descreva parte da especificação da abordagem proposta. Dessa forma, os critérios foram estruturados em, no máximo, dois níveis para deixar claro o contexto em que eles são aplicados e uma breve descrição. A seguir é detalhado cada critério de comparação.

- **Extração da informação**
  - **Tarefa**: Tem como objetivo identificar os tipos de tarefas de extração de informações utilizadas em cada trabalho relacionado. As técnicas de Extração de informação (EI) não se limitam àquelas mencionadas nas Subseções 2.1.1 e

2.1.4, podendo ser, por exemplo, caracterizada por busca textual, *entity link*, etc.

- **Método**: Tem como objetivo identificar o modo em que as tarefas foram implementadas, discriminando o tipo de aplicação, os algoritmos e a forma com que o método foi desenvolvido. Por exemplo, o reconhecimento de entidades nomeadas pode ser desenvolvido através de análise gramatical, utilizando, *Dependency Tree syntactic* ou *Part-of-Speech*, dentre outros.
- **Framework**: Este critério representa a biblioteca de *software* que foi utilizada para a implementação do método de EI, colaborando na identificação do componente utilizado para implementar o método, descrito no item anterior. Por exemplo, as bibliotecas SpaCy[3] e Hugging Face[4] são comumente utilizadas para tratar rotinas em PLN por possuir um conjunto robusto de componentes.

- **Representação do conhecimento**
  - **Grafo**: Abrange as formas de representação do conhecimento, identificando nos trabalhos relacionados qual o tipo de representação adotada, como por exemplo: Resource Description Framework (RDF), *Property Graph* e Ontologia, como mencionando na seção 2.5.
  - **Abordagem**: Define o tipo de abordagem de representação do conhecimento, identificando, principalmente, a estratégia de formação. Como observado por Guizzardi, Pastor e Storey(15), as abordagens *Theory-Driven (TD)* e *Data-Driven (DD)* são utilizadas para representar a forma com que o conhecimento foi constituído. Ao definir previamente um esquema, por exemplo, pode-se afirmar que a abordagem é *TD*. Caso contrário, quando o aprendizado é realizado pelos próprios dados por meio, provavelmente, de um algoritmo de AM é classificada como *DD* (15). Contudo, pode ser que ocorra a combinação de ambas as abordagens em um mesmo trabalho relacionado, sendo nesse caso definida como híbrida.
  - **Categorias**: As categorias são estabelecidas como **flexível** quando a abordagem não requer categorias predefinidas, podendo estas serem inferidas a posteriori. Enquanto no outro caso, quando a abordagem exige categorias **predefinidas**, estas devem ser previamente estabelecidas, como *local* e *organização*, para anotar “Roraima” e “ONU”, respectivamente.

- **Anotação de textos**
  - **Tipo**: Este critério abrange as formas que um texto do corpus pode ser anotado. É importante identificar se o processo de anotação foi realizado manualmente,

[3] https://spacy.io/
[4] https://huggingface.co/

semiautomatizado ou automatizado, como mencionado na subseção 2.1.2. A depender do tipo de anotação utilizada no trabalho relacionado, pode indicar a estratégia e parte do objetivo do trabalho. Por exemplo, nos casos em que a anotação é manual, pode indicar que há um corpo robusto de pessoas e que o foco do trabalho de anotação é a qualidade em detrimento da quantidade de termos anotados.

  - **Rotulagem**: Define o tipo de rotulagem (e.g. Multicategoria, Categoria única e Híbrida) que foi utilizada no trabalho. A rotulagem do corpus é definida como Multicategoria, quando o autor determina mais de uma categoria (e.g. local, organização, etc.). Comumente, os trabalhos relacionados utilizam esse tipo de conceituação (132, 133, 29). Em contrapartida, a anotação de categoria única, como o próprio nome diz, envolve a definição de uma única categoria. Esse tipo de anotação é aplicada em alguns casos e cobre múltiplos propósitos (134). Por fim, o híbrido é definido quando o trabalho implementa ambos os tipos de rotulagem.
  - **Técnica**: Define a técnica de anotação implementada no trabalho. Há casos em que o autor adotou um método probabilístico, como por exemplo, Conditional Random Field (CRF), heurístico, supervisão à distância, dentre outros.
  - **Modelo de IA**: Define o modelo de IA (e.g. LSTM, SciBERT, BERT, Generative Pre-trained Transformers (GPT), etc.) utilizado no experimento do trabalho, como mencionado na seção 2.2.

- **Domínio**: Define o domínio em que o trabalho relacionado realizou os experimentos. Esse critério é importante para mapear possíveis abordagens de solução, amplamente utilizadas, como ocorre na área biomédica através do BioBERT (21).

Nas subseções a seguir, os trabalhos relacionados são especificados, contendo o resumo, a aplicação da abordagem e a análise crítica do trabalho em relação aos objetivos da tese. Cabe destacar que os trabalhos são mencionados uma única vez, não havendo sobreposição.

3.2.3 Geração de Knowledge graph a partir de textos com apoio de modelo de linguagem

Nesta subseção, os nove trabalhos são organizados através do método de mineração de texto adotado e classificado em três grupos. No primeiro grupo, são destacadas as abordagens que adotaram métodos de mineração tradicionais (e.g. modelos probabilísticos ou *Dependency Tree syntactic* (DTs) ou *Part-of-Speech* (PoS)). No segundo, constam os trabalhos que incorporaram Modelos de Linguagem (ML) generalistas baseados em

Transformer, como BERT e GPT. No terceiro grupo, são destacados pormenorizados os trabalhos que adotaram ML ajustados ao contexto. Ao término, é apresentado o Quadro 3 que reúne todos os trabalhos relacionados desta perspectiva de análise.

No primeiro grupo, o trabalho de Rios-Alvarado et al.(135) relata a geração de um KG em RDF, contendo 5 mil triplas, com o apoio de PLN, através de regras de expressão regular e PoS, explorando textos não-estruturados no idioma espanhol, nos domínios de ciência da computação e finanças. O *pipeline*[5] é composto de tarefas das NER, RE e *Entity Linking* (EL). Na tarefa NER, os resultados foram distribuídos pelos conjuntos de entrada (KI, CS e FN) e alcançaram, respectivamente, 459, 291 e 4967 entidades. Na RE, foram alcançados 55% de precisão e 66% de *recall* nas relações entre as entidades. No trabalho de Dang, Phan e Nguyen(136), foi apresentado o **G**raph of m**E**ntal-health and **N**utrition **A**ssociation (GENA)[6], um *Property Graph* (PG) gerado com o apoio de PLN, explorando artigos da PubMed (137), nos domínios de nutrição e saúde mental através de cinco categorias de entidades e relações. A abordagem usa um modelo híbrido para lidar com tarefas NER, através do ScispaCy (138) treinado com o BC5CDR, apoiado por ontologias de nutrição da *FoodOn Ontology* (139), de termos bioquímicos da ChEBI (140), e de saúde mental através das ontologias APADIS-ORDERS, ASDTTO e MFOMD (141). Assim, GENA detecta novas relações entre os domínios que não existem no corpus BC5CDR (142), combinando PoS com DTs, além de usar uma LSTM para extrair relações existentes entre as categorias de entidades. Como resultado, GENA alcançou 94% de novas relações.

Ademais, há abordagens que apostam em conteúdos como livros didáticos e documentos regulatórios em função da padronização e qualidade dos textos. Como ocorre em Liu et al.(25), que um KG de 1.300 triplas foi gerado com o apoio de PLN, combinando tarefas de NER e RE com aprendizado por reforço, a partir de textos de um livro didático[7] da área de aviação. Como resultado, o desempenho do modelo alcançou, na métrica F1-score, 89% para entidades e 91% para relações. Finalmente em Zhao, Huang e Ding(26), é gerado um KG, a partir de textos de regulamentos militares da República Popular da China. Para a anotação, é utilizado um método estatístico de segmentação de palavras combinada com DTs para tarefa de NER. Na tarefa de RE, é utilizado o CRF e o PoS para indicar a classe gramatical entre as entidades.

Em resumo, apesar dos resultados, esses trabalhos são limitados à estrutura dos textos, ao domínio de aplicação e ao conjunto de categorias predefinidas. Como alternativa, podem ser utilizados ML baseado em *Transformer* combinado com uma abordagem flexível de categorias.

[5] Uma sequência ordenada e automatizada das etapas de treinamento ou ajuste de um ML.
[6] https://github.com/ddlinh/gena-db
[7] Aviation Manufacturing Engineering Manual: Aircraft Assembly

No segundo grupo, com a adoção da arquitetura *Transformer*(83), surgiram Modelos de Linguagem (ML) generalistas, como BERT e GPT, com maior capacidade semântica e treinados com um conjunto vasto de dados. No trabalho de Trajanoska, Stojanov e Trajanov(143), é gerado um KG a partir de textos do corpus News API[8], com foco na área de sustentabilidade, utilizando o GPT e o Relation Extraction By End-to-end Language generation (REBEL) (144). Para a anotação do corpus, é utilizada uma estratégia híbrida (manual e semiautomatizada), incluindo pré-anotação e curadoria humana. Como os autores não detalham os resultados das métricas, as análises comparativas são prejudicadas. No trabalho de Parović, Li e Du(145), é gerado um KG nos domínios de Inteligência Artificial (IA) e Biomédico, explorando a capacidade do GPT-3.5-Turbo. O LLM foi implementado via API da OpenAI, através de *prompt-based extraction*, com o objetivo de representar o conhecimento de modo que seus dados possam ser reutilizáveis. Os resultados alcançaram uma precisão de 84% nas relações no domínio de IA e 76% no biomédico. Finalmente, em Silveira e Cavalcanti(146), é gerado um KG através da abordagem **P**redicate **LA**bel**IN**g (PLAIN)[9] com o objetivo de enriquecimento de *datasets*. Como a PLAIN explora um conjunto de dados local, ela utiliza a ferramenta OpenNRE (85), que faz uso do BERT, para apoiar a extração das relações nas sentenças de textos a partir de um par de $(sujeito, objeto)$.

Em síntese, apesar dos resultados, esses trabalhos são limitados a ML generalistas e suscetíveis aos problemas de alucinações e imprecisões de domínio, como observado em Saba(16). Uma alternativa é adotar ML ajustados ao domínio, como o SciBERT (34) e BioBERT (21). Outra alternativa é utilizar um ML e ajustá-lo ao domínio através de um corpus anotado.

Diferentemente dos trabalhos apresentados, há abordagens que utilizam ML ajustados ao domínio, em especial o BERT, através de corpora anotados com categorias predefinidas, minimizando os problemas de alucinação e imprecisões. Dessa forma, essas abordagens aproveitam a capacidade de aprendizado dos ML para refinar o seu raciocínio e gerar um KG com maior precisão. Sendo assim, os artigos apresentados a seguir, são detalhados individualmente a fim de contribuir com a análise comparativa desta tese.

No terceiro grupo, o trabalho de Zhu, Li e Su(147) apresenta uma abordagem de geração de KG com base na mineração de conhecimento, apoiado por técnicas de PLN, combinando o BERT e análise sintática (*syntactic parsing*), a partir de um corpus de textos de documentos regulatórios no domínio da indústria de Arquitetura, Engenharia e Construção (AEC). Devido à escassez de corpus nesse domínio, a anotação é manual e a abordagem é composta por quatro etapas. Na primeira etapa, obtém-se o conjunto de dados de entrada, incluindo *Data Augmentation* para gerar novas amostras de texto

[8] https://newsapi.org
[9] https://github.com/rafans222/plain

artificialmente (65) e Delphi para incrementar a qualidade dos conjuntos de dados do domínio (148). Além disso, um extrator de cláusulas baseado no modelo BERT pré-treinado é implementado e ajustado de acordo com os dados do domínio. Em seguida, um mecanismo de extração de constituintes foi desenvolvido com base em análise sintática, incluindo PoS para indicar a classe gramatical e análise de dependência. Na quarta etapa, é realizada a modelagem do conhecimento a partir das tuplas e atributos extraídos que são categorizados em: entidades, relações, atributos e restrições. Com base nisso, é utilizada a biblioteca RDFLib[10] para gerar o KG. Apesar da abordagem ser consistente para geração de KG, a estrutura de extração é centrada em análise gramatical, podendo levar a inconsistências em função das variações linguísticas.

Por fim, em Trappey, Liang e Lin(149), é proposta uma abordagem de geração de KG com foco na área de mineração de patentes aplicada ao domínio químico de captura de carbono. O objetivo do trabalho é colaborar com pesquisadores e engenheiros a entenderem inovações em patentes de maneira mais rápida e eficaz. Com esse intuito, um corpus contendo 879 textos de patentes é coletado do European Patent Office (EPO) no domínio *Carbon Capture and Storage* (CCS). Esse corpus é submetido a três ML distintos. Em primeiro lugar, o ALBERT(150) é utilizado para classificar parágrafos de patentes, diferenciando os químicos dos não químicos. Ele foi implementado por meio da biblioteca TensorFlow[11]. Em seguida, o KeyBERT(151) para extração de termos-chaves com o apoio do ChemDataExtractor[12] e outras bibliotecas. E, por fim, o Sentence-BERT(152) para medir a similaridade semântica entre os termos.

Os rótulos *Technology Component* (TC), *Functional Mechanism* (FM) e *Innovation Objective* (IO) são as classes atribuídas aos termos extraídos de KeyBERT e Sentence-BERT. Posteriormente, os grafos-resumo são gerados, utilizando o Cytoscape[13], alcançando 81% de informação relevante baseado na métrica *retention rate*, que mede o engajamento e a fidelização de acordo com a proporção de usuários que continuam ativos após um certo período de tempo (65). Cada grafo-resumo é composto de nós, representado pelas classes (TC, FM e IO), e as arestas são estabelecidas através das relações entre as entidades (TC-FM, FM-IO e TC-IO). Vale destacar que várias ferramentas foram orquestradas para executar os experimentos. Entretanto, o trabalho é restrito a um domínio específico e com poucos textos de patentes, limitando a sua aplicação em outros contextos.

No Quadro 3, são apresentados os aspectos abordados nesta perspectiva de análise, destacando as categorias analisadas individualmente e discutidas em cada trabalho relacionado. No geral, quase todos os trabalhos executam tarefas NER e RE, adicionalmente usa-se *Entity Link* para enriquecimento de dados (135, 143) e aprendizado por reforço

[10] https://rdflib.readthedocs.io/en/stable/
[11] https://www.tensorflow.org/
[12] http://chemdataextractor.org/
[13] https://cytoscape.org/

Quadro 3 – Trabalhos relacionados à tese sob a perspectiva de análise de geração de KG a partir de textos com apoio de ML

Ref.	**Grafo**	**Método**	**Tarefa**	**Framework**	**Domínio**
(135)	RDF	REGEX+ PoS	NER+ RE+ EL	FreeLing+ NLTK+ SpaCy+ RDFLib	Educacional
(136)	PG	PoS+DTs+ LSTM	NER+ RE	SpaCy	Nutrição e Saúde Mental
(25)	PG	HMM+ CRF+BiLSTM	NER+ RE+ AR	SpaCy	Aviação
(26)	RDF	PoS+ DTs	NER+ RE	NI	Militar
(143)	RDF	GPT-4+ REBEL	NER+ RE+ EL	*Prompt*	Sustentabilidade
(145)	PG	GPT-3.5 Turbo	NER+ RE	*Prompt*	Inteligência Artificial e Biomédico
(146)	RDF	BERT	NER+ RE	OpenNRE	Wikipedia
(147)	RDF	BERT	NER+ RE	SpaCy+ RDFLib	Arquitetura, Engenharia e Construção
(149)	NI	ALBERT+ KeyBERT+ Sentence-BERT	NER+ RE	TensorFlow+ ChemDataExtractor+ Cytoscape	Químico
Esta tese	RDF	BERTimbau	NER+ RE	SpaCy+ RDFLib	Militar

Legenda: NI - Não Informado; NA - Não se Aplica; PG - Property Graph; RDF - Resource Description Framework.

(25). Em algumas dessas publicações, foram utilizados métodos tradicionais, baseados essencialmente em análise sintática e gramatical (135, 136, 25, 26). Por sua vez, em outras publicações foram utilizados ML generalistas (143, 145, 5). Enquanto que outros trabalhos aplicaram ML ajustados ao contexto (147, 149).

A maioria dos trabalhos adotou o *framework* SpaCy[14] para lidar com PLN (135, 136, 25, 147), no entanto há algumas exceções que optaram pelo uso de *prompts* (143, 145), além de outras ferramentas (146). Em alguns desses trabalhos combinaram com o SpaCy a biblioteca RDFLib[15] para gerar o KG (135, 147). Além disso, é possível notar a extensa variabilidade de domínios. Finalmente, quase todos os trabalhos utilizam representações do conhecimento baseadas em grafo, em função da flexibilidade de estrutura de dados.

[14] https://spacy.io/
[15] https://rdflib.readthedocs.io/

Todavia, dada a diversidade de implementação de KG existentes, há trabalhos que optam por implementações baseadas em *Property Graph* (25, 136, 145), bem como em RDF (135, 26, 143, 146, 147).

3.2.4 Anotação de corpus para ajuste fino de Modelos de Linguagem

Nesta subseção, os sete trabalhos são organizados através da estratégia de anotação do corpus (manual, semiautomatizada e automatizada). No primeiro grupo, são destacadas as abordagens que adotaram ferramentas que oferecem suporte à anotação manual, como Prodigy[16]. No segundo grupo, constam os trabalhos que incorporaram técnicas semiautomatizadas, incluindo métodos probabilísticos e supervisão à distância. No terceiro grupo, são destacados os trabalhos que adotaram métodos de anotação com suporte automatizado, incluindo bases de conhecimento para apoiar os métodos supervisionados à distância. Ao término, é apresentado o Quadro 4, reunindo todos os trabalhos relacionados desta perspectiva de análise.

Como mencionado, a anotação manual de um corpus é uma atividade custosa e envolve recursos de toda ordem. Ao adotar a anotação manual, espera-se que haja alta qualidade de anotação e curadoria em detrimento de velocidade. No primeiro grupo, o trabalho de Nundloll et al.(132) apresenta uma abordagem para extrair informações de textos históricos e científicos do *Journal of Botany*, no domínio de botânica e ecologia. A abordagem adota a anotação manual, com base em seis categorias de entidades, utilizando a ferramenta Prodigy. Para extrair as entidades nomeadas, é usado um modelo estatístico através dos componentes *tagger* e *parser* da biblioteca SpaCy. Os resultados da extração são armazenados em um KG no GraphDB. Esse KG é utilizado para enriquecimento dos dados de geolocalização, utilizando a ontologia GeoSPARQL[17]. Como resultado, a abordagem alcançou 82% de acurácia, obtendo uma melhoria considerável frente aos outros modelos que não ultrapassaram 60%.

No trabalho de Zhang et al.(133), é apresentada uma abordagem que constrói o corpus anotado SciER a partir de anotações manuais de textos de artigos científicos nos domínios de IA e PLN. Basicamente, a anotação é realizada por duas ou mais pessoas com base em três categorias de entidades (e.g. *datasets*, *methods* e *tasks*) e suas relações (e.g. *used-for*, *feature-of*, *part-of*, etc.). Para garantir a qualidade da anotação, foi adotada a validação cruzada. Nos experimentos, foram utilizados os LLMs GPT-3.5 Turbo (*Zero-shot* / *Few-shot*), Llama3-70B (*Few-shot*), Qwen2-72B (*Few-shot*), e para linha de base os ML BERT e SciBERT). Como resultado dos experimentos, o SciBERT se destacou dos demais LLMs, alcançando na métrica F1-score, respectivamente, 85% para NER e 80% para RE.

Em suma, apesar dos resultados, as abordagens são limitadas em função da anotação

[16] https://prodi.gy/
[17] https://www.ogc.org/standard/geosparql/

ser manual, aumentando o custo de criação e manutenção do corpus. Além disso, ao adotar um conjunto de categorias predefinidas, limita-se a expansão da abordagem a outros domínios.

No segundo grupo, há abordagens que adotam métodos probabilísticos de anotação visando minimizar custos e acelerar a anotação. Em Chantrapornchai e Tunsakul(153), a abordagem apresentada extrai informações de documentos na Web no domínio de turismo. Para tal, são propostos dois métodos que requerem a análise léxica, sintática e semântica. O primeiro método utiliza a biblioteca SpaCy. O segundo método utiliza o BERT para ser ajustado ao domínio, a partir do corpus WordNet[18], com base em categorias de entidades e relações predefinidas. Em relação à RE, é utilizado um analisador sintático de dependências entre as entidades nomeadas a partir de métodos supervisionados à distância. Como resultado, o BERT ajustado ao domínio alcançou 99% de acurácia em comparação com o SpaCy, que alcançou 95%.

Enquanto que no trabalho de Pan et al.(134), a abordagem proposta apoia a construção de um corpus, contendo 31 mil artigos científicos e 449 mil menções de *datasets*. Para anotação, a abordagem combina métodos de supervisão à distância com a anotação manual. Para ser flexível, a abordagem implementa uma única categoria, denominada DATASET, para rotular o corpus. O objetivo da abordagem é detectar menções de dados (*dataset mentions*), inspiradas no conceito da tarefa NER, em conjuntos de dados de artigos científicos. Para lidar com o ruído da supervisão à distância, foram incluídas fases de revisões humanas sucessivas. Em relação ao treinamento, foram utilizadas instâncias de ML baseados em *Transformers*, como o BERT(12), SciBERT(34), RoBERTa(33), dentre outros. Os resultados demonstraram que o SciBERT se destacou dos demais, alcançando 89% de desempenho.

Em síntese, embora os resultados de ambos os artigos sejam promissores, há limitações em relação ao ruído da supervisão à distância, além de implicações nos métodos NER, bem como nas dependências gramaticais dos textos em virtude da análise sintática. Algumas abordagens adotaram bases de conhecimento externas que podem minimizar os problemas decorrentes do ruído. Contudo, existem formas distintas de implementação e tratamento para lidar com o ruído, como detalhado nos trabalhos a seguir.

No terceiro grupo, o trabalho de Zhou et al.(29) propõe uma abordagem que cria cenários de simulação a partir de textos do cenário operacional do domínio militar, utilizando técnicas de PLN para reconhecer entidades nomeadas com base em categorias predefinidas (e.g. força, plataforma, locais, etc.). Nesta abordagem, é usada a supervisão à distância combinada com uma base de conhecimento externa para nortear a anotação do corpus. Para o treinamento do NER, é utilizada a Rede Neural Recorrente de memória de curto e longo prazo (LSTM), a qual aprende as dependências entre elementos em

[18] https://wordnet.princeton.edu/

sequência através de conexões recorrentes. Como resultado, a abordagem alcançou uma melhoria global de 9% na métrica F1-score ao comparar com outros métodos aplicados no domínio. Apesar do resultado, o trabalho adota redes neurais, limitando seus resultados e não aborda sobre a extração de relações que em um cenário de simulação operacional é fundamental.

Por sua vez, no trabalho de Fries et al.(30), a abordagem Trove realiza o ajuste fino BioBERT(21) através de seis níveis de supervisão à distância, combinando ontologias biomédicas com um modelo probabilístico para anotar automaticamente as categorias de entidades (e.g. doenças e distúrbios, procedimentos clínicos, fármacos e substâncias químicas, etc.). Para o ajuste fino, são utilizados textos de prontuários clínicos, além dos corpora BC5CDR, NCBI-Disease, JNLPBA, Species-800, CHEBI, e Biosses. Como resultado, a abordagem alcançou valores de desempenho aproximados, em torno de 90% a 95%. Apesar do resultado, como o trabalho combina várias ontologias, pode haver ruídos em função de sobreposições e relações ambíguas.

Finalmente, em Kim, Görz e Geisler(154), a abordagem KONDA realiza a anotação semântica de *datasets* de pesquisa com o apoio do GPT-4 a partir de textos de documentos de pesquisa. Além disso, KONDA gera um grafo de conhecimento em RDF e utiliza o LLM para sugerir anotações semânticas com foco na simplificação do processo. Os resultados demonstraram que KONDA alcançou uma precisão média superior a 80% na identificação de entidades e relações em *datasets* de pesquisa distintos. Em tarefas de mapeamento semântico com ontologias, houve uma leve queda na precisão, em torno de 70% a 75%, dadas as ambiguidades conceituais. Após a intervenção manual dos curadores, a qualidade final de KONDA subiu para mais de 90% de acertos. Em resumo, apesar dos resultados, o trabalho é limitado à interpretação do GPT-4, dificultando a explicabilidade de suas decisões. Como há restrições de tamanho da janela de contexto do LLM, não é possível submeter muitos documentos.

No Quadro 4, são apresentados os aspectos abordados nesta perspectiva de análise, destacando as categorias analisadas individualmente e discutidas em cada trabalho relacionado. Na maioria dos casos, os trabalhos adotam o tipo de rotulagem baseada em multicategoria (*multicategory*). Porém, há casos em que a categoria única pode flexibilizar a anotação (134) e favorecer outras aplicações. Quando observamos o método de anotação, na maioria dos trabalhos foi adotada a supervisão à distância para tornar a atividade menos custosa. Contudo esse método não é aplicado em anotações tradicionais (132). Os métodos supervisionados à distância podem ser potencializados a partir da combinação com ontologias (29, 145) e *Prompts* (154) com o objetivo de minimizar o ruído.

Por sua vez, em relação ao tipo de anotação, há abordagens que adotaram métodos manuais que provêm alta qualidade, todavia com alto custo (132, 133). Nos trabalhos de Chantrapornchai e Tunsakul(153) e Pan et al.(134) foi adotado o tipo semiautomatizado,

Quadro 4 – Trabalhos relacionados à tese sob a perspectiva de Anotação de corpus para ajuste fino de Modelos de Linguagem

Ref.	**Rotulagem**	**Técnica**	**Tarefa**	**Tipo**	**Modelo de IA**	**Domínio**
(132)	Multi	Estatístico (*parser* e *tagger*)	NER+ EL	MN	SpaCy+ GraphDB+ GeoSPARQL	Ecologia e Botânica
(133)	Multi	*Prompt* zero-shot few-shot	NER+ RE	MN	GPT-3.5 Turbo Llama3-70B Qwen2-72B BERT SciBERT	IA
(153)	Multi	Supervisão à distância	NER+ RE+ AR	SAT	BERT e SpaCy	Aviação
(134)	Categoria única	Supervisão à distância	NER+ RE	SAT	BERT SciBERT RoBERTa	Artigos científicos
(29)	Multi	Supervisão à distância+ Base de conhecimento	NER+ RE	AT	Rede Neural LSTM	Militar
(145)	Multi	Supervisão à distância+ Ontologias	NER+ RE	AT	BioBERT	Biomédico
(154)	Multi	Supervisão à distância+ Ontologias+ Prompt	NER+ RE+ EL	AT	GPT-4	Pesquisa Científica
Esta tese	Categoria única e Multi	Supervisão à distância+ Recursos Semânticos+ Heurística	NER+ RE	SAT	Bertimbau SciBERT* RoBERTa*	Militar e outros domínios

Legenda: NI - Não Informado; NA - Não se Aplica; AT - Automatizado; MN - Manual; SAT - Semiautomatizado; Multi: Multicategoria; IA - Inteligência Artificial.

evidenciando a qualidade e a quantidade de termos anotados, entretanto esses trabalhos não apuram os resultados alcançados com essa estratégia. Outros trabalhos adotaram o tipo de anotação automatizado, pois ele prevê alta quantidade de termos anotados com baixa intervenção humana (29, 145, 154). Finalmente, nota-se uma variabilidade de modelos de linguagem, em destaque aos baseados na arquitetura *Transformer*, com exceção ao trabalho de Zhou et al.(29) que utilizou uma Rede Neural Recorrente de Memória de Curto e Longo prazo (LSTM). Além disso, outra variabilidade foi destacada nos domínios de aplicação, demonstrando como esses modelos podem ser ajustados a diversos contextos.

3.2.5 Apoio na geração de modelos de domínio

Há aplicações diversas sobre ML seja para extrair informações, reconhecer entidades nomeadas, extrair relações entre entidades e até interligar entidades a repositórios externos com objetivo de enriquecer seus dados. No entanto, há tempos que a atividade de construir Modelo de Domínio (DM) a partir da linguagem natural tem sido um desafio. Porém, com os avanços recentes de PLN, em especial os ML baseados na arquitetura *Transformer*, foi possível desenvolver abordagens capazes de apoiar essa atividade. Entretanto, como esses modelos são subsimbólicos, há riscos de suas respostas serem influenciadas por vieses ou até alucinações dada a sua natureza de aprendizagem, impactando os resultados da modelagem (16, 15).

Nesta subseção, são abordados os nove artigos relacionados que propõem a construção de DM a partir da submissão de textos a um ML, discutida na seção 1.4. Como há um número considerável de artigos, primeiramente optou-se por agregar alguns deles de menor impacto relacionado ao tema, consolidando a sua análise, avaliação e discussão. Por outro lado, os demais artigos com alta relevância são analisados individualmente. Ao término, é apresentado o Quadro 5, reunindo todos os trabalhos relacionados desta perspectiva de análise.

Os primeiros trabalhos usavam o PLN para identificar conceituações do domínio com base em regras e métodos heurísticos de dependências sintáticas. Geralmente, usando marcadores *Part-of-Speech* ou *PoS tagging* a fim de gerar diagramas conceituais a partir de textos de especificação de requisitos de software escritos em linguagem natural (155, 156, 157). Em Lucassen et al.(155), um experimento controlado explorou textos de *user stories* a partir de técnicas de PLN com base em onze regras heurísticas e alcançou resultados em torno de 92 a 97% de precisão e de 88 a 98% de *recall* na identificação de entidades e relações.

No trabalho de Shweta, Sanyal e Ghoshal(156), é proposta uma abordagem em duas fases que extrai diagramas de classe através de descrições de casos de uso a partir de doze regras linguísticas (e.g. substantivos, adjetivos, verbos, etc.), explorando termos em sentenças, com o objetivo de mapear construtos (e.g. classes, relações, atributos e métodos). Na primeira fase, a descrição do caso de uso é convertida em um modelo intermediário através das regras predefinidas. Na segunda fase, são aplicadas regras de identificação dos construtos no modelo intermediário, utilizando técnicas de PLN, através da ferramenta Stanford CoreNLP[19]. Ao aplicar a abordagem em três estudos de casos, os autores compararam cada construto com modelos de domínio, classificados como *gold standard* e criados por humanos. Como resultado, a abordagem alcançou uma precisão média de 85% para classes e 86% para relações.

[19] https://stanfordnlp.github.io/CoreNLP/

Ademais, há casos em que um só método não resolve o problema, como em Ramzan et al.(157), que é proposto um modelo híbrido que combina a abordagem baseada em regras e PLN com algoritmos de AM. As regras são aplicadas para extrair classes, atributos e métodos a partir de textos de requisitos de software. Para identificar os tipos de relacionamento entre classes (associação, agregação, composição, herança) são utilizados quatro algoritmos de AM. O primeiro é o Support Vector Machine que é usado para categorizar os relacionamentos entre classes. Já o Random Forest é empregado como modelo comparativo. Enquanto que o Naive Bayes realiza a análise experimental. Enfim, o algoritmo de Regressão Logística mede o desempenho. Como resultado, o método alcançou 95% de acurácia.

Em síntese, apesar dos resultados desses trabalhos serem promissores, as abordagens baseadas em regras são estáticas, acopladas ao domínio e dependem da qualidade dos textos de entrada para terem um bom desempenho.

Posteriormente, surgiram abordagens para atender multidomínios, usando engenharia de *prompts* através de LLMs de propósito geral, i.e., sem recorrer a *fine-tuning* ou conjuntos de dados de treinamento. Em trabalho recente, Prokop et al.(158) combinaram o GPT-4, utilizando Recuperação de Geração Aumentada (RAG), para fornecer sugestões aos usuários de modelos de domínio a partir de descrições textuais. A avaliação das sugestões foi realizada por seres humanos, comparando cada construto sugerido com os modelos de domínio pré-concebidos. Embora haja sugestões de modelos, os resultados são limitados, requerendo intervenção humana para ajustar as respostas, inclusive há relatos no trabalho de alucinações do ML com sugestões fora do domínio.

No trabalho de Fill, Fettke e Köpke(159), as interações com o ChatGPT são executadas com o objetivo de apoiar tarefas de modelagem conceitual. Após dar entrada no *prompt* com o texto correspondente a um minimundo, o usuário envia uma instrução para gerar o modelo de domínio no formato JSON. Apesar dos resultados, os autores indicam que os DM gerados ainda exigem revisão humana dadas as imprecisões nos construtos (e.g. classes inexistentes, associações incorretas, etc.).

Finalmente, em Chaaben, Burgueno e Sahraoui(160), é proposta uma abordagem *few-shot prompt learning* (poucas tentativas), utilizando o GPT-3, que mapeia os elementos de modelos (e.g. classes, atributos, associações, etc.) a partir de textos a fim de completar os elementos de modelos parciais. Os resultados demonstraram que a abordagem alcançou um desempenho razoável, em destaque a precisão e o *recall* que atingiram, respectivamente, 57% e 45% para classes. Contudo, houve resultados que variaram entre 0 e 10% de recall para domínios não conhecidos. Embora os resultados sejam promissores, a abordagem é dependente do *prompt* construído e os resultados variam de acordo com a mudança de domínio.

Em resumo, apesar desses trabalhos utilizarem LLMs de propósito geral sem

ajustá-los a um domínio específico, os resultados são limitados, ruidosos e imprecisos, algo característico das abordagens baseadas em *Data-Driven (DD)*. Por outro lado, existem abordagens que utilizam ML pré-treinados e, em alguns casos ajustados ao domínio, para apoiar a tarefa de modelagem conceitual a partir de textos em linguagem natural (161, 162). Embora esses modelos não superem completamente as limitações da abordagem DD, as evidências demonstram que a adaptação a um domínio, seja por um ML pré-treinado ou ajustado, por intermédio de um corpus anotado com categorias predefinidas, pode minimizar os reflexos característicos da abordagem subsimbólica. Sendo assim, foram selecionados alguns artigos, os quais são detalhados individualmente para uma análise comparativa, apresentada a seguir.

No trabalho de X, Mittal e Chauhan(161), é proposta uma abordagem de extração de elementos de diagramas de classes a partir de textos de requisitos de software com o apoio de um ML. O objetivo do trabalho é extrair com maior precisão os construtos (e.g. classes, atributos, métodos e relacionamentos) comparado com as abordagens baseadas em regras. Para tal, os autores utilizaram um corpus, contendo trinta e quatro mil sentenças de textos, anotado manualmente através da ferramenta General Architecture for Text Engineering (GATE)[20], com seis categorias correspondentes aos elementos de diagramas de classes com o intuito de ajustar um ML ao domínio nas tarefas de NER e RE. Basicamente, o método utiliza o ML RoBERTa(33) para gerar *embeddings* contextuais dos termos do texto através de um *pipeline* de extração. Esse *pipeline* utiliza hiperparâmetros ajustados (e.g. *learning rate*, *weight decay* e *max-steps*) combinado com a avaliação de diferentes otimizadores (e.g. *adam_torch*, *adam_hf* e *adam_torch_fused*). A definição dos hiperparâmetros foi realizada através de execuções exaustivas para definir os melhores valores. Apesar de RoBERTa ter alcançado 89% de acurácia, os autores também realizaram experimentos com o BERT-base(12) e o DistilBERT(32), que alcançaram resultados aproximados, respectivamente, de 74% e 76% de acurácia. Ao comparar com o modelo baseado em regras, os resultados demonstram que houve uma melhoria na extração em média de 7% nas classes, 9% nos atributos, 5% e 1,5% nos métodos e relações. Em síntese, apesar dos resultados, o método é limitado, depende de requisitos bem formulados e de dados de treinamento bem formatados, além de não capturar imprecisões nos textos das especificações de software.

Por fim, em Babaalla, Jakimi e Oualla(162), análogo ao trabalho anterior, é proposto um *pipeline* que emprega ML baseado em *Transformers* (e.g. BERT, RoBERTa, XLNet, SpanBERT, MiniLM e ELECTRA) ajustado ao domínio para extrair construtos (e.g. classes, atributos, métodos, associação, agregação, composição e herança) a partir de textos de requisitos de software. Esses construtos são estruturados em uma *Domain Specific Language (DSL)* intemediária. A DSL é uma linguagem de modelagem para expressar um domínio (106). Em seguida, é gerado o diagrama de classes e, por fim, produzido

[20] https://gate.ac.uk/

o código-fonte. O corpus é composto de textos do domínio e anotado com categorias baseadas nos construtos de acordo com o padrão *Inside, Outside, Beginning* (IOB). Para o ajuste fino, foi utilizada a biblioteca HuggingFace Transformers[21], o otimizador AdamW e os hiperparâmetros *learning rate* com 2e-5 e *batch size* de 16. Além disso, cada ML foi instanciado juntamente com o corpus anotado. Ao comparar cada ML ajustado, os resultados demonstram que houve uma acurácia significativa, considerando o melhor valor alcançado na métrica F1-Score de 98,6% nas classes com o XLNet. Contudo, nos demais construtos com o BERT, os resultados alcançaram entre 95 a 98% . Embora os resultados sejam promissores, ainda há limitações em função da baixa adaptabilidade a contextos reais e da qualidade dos corpora.

Quadro 5 – Trabalhos relacionados à tese sob a perspectiva de geração de modelos de domínio

Ref.	**Abordagem**	**Método**	**Categoria**	**Framework**	**Domínio**
(155)	DD	RH+PLN	Predefinido	SpaCy	Engenharia de Software
(156)	DD	RH+PLN	Predefinido	Stanford CoreNLP	Engenharia de Software
(157)	DD	RH+PLN +AM	Predefinido	SpaCy	Engenharia de Software
(158)	DD	GPT-3 e 4	Flexível	ChatGPT	Variados
(159)	DD	GPT-3 e 4	Flexível	ChatGPT	Variados
(160)	DD	GPT-3	Flexível	ChatGPT	Variados
(161)	DD + Fine-tuning	RoBERTa BERT-base DistilBERT	Predefinido	NI	Engenharia de Software
(162)	DD + Fine-tuning	BERT RoBERTa XLNet SpanBERT MiniLM ELECTRA	Predefinido	HuggingFace Transformers	Engenharia de Software
Esta tese	Híbrida	BERTimbau	Flexível	SpaCy + RDFLib + SPARQL	Militar

Legenda: NI - Não Informado; NA - Não se Aplica; DD - Data-Driven; TD - Theory-Driven.

No Quadro 5, são apresentados os aspectos abordados nesta perspectiva de análise, destacando as categorias analisadas individualmente e discutidas em cada trabalho relacionado. No geral, cabe destacar que todos os trabalhos relacionados utilizam a abordagem Data-Driven (DD). Porém, há trabalhos que incorporaram o *fine-tuning* em seu *pipeline*

[21] https://huggingface.co/docs/transformers/index

com o intuito de minimizar os problemas característicos (e.g. alucinações, imprecisões, etc.) dessa abordagem (161, 162).

Em relação ao método aplicado, alguns trabalhos combinam regras heurísticas com PLN, explorando relações sintáticas dentro dos textos (155, 156), incluindo algoritmos de AM para refinar o método e consequentemente melhorar os resultados (157). Contudo, como o método de regras heurísticas é restrito aos padrões gramaticais e ao domínio aplicado, há trabalhos que para superarem essas restrições, utilizam ML baseados em *Transformer* tanto generalistas (158, 159, 160) quanto pré-treinados (161, 162). As restrições de aplicação também são influenciadas pelo tipo de categoria selecionada. Há trabalhos que adotaram categorias predefinidas, restringindo os resultados da geração do DM (155, 156, 157). Em contraste, outros trabalhos permitem o uso de categorias flexíveis, alcançando resultados mais abrangentes, porém com menor precisão (158, 159, 160).

Por sua vez, em relação aos *frameworks*, há trabalhos que utilizaram o SpaCy (155, 157), principalmente onde o uso de PLN foi mais evidente, com exceções à aplicação de Stanford CoreNLP (156) e HuggingFace (162). Por fim, os trabalhos relacionados são aplicados em domínios variados, essencialmente com LLMs generalistas e sem uso de *fine-tuning* (158, 159, 160). No entanto, chamou a atenção que 62% dos trabalhos analisados foram aplicados no domínio de Engenharia de Software, evidenciando uma alta demanda dessa área (155, 156, 157, 161, 162).

Nesta seção foi apresentada a análise dos trabalhos relacionados combinando as perspectivas de análise com as categorias de comparação. Nesse sentido, foram destacados os pontos principais de cada trabalho com base nos objetivos desta tese. Na próxima seção, os trabalhos analisados são listados e comparado-os com a abordagem proposta.

3.3 Considerações finais sobre os trabalhos relacionados

Nesta seção, as abordagens analisadas na seção anterior são comparadas com a abordagem proposta neste trabalho a partir da combinação das perspectivas de análise e dos critérios de comparação apontados descritos nos Quadros 3, 4 e 5. Para tal, é destinado um resumo para cada perspectiva de análise abordando os pontos convergentes e divergentes à abordagem proposta, destacando algumas de suas características e diferenciais de implementação.

Ao considerarmos a perspectiva de análise **Geração de KG a partir de textos com apoio de modelo de linguagem** destacada no Quadro 3, podemos observar que há alguns pontos convergentes com a abordagem proposta, como a adoção do *framework* SpaCy, praticamente uma unanimidade no universo analisado para lidar com PLN, assim como o RDF para representação de conhecimento, principalmente por ser flexível e extensível a diversos domínios. Além disso, destaca-se a adoção da biblioteca RDFLib para manipulação

dos dados no grafo e, finalmente, a aplicação das tarefas NER e RE que contribuem para a formação dos nós e arestas do KG. Entretanto, há alguns pontos de divergência com a nossa abordagem, principalmente com a definição de categorias predefinidas para ajustar o ML. Acreditamos que ao adotar um metamodelo somente com as metacategorias para formação do KG, faz com que a nossa abordagem torne-se mais flexível que as demais, bem como essa estratégia favorece a extensão para outros domínios. Por fim, em relação ao método, adotamos o BERTimbau ajustado ao domínio militar em função dos textos de nosso estudo de caso serem em Língua Portuguesa. Ademais, a nossa abordagem permite o uso flexível de outros ML baseados na arquitetura *Transformer*.

Por sua vez, ao considerarmos a perspectiva de análise **Anotação de corpus para ajuste fino de Modelos de Linguagem**, destacada no Quadro 4, é possível identificar que há convergências em sua maioria na aplicação das tarefas, nos modelos de linguagem e nos tipos de rotulagem. Nesse último, adotamos em nossa abordagem uma rotulagem híbrida por entendermos que a multicategoria é propícia para aplicação e interação com o ML ajustado, assim como a rotulagem de categoria única favorece a geração de KG. Em essência, há também convergência com a maioria dos trabalhos na adoção de métodos supervisionados à distância, porém estendemos a sua aplicação ao combinar com Recursos Semânticos (RS) (ontologias e taxonomias) e um método de anotação heurística. Em relação ao tipo de anotação, também adotamos o tipo semissupervisionado dado os seus benefícios de combinar a anotação manual e automatizada. Entretanto, diferentemente dos outros trabalhos foi incorporada uma métrica quantitativa para avaliar a qualidade da pré-anotação, medindo a eficácia e a contribuição do método, bem como otimizando o trabalho do curador.

No que tange à geração de DM, diferentemente dos trabalhos listados no Quadro 5, esta tese combina as abordagens DD e TD, alinhada às evidências demonstradas em Guizzardi, Pastor e Storey(15). Por um lado, no caso do DD, aproveitando as inferências estatísticas dos modelos subsimbólicos. Por outro lado, beneficiando-se da definição de um esquema simbólico característico do TD. Porém, como o uso de um esquema com categorias predefinidas restringe os resultados, optamos por adotar um metamodelo que permite utilizar categorias flexíveis e mais abrangentes, possibilitando a exploração em outros contextos.

Em relação ao domínio, diferentemente dos outros trabalhos relacionados, esta tese foi aplicada no domínio militar pelo caráter inovador e dada a capilaridade de assuntos que pode atuar. Por fim, utilizamos o modelo de linguagem BERTimbau em função de seus resultados em Língua Portuguesa superarem os outros ML multilínguais (36). Além disso, o BERTimbau permitiu o ajuste fino a partir de textos do domínio, alcançando resultados superiores aos experimentos realizados, como abordado no Capítulo 6. Contudo, vale ressaltar que a nossa abordagem proposta nesta tese não é restrita ao BERTimbau,

podendo instanciar outros ML.

Nesta seção, foram apresentados minuciosamente os trabalhos relacionados, bem como as análises comparativas com a abordagem proposta nesta tese de acordo com as perspectivas de análise e os critérios de comparação. No próximo capítulo, a abordagem proposta é discutida pormenorizada levando em consideração os desafios impostos e as hipóteses, como abordado no Capítulo 1.

4 ABORDAGEM IDEA-C2

Como abordado na seção 1.2, os desafios no cenário de Comando e Controle (C2) são amplos e exigem das Forças Armadas (FAs) o preparo para o emprego em operações militares frente às demandas impostas. Nesse cenário, um dos desafios envolve o elevado número de papéis exercidos pelas FA e variados tipos de operações de guerra e não guerra em que elas atuam. Por isso, a realização de treinamentos através de exercícios de operações singulares ou conjuntas são fundamentais para o aprimoramento das capacidades da Força, exigindo um ciclo de aperfeiçoamento mais ágil no mesmo ritmo que surgem os desafios inerentes aos cenários. Dessa forma, a extração de conhecimento a partir de materiais doutrinários é uma alternativa viável para apoiar os processos que envolvem o aperfeiçoamento de pessoal, além de contribuir com aplicações variadas.

Durante a pesquisa, foram realizadas experimentações a partir de amostras de um conjunto de textos oriundas de Doutrinas Militares (DML), as quais produziram algumas evidências sobre os problemas identificados. Com base nessas evidências, foi possível estabelecer requisitos para conceber a abordagem proposta, oferecendo uma solução viável aos problemas apontados na seção 1.2. Neste capítulo, é apresentada a especificação da abordagem, detalhando as características conceituais, os elementos arquiteturais e os processos de negócio, apresentados nas seções 4.1, 4.2 e 4.3, respectivamente.

4.1 Abordagem híbrida IDEA-C2

A abordagem IDEA-C2 (generat**I**on of knowle**D**ge graphs bas**E**d on **A**rtificial intelligence of **C2** Domain) é composta por um conjunto de componentes e artefatos combinando as características das abordagens TD e DD para conceber um DM, como ilustrado na Figura 12. Os componentes centrais **IDEA-C2-Language Model** e **IDEA-C2-Knowledge Graph** são artefatos intermediários gerados para apoiar o desenvolvimento do **IDEA-C2-Domain Model**.

No topo da Figura 12, os artefatos simbólicos incluem o **Metamodelo C2RM**, que incorpora as relações predefinidas **General** e **C2 domain relation**. Os artefatos simbólicos adicionais, tais como RS (e.g. glossários, taxonomias e ontologias), colaboram na identificação das instâncias do metamodelo e apoiam a anotação do **Corpus C2 Anotado e Curado**. Na parte inferior, encontram-se os artefatos sub-simbólicos ML pré-treinado e o **Corpus C2**, representado por C''. Esses artefatos são utilizados para construir IDEA-C2-Language Model (IDEA-C2-LM) ajustado ao domínio. Na realidade, o IDEA-C2-LM indiretamente aproveita o contexto semântico do corpus enriquecido com o Recurso Semântico (RS), contribuindo com a geração do **IDEA-C2 Knowledge Graph**

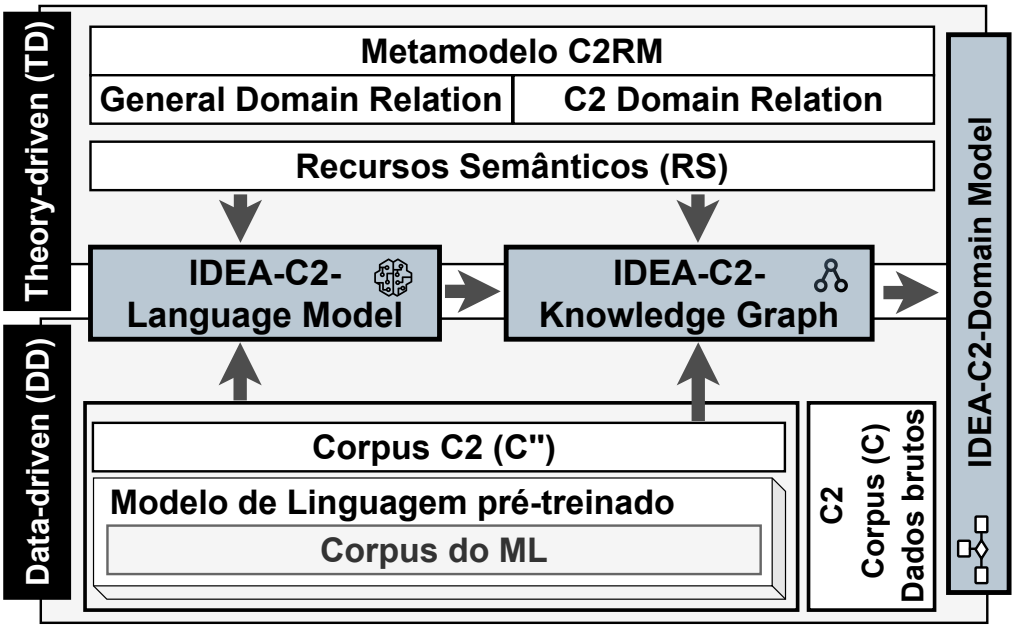


Figura 12 – Visão geral da abordagem híbrida IDEA-C2.

(IDEA-C2-KG). Finalmente, o **IDEA-C2 Domain Model** (IDEA-C2-DM) é derivado das interações com IDEA-C2-LM e IDEA-C2-KG. Embora IDEA-C2 tenha sido concebida para o domínio C2, a abordagem é flexível e pode ser adaptada a outros domínios a partir da submissão de outro corpus e de outros RS.

4.2 C2RM: Command and Control Relations Metamodel

Command and Control Relations Model (C2RM) é um metamodelo que tem como objetivo definir uma estrutura para representar as entidades reconhecidas e prover a semântica das relações entre as entidades de um determinado domínio, como ilustrado na Figura 14. Como um dos desafios é lidar com a flexibilidade de categorias, no metamodelo são definidos construtos de alto nível de abstração, **Entity** e **Relation**. Esses construtos são capazes de prover categorias abrangentes para extrair informações do corpus. No caso de *Relation*, esta foi especializada com relações de domínio geral, como exploradas nos trabalhos de Spala et al.(89) e Augenstein et al.(163) (como definição de termo, hiperônimo-hipônimo, parte-todo, sinônimo-equivalente). Além disso, *Relation* foi especializada com relações no contexto de C2, como por exemplo como *responsible_for*, as quais foram criadas com base no entendimento e observação dos padrões textuais utilizados nas DML, tendo como alvo os termos expressos no domínio, como veremos mais adiante.

A elaboração do metamodelo foi inspirada nos estudos de Muller(105), Fowler(97) e Brambilla, Cabot e Wimmer(106), utilizando a notação baseada no Modelo de Entidade-Relacionamento (MER) (98, 63). Como mencionado na seção 2.4, neste trabalho, assumimos

que metamodelos são abstrações que utilizam construtos de alto nível para descrever modelos conceituais em níveis de abstração mais baixos.

Os textos extraídos das DML não possuem estrutura padronizada, e ainda, o domínio de C2 é muito amplo e variado, podendo implicar na identificação de objetos muito diversos, o que dificultaria a pré-definição de categorias para anotá-lo. Nesse sentido, como a construção de metamodelos deriva de conceitos mais abstratos e de alto nível, nos cenários de aplicação em que se faz necessária a flexibilidade de estruturas de representação de dados, um metamodelo pode ser uma solução a ser aplicada (105).

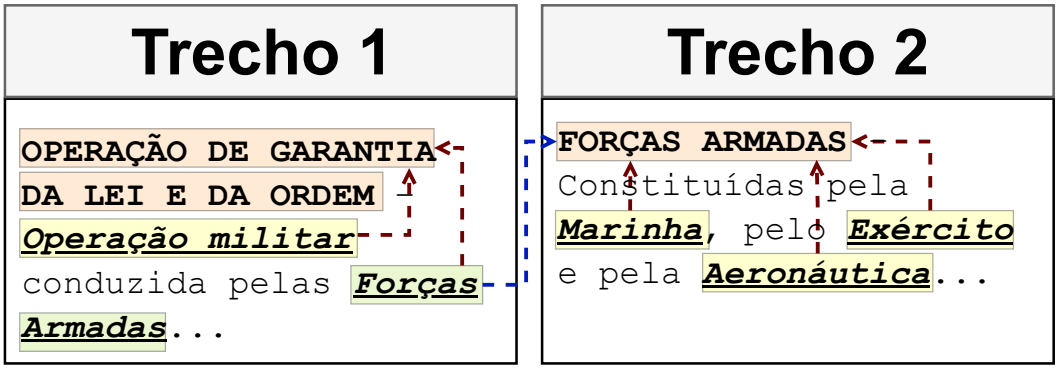


Figura 13 – Trechos extraídos do Glossário de Termos do EB (1). Imagem do autor.

Um exemplo de problema que favorece a aplicação do metamodelo C2RM pode ser visualizado a partir da ilustração da Figura 13, que se baseou em dois trechos extraídos do Glossário de Termos do EB (1). Nela, pode ser identificado que ambos os textos possuem um certo grau de estrutura (termo e definição). À esquerda, é expresso o "*termo definidor*", e, à direita, a sua "*definição*". Note que os textos nesses documentos expressam informações, até com relação à sua função e tipologia.

Esses trechos de textos são ricos porque discorrem sobre objetos do domínio de C2 e ainda explicitam relações entre eles. Por exemplo no trecho 1, há uma relação entre *operação militar* e *operação de garantia da lei e da ordem*, e outra entre *forças armadas* e *operação de garantia da lei e da ordem*. Assim como, no *trecho 2*, *marinha*, *exército* e *aeronáutica* relacionam-se com *forças armadas*. Todas essas relações são ilustradas por setas tracejadas na cor vermelha. Pode haver também relações implícitas entre os trechos. No exemplo, há uma relação entre os termos "forças armadas" dos *trechos 1* e *2*, ilustrada pela seta tracejada em azul.

Outro ponto que o exemplo da Figura 13 ilustra diz respeito à representação dos objetos, como mencionado na seção 2.3. Em estudos iniciais, observou-se que os objetos podem ser interpretados como descritores de algo. Como no caso de *forças armadas*, que pode ser modelada como uma abstração de entidade ou uma classe com suas instâncias *marinha*, *exército* e *aeronáutica*. Por outro lado, a expressão *forças armadas* pode ser

representada como uma abstração de hierarquia com as especializações *marinha*, *exército* e *aeronáutica*, ou seja, sem nenhuma instância. Assim, neste exemplo mostramos que embora as inferências possam identificar o termo "marinha" em diferentes níveis de abstração, ora como classe que especializa "forças armadas", ora como instância, o metamodelo C2RM pode representar ambas as ocorrências como **Entity**.

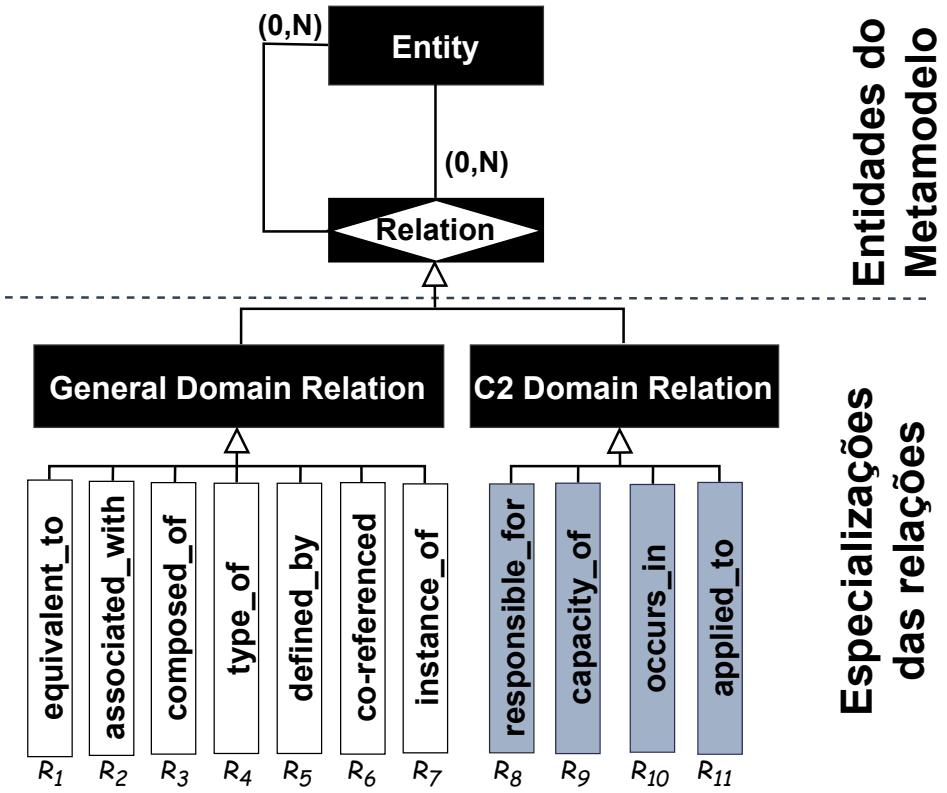


Figura 14 – Command and Control Relations Model (C2RM). Imagem do autor.

A seguir detalhamos melhor o metamodelo C2RM, ilustrado na Figura 14. Como foi mencionado, o C2RM apresenta as características de versatilidade e flexibilidade através dos construtos de alto nível **Entity** e **Relation**. O construto **Entity**, $E = \{e_1, e_2, ...e_n\}$, representa as entidades nomeadas reconhecidas no texto, por exemplo, Forças Armadas, Marinha, João, Operação GLO, etc. Observe que optou-se por não representar categorias fixas de entidades para aumentar a flexibilidade da abordagem. Da mesma forma, o construto **Relation**, $R = \{R_1, R_2, ...R_n\}$, refere-se à relação obtida através do autorrelacionamento do construto **Entity**, com cardinalidade M:N (muitos para muitos).

Note que é possível estabelecer relações quaisquer entre termo ou objeto no texto a qualquer outro termo, classificado somente como **Entity**, atribuindo a responsabilidade contextual e semântica para o construto **Relation**. Em outras palavras, a anotação de

entidades genéricas através de relações específicas permite que a definição da categoria das entidades seja realizada a posteriori, de acordo com um conjunto de possibilidades. Em estudos iniciais, observou-se que há trabalhos que utilizam várias categorias como estratégia para anotar os textos. Esse tipo de estratégia denomina-se **Multicategory** ou multicategorias. No presente trabalho, foi definida a estratégia denominada *Singlecategory* ou categoria única para definir entidades.

O construto **Relation** do C2RM possui duas especializações: **General Domain Relation** e **C2 Domain Relation**, que representam, respectivamente, as relações genéricas de fora do contexto de C2 e aquelas específicas do próprio contexto. Como essas especializações possuem características únicas, são definidas onze especializações que representam a semântica dos relacionamentos. As especializações, R_1, R_2 e R_7, foram inspiradas nas propriedades do RDF (115) que denotam equivalência, associação e instância. Já as especializações, R_3 e R_4, foram inspiradas no trabalho de Augenstein et al.(163), denotando composições e hierarquias. Enquanto que R_5 e R_6 foram inspiradas no trabalho de Spala et al.(89), denotando definição de termo e *co-referência*. Por fim, de R_8 a R_{11}, são especializações que provêm do contexto de C2, denotando responsabilidade, capacidade, ocorrência e aplicação. Todas as especializações são detalhadas no Apêndice A.

Nesta seção, foram apresentados os aspectos de construção do C2RM, detalhando a estratégia *singlecategory* especificada como *Entity*, bem como as especializações das relações do metamodelo. Na próxima seção, é apresentado o macroprocesso da abordagem IDEA-C2.

4.3 Processo IDEA-C2

Nesta seção, é apresentado o macroprocesso da abordagem IDEA-C2, o qual é composto por quatro subprocessos especificados de acordo com a notação *Business Process Model and Notation (BPMN)*[1] (Figura 15). Basicamente, os dois subprocessos iniciais apoiam os usuários especialistas na preparação dos principais artefatos da abordagem. Os usuários especialistas possuem conhecimentos acerca do negócio para anotar corpus de textos, bem como experiências e práticas para aplicar técnicas de aprendizado de máquina. Por sua vez, os subprocessos finais permitem que especialistas do domínio interajam com os artefatos gerados para apoiar na obtenção de conhecimento. Esses usuários especialistas de domínio são profissionais com perfil técnico em tecnologia da informação e experiência na construção de modelos de domínio.

Resumidamente, o subprocesso **Anotar corpus** cobre desde a entrada dos textos, incluindo a anotação semiautomatizada do corpus e sua curadoria, detalhado na subseção 4.3.1. Em seguida, o subprocesso **Ajustar modelo de linguagem** obtém como

[1] https://www.omg.org/spec/BPMN/2.0/

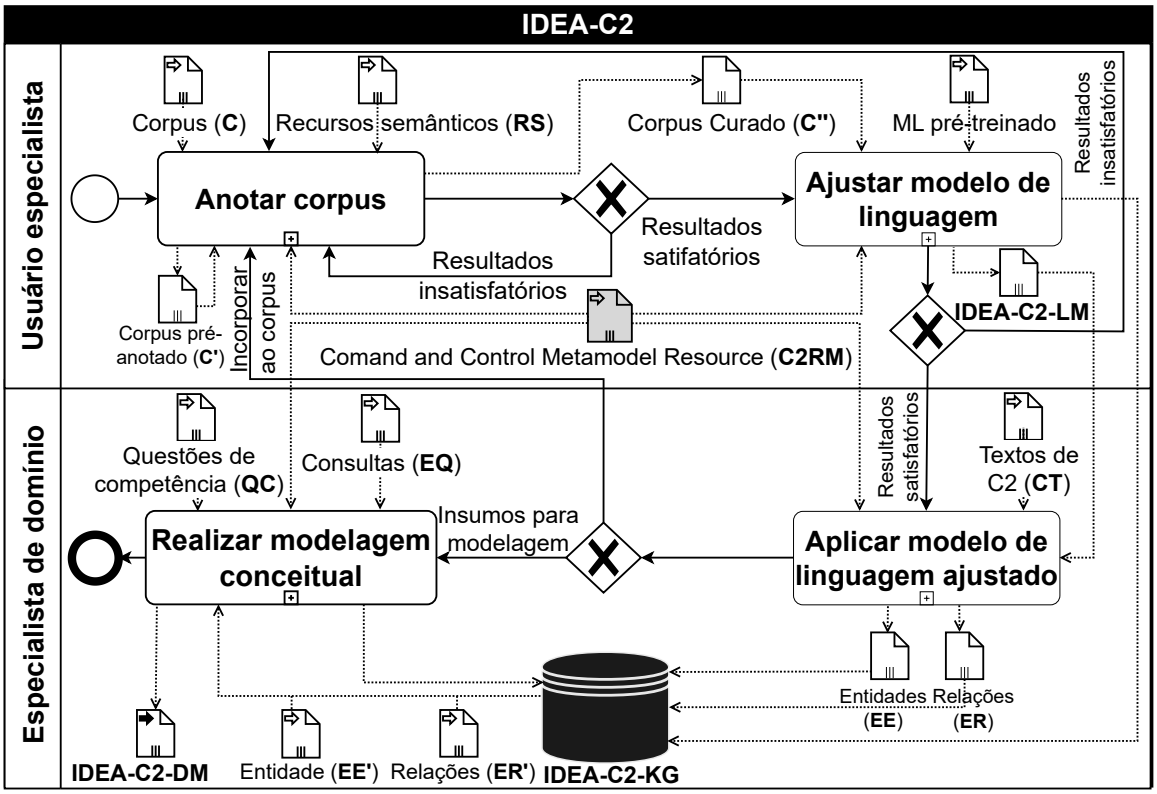


Figura 15 – Macroprocessos da abordagem IDEA-C2. Imagem do autor

entrada o corpus anotado e o ML pré-treinado para gerar um ML ajustado (IDEA-C2-LM) e um grafo (IDEA-C2-KG), detalhado na subseção 4.3.2. No subprocesso **Aplicar modelo de linguagem ajustado**, são submetidos textos ao IDEA-C2-LM para identificar entidades nomeadas e extrair relações, detalhado na subseção 4.3.3. Por fim, o subprocesso **Realizar modelagem conceitual** apoia os usuários na elaboração de um Modelo de Domínio (DM) a partir da exploração dos artefatos IDEA-C2-LM e IDEA-C2-KG, detalhado na subseção 4.3.4.

4.3.1 Anotação do corpus

A abordagem IDEA-C2 é iniciada através do subprocesso **Anotar corpus** que é estruturado pelos processos de **Preparar anotação**, **Definir regras de pré-anotação**, **Realizar pré-anotação**, além de **Realizar curadoria da anotação**. O processo **Preparar anotação** obtém o C2RM, os Recursos Semânticos (RS), representado por RS, disponíveis e alguns trechos de textos oriundos das doutrinas do domínio de C2, representadas por U. No entanto, apenas $C = \{s_1, s_2 \ldots, s_n\}$, que é um subconjunto de U ($C \subsetneq U$), é a entrada inicial deste processo e constitui o corpus. Considera-se que o RS pode ser um glossário ou vocabulário com termos e suas definições, o qual pode ser selecionado aleatoriamente pelo usuário especialista como fonte de dados que represente formalmente o domínio. Nesse caso, o uso do RS foi inspirado, por um lado, na proposta do Chaudhri

et al.(27) para construir um *knowledge base* a partir de uma representação formal do conhecimento. E, por outro lado, alinha-se à proposta do trabalho de Mintz et al.(31) no uso do RS como uma base de conhecimento externa para apoiar a anotação de um corpus. Assim, assume-se que os termos de RS constituem um conjunto de entidades a serem anotadas em C. Além disso, o conteúdo de RS é aproveitado para compor o corpus C. Como exemplo, foi extraído o fragmento s_1:"**Operação de garantia da lei e da ordem: Operação militar conduzida pelas Forças Armadas, por decisão do Presidente da República.**" que representa um trecho de texto extraído de C. Esse fragmento será explorado nos demais processos e subprocessos da abordagem.

Em seguida, o processo **Definir regras de pré-anotação** obtém o metamodelo conceitual C2RM, constituído de construtos de alto nível de abstração, *Entity* (E), *Relation* (R) e suas especializações ($R_1, R_2, ...R_n$), como metacategorias para anotar C. Com base em RS, são desenvolvidas regras de expressão regular para explorar automaticamente padrões textuais em C com o objetivo de identificar instâncias de relações. Tais padrões podem ser criados pelo usuário especialista a partir da observação da frequência com que esses padrões são evidenciados nos textos utilizados como fontes de dados. Para cada expressão regular t_p, é criada uma regra de mapeamento, formalmente $MR_p = (R_k, t_p)$, em que R_k é uma especialização de relação C2RM que deve ser associada à ocorrência de t_p, apresentado no Quadro 6. Note que é comum em expressões regulares o uso de máscaras, como por exemplo o asterisco "*", que representa o uso de um termo curinga. Por exemplo, na expressão "por decisão d*", o trecho "d*" corresponde a variações como: "da", "de" e "do". Por exemplo, ao analisar s_1, encontram-se as expressões "conduzida" e "por decisão do", que darão origem às expressões regulares utilizadas como regras para anotar as relações, respectivamente, "*applied_to*", R_{11}, e "*responsible_for*", R_8.

Cada entidade (e_n) identificada em C durante este subprocesso é transformada em uma instância de C2RM, ou seja, $e_n \in E$. Os termos de RS, "OPERAÇÃO DE GARANTIA DA LEI E DA ORDEM", e_1, "OPERAÇÃO MILITAR", e_2, "FORÇAS ARMADAS", e_3, e "PRESIDENTE DA REPÚBLICA", e_4, são exemplos de e_n identificadas em s_1. As relações são instanciadas, utilizando uma abordagem supervisionada à distância (31) de acordo com as regras definidas, detalhada no processo **Realizar pré-anotação**.

No processo **Realizar pré-anotação**, é gerado o corpus pré-anotado C' com as entidades, e_n e as relações R_k, 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_b, l_f) de e_j, e_m em s_i, onde l_b corresponde a primeira posição após e_j e l_f a posição anterior a e_m. Ao encontrar o termo da expressão regular no texto entre o par de entidades nas posições, l_b e l_f, é pré-anotado o texto de s_i, através da tripla (e_j, R_k, e_m) em C'.

Note que na Figura 16, é destacado um exemplo de pré-anotação com base em uma

Quadro 6 – Algumas regras de pré-anotação aplicadas ao Corpus C.

Mapeamento de Regras (MR_p)			
R_k	**Especializações do C2RM**	**Expressões regulares**	t_p
R_1	equivalent_to	"*mesmo que*"	t_1
R_2	associated_with	"*ver*"	t_2
R_3	composed_of	"*compost* p*" "*conjunt* d*"	t_3 t_4
R_4	type_of	"*tipo de **" "*subunidade de**"	t_5 t_6
R_8	responsible_for	"*executado p**" "*responsável p**" "*designad**" "*por decisão d**"	t_7 t_8 t_9 t_{10}
R_9	capacity_of	"*capacidade d* *"	t_{11}
R_{10}	occurs_in	"*ocorre em *"	t_{12}
R_{11}	applied_to	"*emprega* *" "*aplica**" "*conduzid**"	t_{13} t_{14} t_{15}

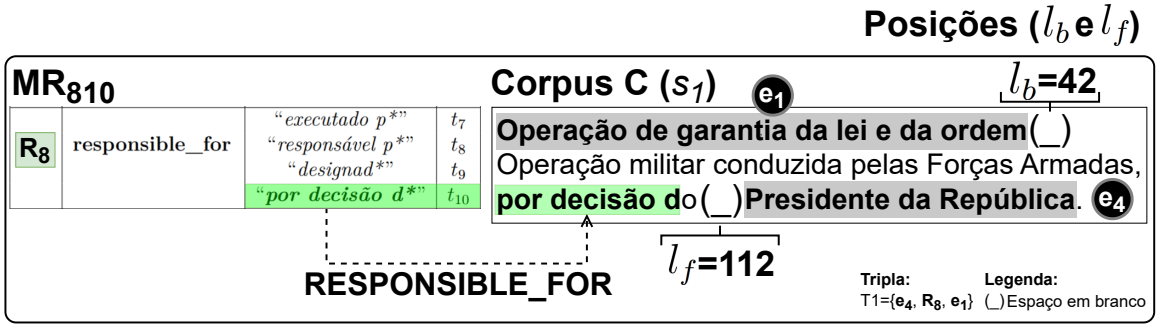


Figura 16 – Pré-anotação baseada na regra M_{810} (e_4,R_8,e_1). Imagem do autor.

pesquisa exaustiva em s_1 no corpus C da regra de pré-anotação MR_{810}. Ao encontrar a ocorrência de t_{10} entre um par de entidades (e_1, e_4), então MR_{810} é aplicada e a instância de relação (e_4, R_8, e_1) é criada, assim como o texto é pré-anotado. Como resultado da aplicação de todas as regras de mapeamento predefinidas, C dá origem a um novo corpus pré-anotado denominado C'. Cabe ressaltar que apesar do exemplo ter somente explorado a pré-anotação com o mapeamento da regra MR_{810}, os demais mapeamentos possuem o mesmo comportamento. Outro ponto a destacar é que a pré-anotação não prevê uma hierarquia de prioridade semântica entre as relações. Assim, caso haja conflito de regras, uma nova tripla com a relação MR_p correspondente é incluída. Por exemplo, vamos supor que o usuário especialista também tenha associado a expressão regular "por decisão d*" à especialização *capacity_of*, R_9, assim, criando a MR_{910}. Repare que a expressão regular "por decisão d*" possui dois valores semânticos distintos associados às regras MR_{810} e

MR_{910}. Para tal, o par de entidades (e_1, e_4) também é relacionado por R_9 através de (e_1, R_9, e_4) e o texto é pré-anotado em C' com ambas as regras. Nesse caso, cabe ao usuário especialista analisar os textos anotados através do processo **Realizar curadoria da anotação** e decidir a ação a ser realizada.

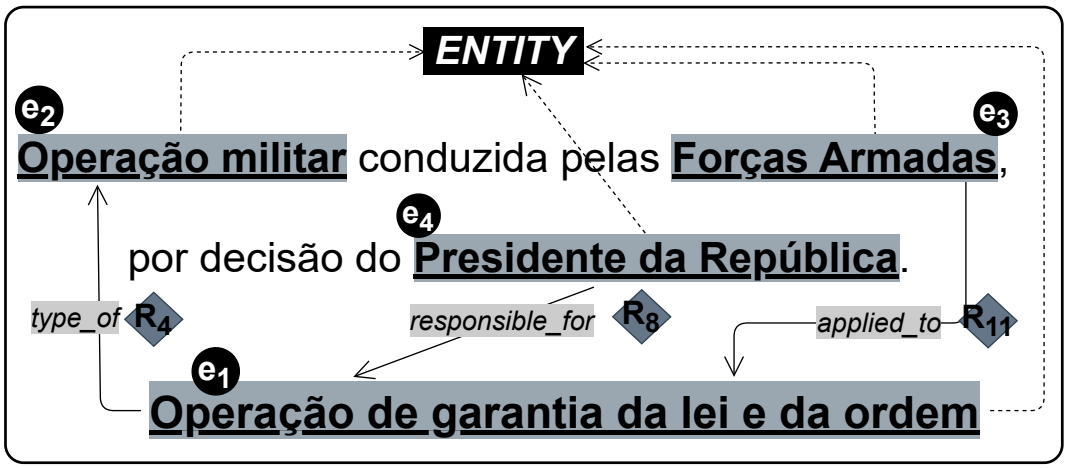


Figura 17 – Exemplo de pré-anotação do texto s_1 do contexto de C2 suportados por C2RM. Imagem do autor.

Ao término do processo **Realizar pré-anotação**, na Figura 17, é ilustrado o fragmento de s_1 pré-anotado. Note que foram identificadas quatro entidades, representadas por $E = \{e_1, ..., e_4\}$, e três triplas de entidades e relações, representadas por $T_q = (e_j, R_k, e_m)$. Além disso, são gerados aleatoriamente os subcorpora SC' e SM. Os subcorpora correspondem a 10% do total de sentenças de C e possuem as mesmas sentenças, exceto as anotações. Esses subcorpora foram criados com o objetivo de validar a pré-anotação por amostragem, minimizando a necessidade de validar todo o corpus. O subcorpus SC' é composto por textos pré-anotados $|(SC' \subset C')|$. Já o subcorpus SM é composto por textos sem anotação $|(SM \subset C)|$. Ambos os subcorpora são utilizados no processo **Realizar curadoria da anotação**.

No processo **Realizar curadoria da anotação**, são avaliadas por amostragem as pré-anotações de C', a partir de SC' e SM, com o objetivo de gerar o corpus curado C''. Para tal, o especialista anota SM manualmente, apoiado por ferramenta de anotação, e gera SM'. Em seguida, o especialista submete ambos os subcopora SC' e SM' para comparação automatizada. Por exemplo, ao compararmos o termo do fragmento de s_1, "OPERAÇÃO MILITAR", note que ele pode ser anotado em ambos os subcorpora como *entity*, representando um *True Positive*, $TP = |(SC' \cap SM')|$. Contudo, caso haja divergências e o usuário anote manualmente em SM', por exemplo, "DECISÃO" como *entity*, a abordagem identifica um *False Negative*, $FN = |(SM' - SC')|$, pois esse termo anotado não existe em SC'. O contrário pode ocorrer, representando um *False Positive*, $FP = |(SC' - SM')|$. Os valores de TP, FP e FN são utilizados para calcular as métricas

precisão e *recall* com o objetivo de avaliar a pré-anotação. Similarmente, o mesmo tipo de 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").

Ao término, o usuário especialista avalia os resultados da pré-anotação com base nas métricas de precisão e *recall* das entidades e relações. A precisão indica ao especialista o quão a pré-anotação é confiável, pois mede a proporção de TP na amostragem. Já a métrica *recall* indica a cobertura do resultado na amostragem, pois mede realmente o que é positivo através da proporção de TP com FN. Caso os valores das métricas sejam insatisfatórios, o especialista deve retornar ao subprocesso **Definir regras de pré-anotação** e realizar dois tipos de análises distintas. Na primeira, ao considerar somente as entidades, o especialista deve avaliar o RS instanciado. Na segunda, considerando as relações, o especialista deve revisar as regras de pré-anotação, incluindo as divergências apontadas entre FN e FP. Como a anotação manual de SM' é utilizada como referência (*gold standard*), ao avaliar o valor de FN, o especialista deve corrigir a regra que originou a pré-anotação em SC'.

4.3.2 Ajuste do modelo de linguagem

O subprocesso **Ajustar modelo de linguagem** tem como objetivo realizar o ajuste fino do ML pré-treinado de acordo com as categorias de entidades e relações no corpus anotado, C'', e depois gerar o IDEA-C2-LM. Para tal, obtém-se o corpus C'', e instanciam-se o ML pré-treinado e o C2RM. Cada sentença, s_i, de C'', é pré-processada, padronizada para minúscula e as *stopwords* são removidas. Além disso, o corpus C'' é ordenado aleatoriamente e dividido em 80% para treino e validação e 20% para teste. Cabe ressaltar que os parâmetros de divisão são configuráveis.

Para o ajuste fino do ML, o modelo pré-treinado é inicializado com seus pesos originais e adiciona-se uma camada específica para cada tarefa (e.g. uma camada de classificação para NER e outra para RE), ajustando a arquitetura ao formato dos rótulos do corpus. Em seguida, os textos de C'' são tokenizados com o mesmo *tokenizer* do modelo pré-treinado, alinhando-se os rótulos aos sub-tokens quando necessário, e os dados são organizados em lotes ou *batches* para treinamento supervisionado. Durante o processo, calcula-se uma função de perda apropriada (como *cross-entropy*), que é retropropagada (*backpropagation*) para atualizar os pesos do modelo por meio de um otimizador (e.g., AdamW), com controle de hiperparâmetros como *learning rate*, número de épocas, *batch size* e *dropout*. De acordo com as épocas, avalia-se o desempenho do ML ajustado em um conjunto de validação, utilizando as métricas de precisão, *recall* e F1-score, possibilitando ajustes e prevenção de *overfitting*. Isso ocorre até que o modelo ajustado consiga capturar

os padrões do domínio representado no corpus C''.

Ademais, o IDEA-C2-KG é gerado com base no corpus utilizado no ajuste do IDEA-C2-LM. Entretanto, é necessário incorporar semântica das relações de C2RM ao IDEA-C2-KG. Para tal, o metamodelo deve ser convertido no formato de grafo RDF, como ilustrado na Figura 18.

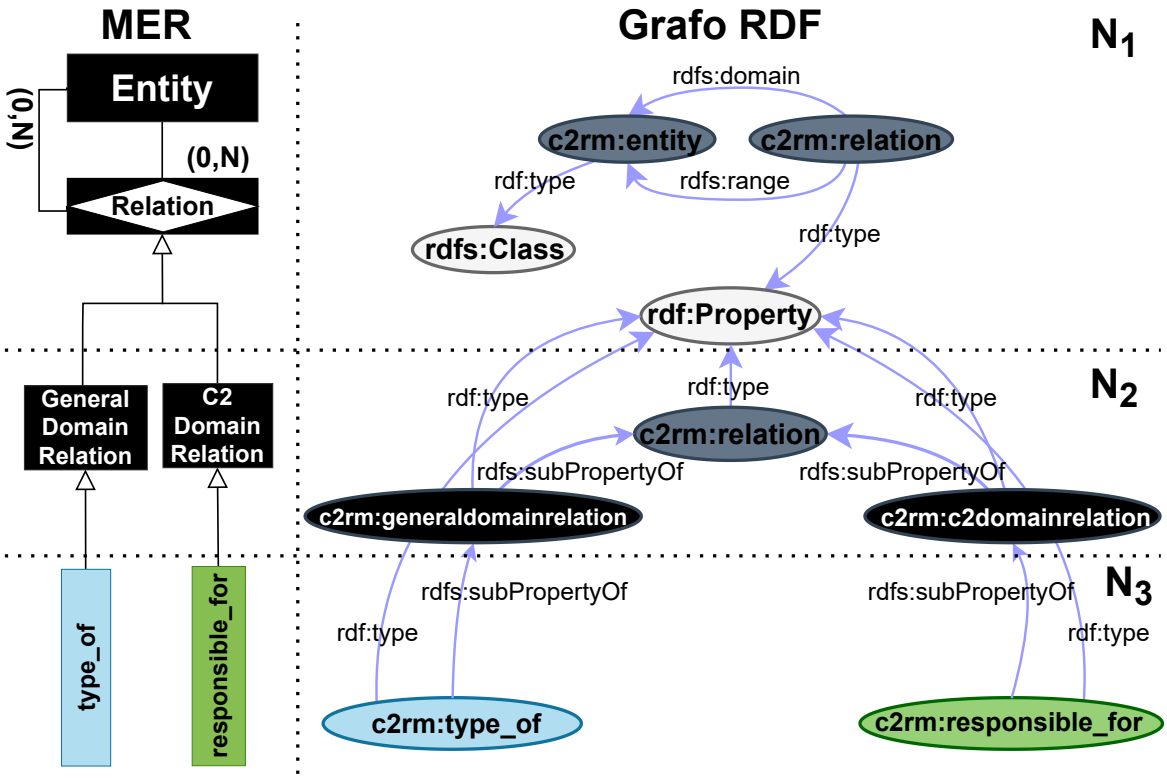


Figura 18 – Representação de C2RM em Grafo RDF. Imagem do autor.

A priori, são definidos os *namespaces* ***c2rm***, responsável por identificar os recursos do metamodelo no grafo, e ***cnt*** destinado aos demais recursos. Na Figura 18, a conversão dos construtos de **C2RM** é organizada nos *frames* N_1, N_2 e N_3. Além disso, à esquerda, a representação denominada **MER** corresponde ao modelo conceitual contendo os construtos do C2RM. À direita, o **Grafo RDF** tem por objetivo representar esses construtos convertidos na forma de recursos RDF, explicitando seus tipos, domínios, faixas de alcance e hierarquias.

No *frame* N_1 da Figura 18, são apresentados os construtos de mais alto nível de C2RM com destaque ao autorrelacionamento de **Entity** e **Relation**, que são instâncias, respectivamente, das metaclasses *rdfs:Class* e *rdf:Property*. Assim, *Entity* e *Relation* são construtos que descrevem os objetos anotados nos textos doutrinários, materializados no grafo através dos recursos *c2rm:entity* e *c2rm:relation*. Além disso, o autorrelacionamento é um artifício de modelagem que agrega aos pares de entidade as suas relações, representados por e_1, R_n, e_2. Note que *c2rm:entity* e *c2rm:relation* são relacionados pelas propriedades

rdfs:domain e *rdfs:range*. Logo, o recurso *c2rm:relation* é uma propriedade cujas instâncias possuem como domínio e "range" instâncias de "entity".

Por sua vez, o *frame* N_2 da Figura 18, corresponde aos construtos das especializações de *Relation*. Note que as especializações *General Domain Relation* e *C2 Domain Relation* são representadas pelos recursos *c2rm:generaldomainrelation*, que agrega as relações genéricas fora do contexto, e *c2rm:c2domainrelation*, responsável pela relações do contexto de C2. Como ambas as especializações são subpropriedades de *c2rm:relation*, elas herdam suas características. Logo, ambas as propriedades são restritas ao domínio de *c2rm:entity*.

No *frame* N_3 da Figura 18, as especializações de *General Domain Relation* correspondem às relações de R_1 a R_7 de C2RM. O *C2 Domain Relation* corresponde as relações de R_8 a R_{11}. Por simplificação, somente foram demonstradas as especializações *c2rm:type_of*, destacado em azul, e *c2rm:responsible_for*, destacado em verde. Como essas especializações são subpropriedades de *c2rm:generaldomainrelation* e *c2rm:c2domainrelation*, elas herdam todas as características de *c2rm:relation*.

Cabe ressaltar que cada especialização possui a sua semântica de representação. Por exemplo, como a especialização *c2rm:type_of* descreve a hierarquia de classes, ela é utilizada para representar a relação entre os termos "Operação militar" e "Operação de garantia da lei e da ordem" expressa em s_1. Ambos os termos são representados através dos recursos do grafo *cnt:operacao_militar* e *cnt:operacao_de_garantia_da_e_da_ordem*, ambos do tipo *c2rm:entity*, os quais são relacionados pela propriedade *c2rm:type_of*. De modo análogo, a relação entre de "Presidente da República" e "Operação de garantia da lei e da ordem" é representada por meio dos recursos *cnt:presidente_da_republica* e *cnt:operacao_de_garantia_da_e_da_ordem* através de *c2rm:responsible_for*.

Com a construção do IDEA-C2-KG concretizada, são avaliadas as métricas de precisão e de *recall* do IDEA-C2-LM, bem como as inferências de entidades nomeadas e relações extraídas. Caso os resultados das métricas ou das inferências não sejam satisfatórios, o usuário especialista retorna para o subprocesso **Anotar corpus** a fim de revisar ou até mesmo refazer a pré-anotação e demais atividades. Caso contrário, o subprocesso **Aplicar modelo de linguagem ajustado** é ativado.

4.3.3 Aplicação do modelo de linguagem ajustado

No subprocesso **Aplicar modelo de linguagem ajustado**, o objetivo é apoiar os especialistas do domínio na extração de entidades e relações a partir de sentenças de texto do contexto de C2, representado por $CT \subset (U - C)$, em que $CT = \{st_1, st_2, ...st_m\}$. Os usuários submetem CT ao IDEA-C2-LM para reconhecer um conjunto de entidades nomeadas e relações entre elas. Para exemplificar, vamos supor que o fragmento de s_1 fosse submetido ao IDEA-C2-LM, obtêm-se como resultados os conjuntos de dados de

Entidades nomeadas reconhecidas, EE, e as Relações entre essas entidades, ER. Ambos os resultados são apresentados, respectivamente, nos Quadros 7 e 8. Cabe destacar que a partir das instâncias dos conjuntos, EE e ER, oportunamente o usuário especialista pode executar a pré-anotação, utilizando como rótulos tais conjuntos sobre o conteúdo de CT. Com isso, o usuário especialista pode incorporar o CT pré-anotado ao corpus C'' para retreinamento do IDEA-C2-LM.

Quadro 7 – Entidades nomeadas reconhecidas na submissão de s_1 ao IDEA-C2-LM.

Entidades nomeadas reconhecidas (EE)	
(e_n)	**Entidades**
e_1	*Operação de garantia da lei e da ordem*
e_2	*Operação militar*
e_3	*Forças Armadas*
e_4	*Presidente da república*

Quadro 8 – Triplas de entidades e relações extraídas da submissão de s_1 ao IDEA-C2-LM.

Triplas de Entidades e Relações (ER)			**(e_i, R_j, e_k)**
Operação de Garantia da Lei e da Ordem	type_of	*Operação Militar*	(e_1, R_4, e_2)
Forças Armadas	applied_to	*Operação Militar*	(e_3, R_{11}, e_1)
Presidente da República	responsible_for	*Operação de Garantia da Lei e da Ordem*	(e_4, R_8, e_1)

Com base nos resultados, o IDEA-C2-KG pode ser enriquecido com novos recursos e relações que provêm da interação com o IDEA-C2-LM. Formalmente, o IDEA-C2-KG é formado por pares de (EE, ER), em que $EE = \{e_1, e_2, ...e_n\}$ é o conjunto de entidades que corresponde aos nós do grafo, assim como as relações são representadas como um conjunto de arestas sob a forma de triplas: $ER = \{(e_i, R_j, e_k) \mid e_i, e_k \in EE$ estão semanticamente relacionados por $R_j \subset R$, em algum $st_l \in CT\}$.

Na Figura 19, o conjunto EE é instância de **rdfs:Class**, denominado como **c2rm:entity**. Cada entidade EE é formada pela tripla T_i=$(e_i, \textbf{rdf:type}, \textbf{c2rm:entity})$, como exemplo os recursos *cnt:presidente_da_republica* e *cnt:operacao_de_garantia_da_lei_e_da_ordem*. Além disso, para cada tripla (e_i, R_j, e_k) do conjunto ER, R_j é mapeada através de uma propriedade equivalente expressa no formato RDF, como apresentado no Quadro 9. Por exemplo, em destaque a tripla $T_3 = (e_4, R_8, e_1)$ que representa os recursos: *cnt:presidente_da_republica*, *c2rm:responsible_for*, *cnt:operacao_de_garantia_da_lei_e_da_ordem*.

Para formalizar o mapeamento, note que são consideradas as especializações de C2RM, representadas por **c2rm:**R_j. Quando $j \leq 7$, são mapeadas as propriedades semânticas semelhantes e independentes do domínio, como exemplo a especialização *equivalent_to*, que foi mapeada como *owl:equivalentClass* para representar os recursos de dados equiva-

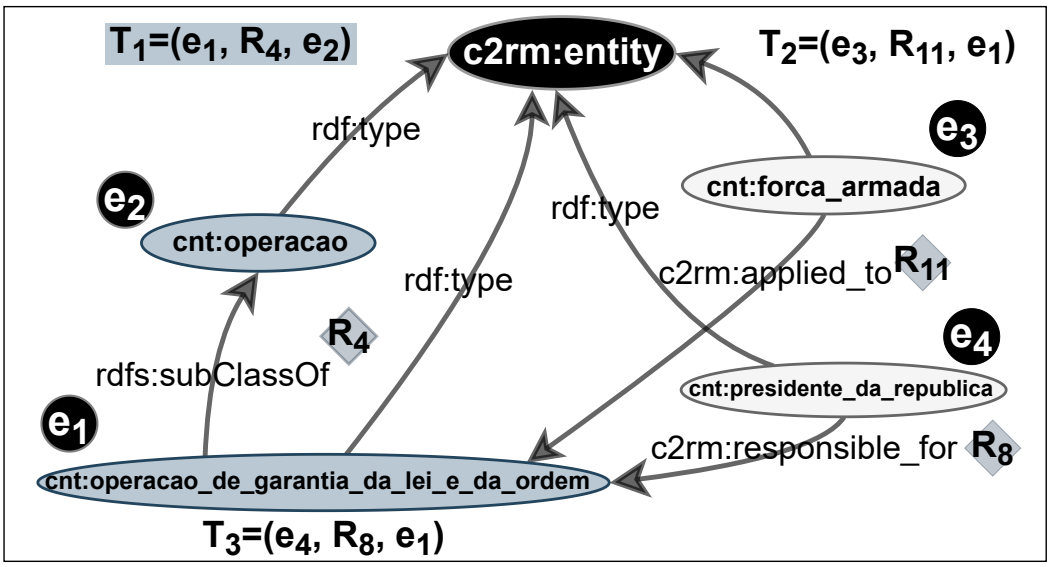


Figura 19 – Fragmento do IDEA-C2-KG que corresponde ao texto ilustrado na Figura 16. Imagem do autor.

Quadro 9 – Mapeamento entre as especializações do C2RM e das propriedades do grafo RDF.

R_j	**Especializações do C2RM**	**Propriedades do Grafo RDF**
R_1	equivalent_to	owl:equivalentClass
R_2	associated_with	rdfs:seeAlso
R_3	composed_of	rdf:Bag
R_4	type_of	rdfs:subClassOf
R_5	defined_by	rdfs:comment
R_6	co-referenced	c2rm:coreferenced
R_7	instance_of	rdf:type
R_8	responsible_for	c2rm:responsible_for
R_9	capacity_of	c2rm:capacity_of
R_{10}	occurs_in	c2rm:occurs_in
R_{11}	applied_to	c2rm:applied_to

lentes. Porém, quando $j \geq 8$, assumem-se as propriedades intrínsecas do domínio, como exemplo a especialização *responsible_for* que foi mapeada como *c2rm:responsible_for*.

4.3.4 Modelagem conceitual de dados

No subprocesso **Realizar Modelagem Conceitual**, os especialistas no domínio têm como objetivo desenvolver um DM ou modelo do domínio, combinando as abordagens DD e TD. A abordagem TD envolve a definição de Questões de Competência (QC), representado por QC, na fase de elicitação de requisitos a fim de nortear a construção do DM na modelagem conceitual (164).

No IDEA-C2, o metamodelo C2RM, através de suas especializações, estabeleceu algumas QC que são exigidas pelas aplicações no domínio C2, tais como: "quem é responsável por algo?" ou "o que é aplicado a quê", como abordado na seção 4.2. Por outro lado, seguindo as especificidades da abordagem DD, as QC podem ser enriquecidas explorando o IDEA-C2-KG através de consultas SPARQL Protocol and RDF Query Language (SPARQL) ou analisando graficamente os nós e arestas e observando as entidades nele representadas. No exemplo da Figura 19, surgem outras QC mais específicas, tais como qc_1: "A que entidades se aplicam as forças armadas?" ou qc_2: "Quem é responsável pelas operações de garantia da lei e da ordem?". Essas perguntas sugerem que "operações de garantia da lei e da ordem" e "forças armadas" são classes com instâncias possíveis, que devem ser representadas no modelo de domínio, como ilustrado na Figura 21. Portanto, cada R_j do C2RM pode ser analisada por meio de seus predicados correspondentes no KG e, então, auxiliar o especialista do domínio a identificar QC e classes e, consequentemente, projetar o modelo de domínio.

Observe que ao explorarmos graficamente o exemplo do IDEA-C2-KG (Figura 19), identifica-se, também, que nele há relações de subclasse (**rdfs:subClassOf**). Essas relações sugerem hierarquias que podem ser criadas no modelo de domínio. Cada hierarquia deve ser analisada para verificar se pode acomodar mais de um nível hierárquico. Uma situação comum na modelagem é identificar algumas subclasses de uma única superclasse S qualquer no KG, as quais estão relacionadas à outra classe A por meio do mesmo predicado P. Essa situação pode sugerir que existe uma relação P com a classe S que é herdada por suas subclasses.

No exemplo, classes como *cnt:operacao_de_garantia_da_lei_e_da_ordem* e *cnt:operacao* parecem fazer parte de uma hierarquia. Note que a relação **c2rm:applied_to** ou "aplicada a" está diretamente conectada a *cnt:operacao_de_garantia_da_lei_e_da_ordem*, mas provavelmente se conecta a muitos outros subtipos de operação, como *cnt:operacao_ofensiva*, *cnt:operacao_defensiva* e assim por diante. Assim, pode ser necessário criar um nível intermediário na hierarquia para representar a classe "operação militar", na qual a relação **c2rm:applied_to** seria então conectada no modelo de domínio e, consequentemente, herdada pelas subclasses correspondentes.

Além disso, as relações "associadas a" ou "associated_with" (**rdfs:seeAlso**), encontradas no IDEA-C2-KG também podem sugerir ao especialista do domínio novas relações no DM. Por exemplo, ao analisar a tripla (**cnt:forcas_armadas, rdfs:seeAlso, cnt:operacao_defensiva**) repare que poderia ser mapeada para uma relação semântica no modelo de domínio, representando que "forças armadas" são aplicadas, por exemplo, em "operação defensiva". Nesse caso específico, poderia explorar a existência ou não de nós no KG relacionados, direta ou indiretamente, à "operação defensiva" que indiquem "ameaça" ou "agressão". Supondo que exista, os nós no KG podem estar relacionados outros nós que

indiquem, por exemplo, tipos de manobras táticas defensivas a ser empregadas, ou melhor "aplicadas a", como **defesa de área** ou **defesa móvel**, quando o objetivo for uma **defesa em posição**, ou de **ação retardadora**, **retreinamento** ou **retirada**, quando envolve o **movimento retrógrado** (165).

Ainda no exemplo (Figura 19), o nó rotulado como "presidente da república" (*cnt:presidente_da_republica*) representa uma função que pode ser desempenhada por uma pessoa. Contudo, note que não há esse nó pessoa no grafo. Ao desempenhar essa função, um presidente pode também ser responsável por diferentes tipos de operações de não guerra, como exemplo as operações de garantia dos poderes constitucionais (1). Nesse contexto, outras funções diferentes podem surgir e estar ligadas a diferentes tipos de operações. Dessa forma, supondo que as funções sejam contingentes, seria necessário representar as pessoas por trás delas. Assim, ao identificar classes que na realidade são funções, pode ajudar o especialista do domínio a projetar hierarquias e esclarecer relações no DM.

Figura 20 – Uma abordagem exploratória para apoiar a modelagem conceitual. Imagem do autor.

No IDEA-C2-KG (Figura 19), ao investigar o predicado **rdf:type**, nota-se que ele corresponde à especialização *instance_of* em C2RM (Quadro 9). Na abordagem DD, o reconhecimento dessa relação pode levar a imprecisões. Por exemplo, na tripla **(cnt:operacao_defensiva, rdf:type, cnt:operacao)**, o objeto da tripla não é uma

instância, mas sim uma subclasse do próprio sujeito. Por outro lado, alguns R_j podem ser perdidos pela abordagem DD, i.e., o KG pode conter instâncias de **c2rm:entity** que não estão conectadas a nenhum outro nó. Essas instâncias podem ser classes relevantes, devendo, portanto, ser revisadas pelo especialista no modelo de domínio.

Até então, todas as ações apresentadas são realizadas pelo especialista do domínio enquanto concebe o DM com o apoio do IDEA-C2-KG. A Figura 20 resume o conjunto de atividades do subprocesso de **Realizar Modelagem Conceitual**. Diferentemente dos outros subprocessos, a elaboração de um modelo de domínio, através da exploração de um KG, deve adotar como premissa a flexibilidade, permitindo que o especialista utilize os recursos do grafo com a maior liberdade possível.

Conforme explicado anteriormente, os usuários podem enriquecer o DM, explorando as relações entre os nós do IDEA-C2-KG. Em primeiro lugar, é possível consultar o KG por meio de expressões SPARQL. Em segundo lugar, a visualização dos resultados da consulta em um grafo também é útil para o especialista do domínio identificar relações explícitas e implícitas. Sendo assim, a elaboração desse subprocesso reúne três atividades (**Identificar objetos**, **Explorar recursos** e **Analisar relações**) e foi inspirado no processo de desenvolvimento de software Iterativo e Incremental (59) combinado com a visão estrutural da abordagem de construção de ontologias SABiOx (166).

Na atividade **Identificar objetos**, obtém-se as QC com o objetivo de identificar as classes, relações, superclasses e subclasses do domínio a partir da investigação no IDEA-C2-KG, interagindo com as demais atividades, como **Explorar recursos** e **Analisar resultados**. Por sua vez, na atividade **Explorar recursos**, o especialista do domínio investiga os recursos de dados do KG a partir de consultas básicas para determinar as classes e relações do domínio, utilizando as propriedades de C2RM tanto associadas ao domínio (e.g. *c2rm:responsible_for* ou *c2rm:applied_to*, dentre outras) quanto as genéricas (e.g. *rdfs:subClassOf*, *rdf:type*, dentre outras).

Por fim, a atividade **Analisar relações** reúne as consultas voltadas a explorar recursos através de relações hierárquicas, inclusive aquelas em que há relações implícitas. O Quadro 10 apresenta um conjunto de expressões de consultas exploratórias em SPARQL para apoiar as atividades do subprocesso de **Realizar modelagem conceitual**. Cabe ressaltar que essas consultas são exemplificativas e não cessam a exploração que pode ser realizada no KG, i.e., é possível que os usuários investiguem outros recursos no KG através de consultas distintas, as quais sejam mais pertinentes à elaboração do modelo de domínio desejado.

Na Figura 21, é ilustrado o modelo de domínio (IDEA-C2-DM) elaborado, além de destacar o mapeamento inicial (1x1) entre alguns recursos do IDEA-C2-KG e os fragmentos do modelo de domínio. Observe que os nós, e_4 e e_1, são mapeados através das classes c_4 e c_1. Da mesma forma, a relação, R_8, é mapeada por meio da relação, LR_8. Além disso, note

Quadro 10 – Expressões de consultas exploratórias para apoiar a elaboração do modelo de domínio.

	Consultas SPARQL	**Atividades de modelagem conceitual**
EQ1	SELECT ?s ?o WHERE {?s c2rm:applied_to ?o . ?s rdf:type c2rm:entity}	Analisar instâncias de entidades reconhecidas que estão conectadas por meio do predicado **c2rm:applied_to** para identificar classes do modelo de domínio.
EQ2	SELECT ?sup ?sub ?p WHERE {?sub rdfs:subClassOf ?sup . ?sub ?p []}	Analisar o predicado **rdfs:subClassOf** para identificar hierarquias e suas relações de classe no modelo de domínio.
EQ3	SELECT ?s ?o WHERE{?s rdfs:seeAlso ?o}	Analisar o predicado **rdf:seeAlso** para identificar novas relações entre as classes do modelo de domínio.
EQ4	SELECT ?c ?i WHERE{?i rdf:type ?c FILTER(?c != c2rm:entity)}	Analisar o predicado **rdf:type** para identificar relações de instâncias imprecisas.
EQ5	CONSTRUCT {?s1 ?p1 ?o1. ?s2 ?p2 ?o2. ?s3 ?p3 ?o3. ?s4 ?p4 ?o4} WHERE{ {SELECT * WHERE {?s1 ?p1 ?o1. FILTER(?s1=*par1*)}} {SELECT * WHERE {?s2 ?p2 ?o2. FILTER(?o2=*par1*)}} {SELECT * WHERE {?s3 ?p3 ?o3. FILTER(?s3=*par2*)}} {SELECT * WHERE {?s4 ?p4 ?o4. FILTER(?o4=*par2*)}} FILTER(?o2=?s1 && ?o4=?s3 && ?s2=?s4 && ?o1=?o3)}	Analisar instâncias de nós conectados a pares (**par1**, **par2**) correspondentes aos nós (**s1**, **s2**, **s3**, **s4**) que possuem nós adjacentes e comuns entre si. Em outras palavras, descobrir nós adjacentes com relações implícitas.

que as relações semânticas do KG também contribuem para a representação gráfica usada no DM. Um exemplo dessa contribuição é a propriedade *rdfs:subClassOf*, que representa a relação de herança entre recursos, utilizada na hierarquia entre as classes **operacao** e **operacao_de_garantia_da_lei_e_da_ordem**.

Por outro lado, as propriedades do C2RM (e.g. **responsible_of** e **applied_to**), apesar de expressar a semântica na relação, elas não influenciam graficamente no tipo de relacionamento no DM, sendo representadas como uma associação simples. Outro ponto em destaque, é a criação da classe generalizada pessoa que não existe no KG. Isso foi possível em função do IDEA-C2 incorporar recursos da abordagem TD, permitindo que os usuários consigam inferir a classe generalizada ao observar outros recursos semelhantes (e.g. comandante de aeronave) por meio das interações com o IDEA-C2-KG.

Portanto, o IDEA-C2-DM é um modelo de domínio que resulta da união das abordagens DD e TD, combinando a análise do IDEA-C2-KG gerado pelo IDEA-C2-LM e as atividades de modelagem conceitual, permitindo enriquecer o conhecimento do especialista do domínio com um conjunto de informações extraídas independentemente

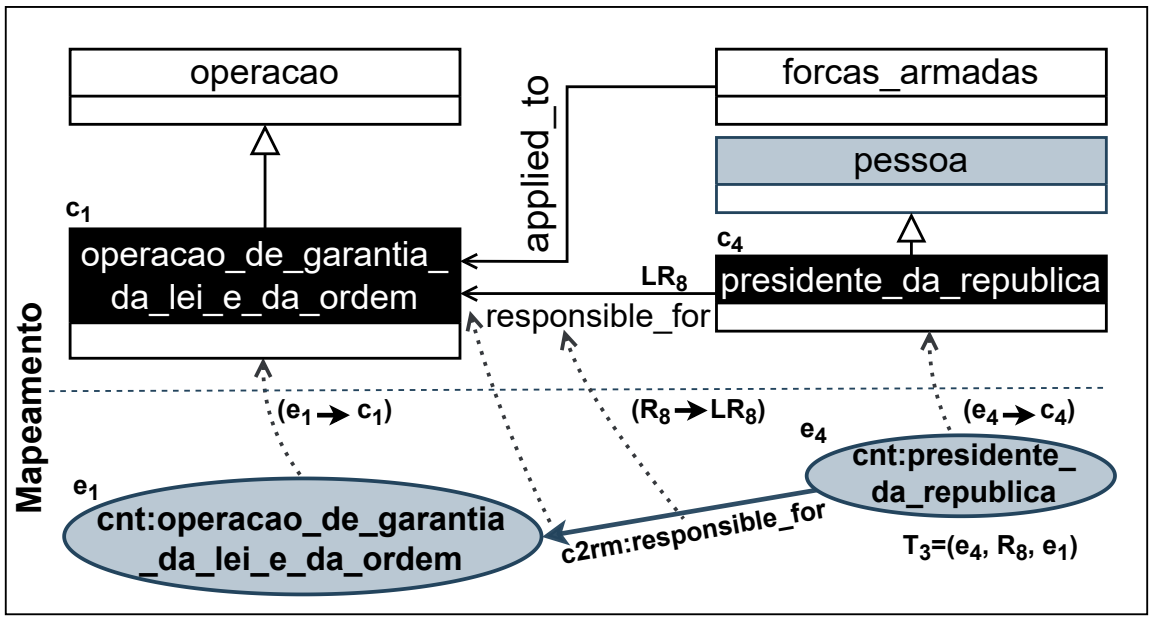


Figura 21 – Fragmento do modelo de domínio (IDEA-C2-DM) baseado no IDEA-C2-KG da Figura 19. Imagem do autor.

dos níveis de abstração que elas se encontram.

4.4 Considerações finais da abordagem IDEA-C2

Nesta seção, a abordagem supervisionada e híbrida IDEA-C2 foi apresentada, explorando as características arquiteturais, o metamodelo C2RM e o seu macroprocesso.

A abordagem IDEA-C2 foi concebida de forma incremental e evoluída ao longo do desenvolvimento da pesquisa. Inicialmente, os desafios associados ao tratamento de textos impuseram a necessidade de investigar abordagens capazes de armazenar dados não estruturados. Diante desse cenário, concebeu-se uma abordagem genérica e flexível que permitiu lidar com tais dados de maneira independente das fontes das quais eram extraídos. Assim, elaboramos o C2RM, adotando o RDF como base para o armazenamento.

Além dos desafios relacionados à estruturação e ao armazenamento dos dados, outro aspecto relevante consistiu em lidar com um Modelo de Linguagem (ML) capaz de realizar inferências sobre textos. Inicialmente, uma das experiências desenvolvidas resultou na publicação de um artigo no qual foi empregada uma Rede Neural Convolucional (CNN) (167). Embora os resultados obtidos tenham sido satisfatórios, observou-se que esse tipo de rede neural apresenta limitações quando aplicado a tarefas de PLN.

A partir da experiência com a CNN, adotou-se no IDEA-C2 o BERT como ML, devido à sua capacidade de adaptação ao domínio de C2. Para tal, publicamos um artigo explorando o IDEA-C2 com a abordagem *Singlecategory*. Esse artigo focou nos subprocessos

Ajustar modelo de linguagem e **Aplicar modelo de linguagem** (5). Posteriormente, visando à melhoria das métricas do IDEA-C2-LM, a pesquisa avançou para o subprocesso **Anotar corpus**, propondo uma estratégia de pré-anotação e curadoria do corpus (2).

Finalmente, elaboramos o quarto artigo com foco no subprocesso **Realizar modelagem conceitual**, apresentando a proposta da abordagem que combina os conceitos TD e DD. Esse artigo foi submetido a um *journal* e ainda encontra-se em avaliação. Além disso, elaboramos um quinto artigo, ainda não submetido, que tornou a abordagem IDEA-C2 híbrida. Isso permitiu a incorporação de um vocabulário externo ao IDEA-C2, utilizando a abordagem *Multicategory*.

No próximo capítulo, a abordagem IDEA-C2 é implementada através de um protótipo que é utilizado para realizar experimentos, os quais são objetos de estudo neste trabalho para avaliação de resultados.

5 IDEA-C2-TOOL: IMPLEMENTAÇÃO DA IDEA-C2

Neste capítulo são apresentados os detalhes da implementação da abordagem IDEA-C2, destacando os aspectos técnicos e os componentes arquiteturais, detalhado na seção 5.1. Além disso, é apresentado o protótipo IDEA-C2-Tool que implementa os pacotes de software da arquitetura da abordagem IDEA-C2, detalhado na seção 5.2.

5.1 Arquitetura da abordagem IDEA-C2

Nesta seção, são apresentados os motivos e as justificativas consideradas para definir o uso da arquitetura em camadas na abordagem IDEA-C2, detalhado na subseção 5.1.1. Além disso, na subseção 5.1.2, é demonstrada a arquitetura em detalhes, incluindo a sua estruturação.

A visão arquitetural tem como objetivo representar a estrutura estática do IDEA-C2 através de camadas, pacotes e componentes de software, distinguindo as responsabilidades, bem como a interação entre cada item da estrutura. A arquitetura do IDEA-C2 foi inspirada na Arquitetura em Camadas (*Layered Architecture*) (99), alinhada aos macroprocessos da abordagem IDEA-C2. Nesse sentido, é necessário que as responsabilidades de cada camada se concentre nas funcionalidades ofertadas, permitindo que a implementação dos pacotes esteja aderente à arquitetura imposta.

5.1.1 Justificativa da Arquitetura em Camadas

A escolha do tipo arquitetural em Camadas ocorreu em função deste tipo oferecer um equilíbrio adequado entre acoplamento, granularidade, escalabilidade e complexidade, atendendo às necessidades e às interações entre os processos do IDEA-C2, favorecendo a organização e clareza das responsabilidades sem a sobrecarga das arquiteturas distribuídas. A arquitetura de um software deve focar na especificação necessária para detalhar os aspectos técnicos, ferramentas e métodos, porém de forma indireta e conceitual. Além disso, não pode haver riscos de superespecificar os elementos técnicos, tampouco de subespecificá-los ao ponto de ocultar o que é necessário (168).

Um dos desafios da área de desenvolvimento de software é implementar sistemas de informação com baixo acoplamento e alta coesão (169, 170). Porém, os desafios para atingir esse objetivo não são simples e demandam tanta experiência do arquiteto de software, bem como do tipo arquitetural selecionado, incluindo a linguagem e a tecnologia aplicada no contexto do negócio. Apesar da arquitetura Orientada a Serviços (SOA) ser muito utilizada em sistemas de informação, decidimos não aplicá-la por não haver requisitos

de interoperabilidade tampouco oferta, publicação, negociação e consumo de serviços em nossa abordagem.

Quando comparada à arquitetura de microsserviços, a granularidade da arquitetura em camadas é de nível médio, permitindo a estruturação do sistema em camadas logicamente distintas, preservando a separação de funcionalidades em conformidade com os processos do IDEA-C2. Essa organização favorece o agrupamento de múltiplos módulos e classes com responsabilidades correlatas em uma mesma camada. Com isso, a organização também contribui para a manutenibilidade, uma vez que as classes não se encontram dispersas por todo o sistema.

Por um lado, uma das características da arquitetura em camadas é o alto acoplamento, principalmente em função dela concentrar as camadas em um mesmo contexto de execução. Por outro lado, esse tipo arquitetural favorece a simplicidade de implementação e entendimento das funcionalidades, bem como a evolução do software (99). Mesmo assim, para lidar com o acoplamento, utilizou-se o princípio de dividir para conquistar na construção dos componentes (170). Dessa forma, mantém-se a alta coesão, reduzindo os problemas de acoplamento de conteúdo (*Content Coupling*), não permitindo que uma função altere o conteúdo de outra função ou procedimento. Além disso, variáveis globais não são compartilhadas, evitando problemas de *Common Coupling* (171).

Cabe destacar que a escalabilidade média da arquitetura em camadas é suficiente para o contexto proposto, permitindo a replicação da aplicação completa em múltiplas instâncias sem necessidade de orquestrações complexas. Por fim, a baixa complexidade da arquitetura reduz custos de desenvolvimento, testes e implantação, tornando-a uma escolha pragmática e eficiente para projetos que priorizam a estabilidade e a manutenção a longo prazo.

5.1.2 Especificação da Arquitetura

Com relação à estrutura da arquitetura do IDEA-C2, buscou-se agregar os macroprocessos da abordagem através dos pacotes de software que englobam cada componente. Basicamente, o núcleo da arquitetura possui duas camadas robustas (**Aplicação e Persistência**) em que não há dependência da infraestrutura. Na Figura 22, é ilustrado um diagrama de componentes de acordo com a notação *Unified Modeling Language (UML)*[1], composto de duas camadas gerais, três subcamadas internas e funcionais, as quais são distribuídas e implementadas através de pacotes e componentes de software.

Na UML, um pacote é utilizado para agrupamento lógico de objetos, artefatos de software, componentes ou subsistemas agregados. O componente representa uma parte modular que encapsula a implementação e expõe um conjunto de interfaces (99). Cabe

[1] https://www.omg.org/spec/UML/

destacar que os componentes da arquitetura do IDEA-C2 são modularizados, realizam tarefas específicas e com baixo acoplamento entre eles. O detalhamento dos elementos arquiteturais são descritos a seguir.

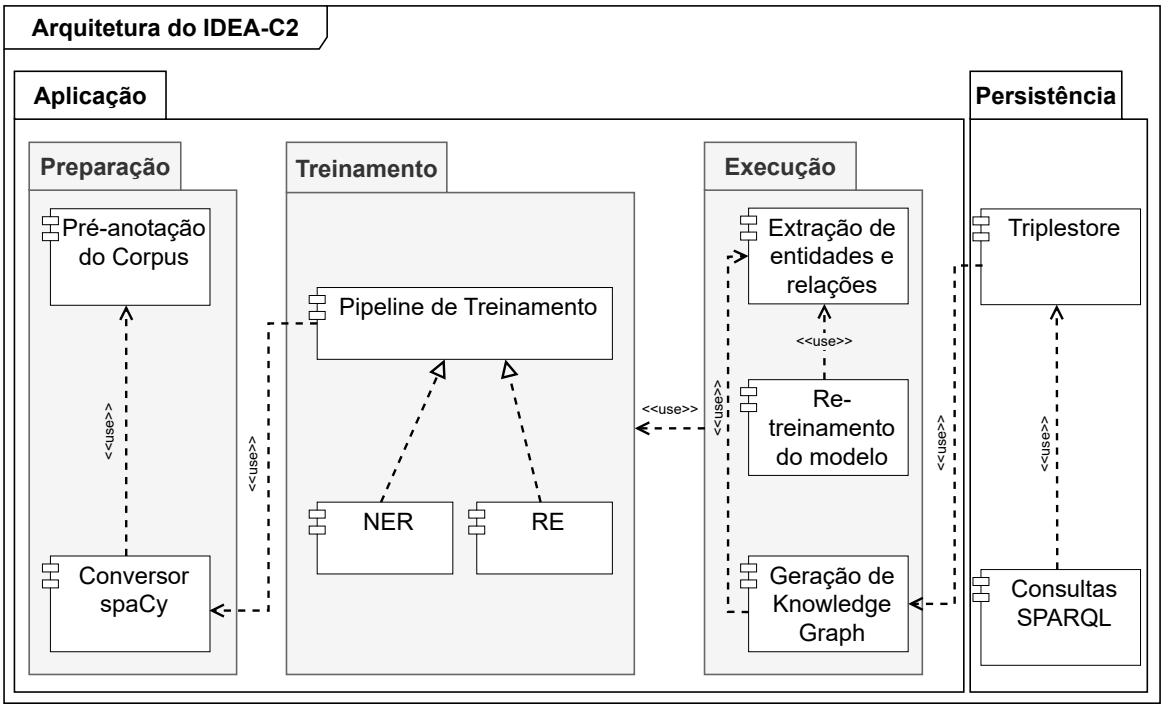


Figura 22 – Arquitetura de software do IDEA-C2. Imagem do autor.

A camada *Aplicação* engloba as principais subcamadas, pacotes e componentes da arquitetura desde a preparação do corpus, incluindo o treinamento e a execução do modelo de linguagem para gerar o KG. A camada *Persistência* não envolve implementação. Na realidade, são componentes de gerência e armazenamento de dados, essencialmente implementados por meio da biblioteca RDFLib[2], que são armazenados no formato RDF Turtle[3], permitindo que seja visualizado e manipulado através de *Triplestores*, como exemplo o GraphDB[4].

A subcamada *Preparação*[5] implementa o subprocesso **Anotar Corpus** e é composta pelos componentes **Pré-anotação do corpus** e **Conversor spaCy**. O componente **Pré-anotação do corpus** é responsável por implementar as rotinas que visam pré-anotar os textos com base em regras de expressão regular a partir do corpus de entrada e das especializações do C2RM. Ao término, é gerado um arquivo no formato JSONL anotado que é importado pela ferramenta Doccano (75). Por sua vez, o componente Conversor spaCy recupera o arquivo JSONL gerado pelo Doccano. Em seguida, o arquivo JSONL

[2] https://rdflib.readthedocs.io
[3] https://www.w3.org/TR/rdf12-turtle/
[4] https://graphdb.ontotext.com/
[5] A subcamada Preparação foi implementada através do protótipo PREAnoTeTool (https://github.com/jonesavelino/preanotetool).

é embaralhado, através de uma rotina randomizada, e dividido na proporção de 80%, para treino e validação, e 20% para teste. Esses novos arquivos são convertidos no padrão intercambiável (*.spacy*) utilizado no treinamento do modelo de linguagem.

A subcamada *Treinamento* implementa o subprocesso Ajustar ML e as rotinas de parametrização do ML, incluindo a instanciação da biblioteca SpaCy[6], até a geração dos modelos de linguagem ajustados ao contexto (IDEA-C2-LM). Na realidade, são gerados dois ML ajustados, um na tarefa NER e outro em RE. O componente *Pipeline de Treinamento* é responsável por parametrizar e instanciar a biblioteca SpaCy a partir dos arquivos de configuração disponibilizados. Nesse arquivo de configuração são definidos os hiperparâmetros, destacados no Quadro 11, nas categorias *Transformer*, *Modelo de Linguagem* e *Pipeline*.

Cabe destacar que NER e RE representam realizações de componentes individuais e especializam rotinas como classificadores. Embora esses componentes sejam funcionalmente dependentes do *Pipeline de Treinamento*, eles também definem parâmetros específicos para cada tarefa. Por exemplo, o tamanho do *batch size*, que define o tamanho do lote do texto, ou *max length*, responsável pelo comprimento máximo do documento que o ML vai processar. Para cada tarefa é gerado um modelo ajustado ao domínio específico. Esses ML ajustados são avaliados através de métricas de desempenho como precisão, *recall* e F1-Score.

A subcamada *Execução* implementa os subprocessos Aplicar ML ajustado e Realizar modelagem conceitual. Além disso, nesta subcamada são implementadas as rotinas de identificação de entidades e extração de relações a partir de textos submetidos ao IDEA-C2-LM, incluindo o seu retreinamento, até a geração do IDEA-C2-KG. Enquanto que no pacote *Extração de entidades e relações*, os textos de C2 são submetidos ao IDEA-C2-LM com o objetivo de extrair as entidades e relações. O pacote *Retreinamento do modelo* recupera textos submetidos ao IDEA-C2-LM e remete ao Doccano para ser agregado ao corpus de treinamento. A única diferença é que ele recupera o texto com as anotações obtidas pelo ML ajustado. Por fim, no pacote *Geração de Knowledge Graph*, as listas de entidades e relações extraídas pelo IDEA-C2-LM são recuperadas e separadas em duas listas. Uma das listas contém as entidades e a outra contém as triplas de pares de entidades e relações. Para adequar as triplas ao padrão RDF, são aplicadas as regras de mapeamento entre as subsespecializações do C2RM com as propriedades do RDF, como mencionado no Quadro 9.

Nesta subseção, os detalhes técnicos que compõem a arquitetura de software adotada no IDEA-C2 foram apresentados. Na seção 5.2, é apresentado o protótipo que utiliza a arquitetura de software.

[6] https://spacy.io/

5.2 Protótipo IDEA-C2-Tool

Nesta seção, é apresentado o protótipo de software IDEA-C2-Tool[7] que foi implementado com o objetivo de realizar experimentos. Além disso, o Quadro 11 reúne as categorias de especificações técnicas essenciais para implementar IDEA-C2-Tool.

Quadro 11 – Especificações técnicas do IDEA-C2-Tool.

	Especificação técnica	**Observações**
Linguagem de Programação	Python	Implementar o código-fonte do protótipo (versão 3.10.12).
Bibliotecas	SpaCy	Manipulação de rotinas de PLN (versão 3.7.2).
	RDFLib	Geração e manipulação do grafo RDF (versão 7.1.4).
	PyPDF2	Extração de textos dos arquivos .PDF (versão 2-3.0.1).
Transformer	*spacy-transformers*	Componente do spaCy para BERT pré-treinados, XLNet e GPT-2 .
Ambiente de desenvolvimento de software	Google Colaboratory	Serviço hospedado do Jupyter Notebook (Versão Pro).
Modelo de Linguagem	neuralmind/bert-base-portuguese-cased	BERTimbau (ML em português).
Pipeline	pt_core_news_sm	Tarefas: NER e RE.
	Tokens	Padrão.
	Word Embeddings	Padrão.
Ambiente de Treinamento	GPU A100 80GB SXM4 (NVIDIA)	-
	80GB GPU Memory	-
	167GB Available RAM	-
	235GB Hard Disk	-
Anotação de Textos	Doccano	Ferramenta de anotação.
Triplestore	GraphDB 10.4.1	Manipulação de grafos RDF.
Mapeamento do componente arquitetural com o código-fonte	Pré-anotação do Corpus	Pre_Anotacao.ipynb
	Conversor spaCy	ConverterDoccanoSpacy3_2.ipynb
	Pipeline NER	FineTuneBERT_Spacy_NER.ipynb
	Pipeline RE	FineTunBERT_Spacy_RE.ipynb
	Extração de entidades e relações	RodaModeloNEReRE.ipynb
	Retreinamento	Retroalimentacao.ipynb
	Geração do KG	Graph.ipynb

[7] <https://github.com/comp-ime-eb-br/S2C2-IME/tree/main/deliverables/idea-c2>

Para o desenvolvimento do protótipo, além da definição arquitetural, algumas decisões de projeto foram tomadas. Essas decisões definiram como alguns artefatos foram implementados ou reusados. A partir dessas decisões foi possível estruturar o IDEA-C2-Tool em dez categorias de especificação técnica, como mencionadas no Quadro 11. As categorias englobam desde a definição da linguagem de programação, incluindo as bibliotecas instanciadas, o modelo de linguagem, dentre outras. A seguir são detalhadas cada categoria de especificação técnica utilizadas no IDEA-C2-Tool.

- **Linguagem de programação:** Com base nos estudos realizados, a linguagem Python foi definida para a implementação do código-fonte do IDEA-C2-Tool. Adotou-se o Python tendo em vista que ele vem sendo amplamente utilizado pela comunidade acadêmica em rotinas de aprendizado profundo e modelos de linguagem. Além disso, nele há *frameworks* robustos e largamente testados, como PyTorch, TensorFlow e Hugging Face Transformers, SpaCy, etc. Outro ponto levado em consideração é que há um conjunto vasto de bibliotecas que agilizam o desenvolvimento, principalmente relacionado ao PLN, tais como: NLTK, Pandas, NumPy, etc. Por fim, a linguagem Python, além de ser *open source*, é compatível com programação de alto desempenho, permitindo executar o código-fonte utilizando *Graphics Processing Unit* (GPU). O GPU foi projetado para arquitetura paralela com vários núcleos de processamento a fim de atender a alta demanda, além de ser essencial no ajuste fino de ML.

- **Bibliotecas:** São pacotes de software implementados, amplamente testados que podem ser integrados ao código-fonte e reutilizados, minimizando erros de codificação e permitindo que o foco seja direcionado na lógica do negócio.

  - **SpaCy:** No levantamento para definir a biblioteca de manipulação de rotinas de PLN, foram considerados o Hugging Face Transformers, o TensorFlow e o SpaCy. Apesar de todas serem amplamente utilizadas, o SpaCy[8] foi adotado em função de possuir uma documentação rica, dotada de exemplos ilustrativos.
  - **RDFLib:** No levantamento, foram consideradas as bibliotecas RDFLib[9] e Owlready2. O RDFLib é uma biblioteca voltada para manipulação de grafos RDF, oferecendo suporte de *parsing* e serialização a diversos formatos, incluindo documentação acessível. O Owlready2, apesar de robusto, ele é voltado a manipulação de ontologias com suporte a OWL 2.0. Nesse sentido, em função de não haver requisitos no IDEA-C2 para explorar OWL, o mais indicado foi adotar o RDFLib.
  - **PyPDF2:** Nessa categoria, foram consideradas as bibliotecas PyPDF2 e PyMuPDF. Contudo, foi selecionada PyPDF2 em função de facilidade de im-

[8] https://spacy.io/
[9] https://rdflib.readthedocs.io/

plementação, robustez para lidar com arquivos grandes e acesso facilitado à documentação técnica.

- ***Transformer*:** Em função da adoção da biblioteca SpaCy, foi definido o pacote *spacy-transformers*[10] que é indicado na documentação oficial para lidar com o ajuste fino de ML, como o BERT.

- **Ambiente de desenvolvimento de software:** Nessa categoria, foram considerados basicamente dois tipos de ambientes de desenvolvimento, um local e outro virtual. Inicialmente, foi realizada uma tentativa de desenvolvimento local, instalando todo o ambiente de desenvolvimento na própria máquina. Nos testes realizados, foram identificados problemas de performance no ajuste fino dos modelos de linguagem, inviabilizando a adoção local. Constatamos que o ajuste fino de um ML, na tarefa de NER, levaria em média 5 horas. Em paralelo, foram realizadas tentativas no ambiente virtual através do Google Colab (sem o serviço de assinatura oferecido pela plataforma). Em nossos testes de bancada, o desempenho do Google Colab foi muito superior à iniciativa do ambiente local, algo em torno de 50% a menos do tempo gasto. Porém, a versão sem assinatura possui uma limitação de tempo de uso associada às unidades computacionais disponíveis. Além disso, nessa versão há restrições que limitam o acesso à infraestrutura robusta com GPU. Nesse sentido, em função do desempenho, custo e tempo, optou-se por investir na plataforma virtual Google Colab Pro[11] (serviço pago) que oferece diversas infraestruturas tecnológicas de alto desempenho, as quais podem ser incrementadas a qualquer tempo mediante à sua aquisição disponível na própria plataforma. Ademais, a plataforma permite rodar o código-fonte de qualquer computador ligado à internet, armazenando o histórico, organizando o controle de versão já interligado ao Github e permitindo o acesso aos dados em um único lugar.

- **Modelo de linguagem:** Em função dos resultados do trabalho de Souza, Nogueira e Lotufo(36) e pela escassez de ML baseados na arquitetura *transformer* em língua portuguesa, optou-se em utilizar o BERTimbau através do pacote *neuralmind/bert-base-portuguese-cased*[12]. Cabe ressaltar que foram testados outros pacotes, como abordado nos experimentos relatados no Capítulo 6.

- **Pipeline:** O *pipeline pt_core_news_sm* obteve os melhores resultados ao ser comparado com outros *pipelines*, como abordado no Capítulo 6.

- **Infraestrutura tecnológica:** Como foi selecionado o Google Colaboratory Pro, alguns testes foram executados com diferentes infraestruturas, a configuração GPU

[10] https://github.com/explosion/spacy-transformers
[11] https://colab.google/
[12] https://huggingface.co/neuralmind/bert-base-portuguese-cased

A100 80GB SXM4 (NVIDIA) alcançou os melhores resultados, porém os custos de unidades de processamento são relativamente altos. Por exemplo, ao utilizar esta configuração com 295 mil unidades de processamento, a taxa de uso cessa em cerca de oito horas.

- **Anotação de textos:** As ferramentas Doccano[13], UBIAI, Prodigy e BRAT foram consideradas e avaliadas. Apesar de Prodigy (integração com SpaCy) e UBIAI possuírem mais recursos, os custos são onerosos e tornariam a pesquisa inviável em função das regras de uso das ferramentas. Em relação ao BRAT, apesar de *open source*, é uma ferramenta mais antiga e possui poucos recursos. Nesse sentido, foi selecionado o Doccano em função de ser *open source*, possuir uma interface amigável para curadoria, ele também permite a importação de blocos de textos JSON em linhas. Por outro lado, como o Doccano não é integrado à biblioteca SpaCy, foi necessário desenvolver um conversor do formato JSON para o padrão proprietário do SpaCy (componente Conversor spaCy).

- **Triplestore:** Apesar da biblioteca RDFLib ser estável e oferecer algumas facilidades de uso, em alguns casos é mais adequado o uso de um *Triplestore*, principalmente quando envolve consultas complexas. Assim, foram avaliados o Apache Jena e o GraphDB. Apesar do Apache Jena ser *open source*, ele não possui uma interface amigável. Em contrapartida, o GraphDB é rico em interface e suporte de uso. Apesar de ser utilizado comercialmente, há uma versão *Free* que atende as necessidades da pesquisa.

- **Mapeamento do Componente arquitetural com o código-fonte:** Esta categoria descreve o mapeamento dos cadernos de implementação do código-fonte no Google Colaboratory que foram codificados à luz dos componentes arquiteturais do IDEA-C2. Cada caderno possui a documentação do código, descrevendo os objetos de entrada, bem como o passo a passo de execução e os resultados podem ser visualizados a cada bloco de instrução de código-fonte executado.

Além dessas categorias de especificações técnicas, foram implementadas funções modulares que permitem o reúso de código, na pasta C2IME, localizada no repositório do projeto[14]. Essas funções são fruto do projeto de modularização do protótipo e têm como objetivo apoiar a execução dos códigos fontes do IDEA-C2. A seguir é detalhada cada função.

- **accuracy_functions:** Atua na medição dos resultados da pré-anotação dos textos.

[13] https://github.com/doccano/doccano
[14] <https://github.com/comp-ime-eb-br/S2C2-IME/tree/main/deliverables/idea-c2/C2IME>

- **enrich_class:** Atua na pré-anotação dos termos, aplicando as regras de expressão regular a partir dos pares de recursos semânticos e parte do texto a ser buscado no corpus.
- **file_functions:** Atua em conjunto com a biblioteca PyPDF2 para extrair os textos dos arquivos no formato PDF.
- **graph_functions:** Atua em conjunto com a biblioteca RDFLib para manipular o grafo RDF.
- **graph_rules:** Atua em conjunto com graph_functions para implementar as operações semânticas, sobre os grafos RDF, vinculadas as propriedades ou especializações de C2RM.
- **main_function:** Organiza o código-fonte de execução fornecendo uma interface única para a execução, via delegação, de outras classes e operações do IDEA-C2, possui um comportamento análogo ao padrão de projeto *Facade* (107).
- **rdf_functions:** Atua em conjunto com graph_functions na serialização das triplas geradas do grafo RDF.
- **relations_functions:** Atua em conjunto com enrich_class na aplicação das regras de pré-anotação, mais precisamente na busca das regras de expressão regular parametrizadas.
- **terms_functions:** Responsável por implementar rotinas que são utilizadas em file_functions, relations_functions, dentre outras funções.

Nesta seção, foram apresentados os pontos relevantes da implementação do IDEA-C2, que abordou desde a especificação arquitetural até os aspectos de implementação do protótipo IDEA-C2-Tool. No Capítulo 6, será abordado sobre a validação dos resultados dos experimentos executados no IDEA-C2-Tool.

6 EXPERIMENTOS E VALIDAÇÃO

Como abordado na seção anterior, o protótipo IDEA-C2-Tool foi construído a fim de apoiar a realização dos experimentos que exploram os cenários de aplicação relacionados ao domínio militar. Para tanto, seis experimentos foram executados para validar as hipóteses da abordagem IDEA-C2 por meio de avaliações quantitativas e qualitativas. Dessa forma, de acordo com os resultados alcançados, foi possível avaliar se a abordagem atende ou não ao seu propósito, bem como a sua utilidade no contexto do estudo de caso.

Neste capítulo, os experimentos ($Ex_1, ...Ex_6$) são organizados e discutidos nas seções de 6.1 a 6.6. Cada seção, explora o cenário de aplicação, incluindo o objetivo e um breve resumo do experimento, além de detalhar a metodologia utilizada na avaliação, bem como os resultados alcançados. Os experimentos Ex_1, Ex_2 e Ex_3 são apresentados nas seções de 6.1 a 6.3, e discutidos resumidamente por já terem sido demonstrados em trabalhos anteriores (2, 5).

Nas seções 6.4 e 6.5, são apresentados os experimentos que confirmam a hipótese H4. Apesar de os experimentos Ex_4 e Ex_5 estarem alinhados à hipótese H4, eles se diferenciam em função do conjunto de textos alvo e da sua amplitude de aplicação. Na seção 6.3, é apresentado o experimento Ex_6, que estende o Ex_1, permitindo a incorporação de uma taxonomia no ajuste fino de um ML, utilizando a abordagem *multicategory*. Por fim, na seção 6.7, é apresentada uma análise crítica dos experimentos, avaliando os resultados alcançados, as limitações e os pontos de melhoria para aperfeiçoar a abordagem IDEA-C2.

6.1 Ex_1: Geração de Modelos de Linguagem na abordagem *Singlecategory*

O experimento Ex_1 tem como objetivo validar a hipótese H2 que propõe o seguinte: "Um metamodelo que permite metacategorizar as entidades e relações pode flexibilizar a anotação de um corpus para o ajuste fino de um ML nas tarefas NER e RE." Esse experimento foi apresentado em detalhes, no trabalho de Avelino et al.(5), cujos resultados são apresentados nas Tabelas 1 e 2, e disponíveis para consulta no GitHub[1].

Para atingir o objetivo deste experimento, ele foi executado em duas etapas. Na primeira, o metamodelo C2RM foi aplicado a partir de corpora fora do contexto de C2, demonstrando que o metamodelo pode ser aplicado em diferentes domínios. Dessa forma, a estratégia **Singlecategory** foi comparada com as abordagens tradicionais que

[1] <https://github.com/comp-ime-eb-br/S2C2-IME/tree/main/deliverables/idea-c2/experimentos/exp1>

utilizam **Multicategory** a fim de avaliar a sua viabilidade de aplicação. Cabe destacar que utilizamos também ambas as estratégias na tarefa de RE em função dos resultados da tarefa de NER interferirem no experimento. Na segunda etapa, o C2RM foi aplicado em um corpus formado por textos de Doutrina Militar (DML) no contexto de C2, utilizando a abordagem *Singlecategory*, para realizar o ajuste fino com base na instância de um ML.

Tabela 1 – Geração do IDEA-C2-LM (*Multicategory* vs. *Singlecategory*). Adaptado de Avelino et al.(5).

Corpus	Tipo	Modelo de linguagem	Tarefa	*Precision*	*Recall*	*F1-Score*
SciERC	Multi	allenai/scibert	NER	65,11%	63,20%	64,14%
			RE	48,27%	20,21%	28,49%
	Single		**NER**	**76,67%**	**79,13%**	**77,88%**
			RE	43,68%	**26,13%**	**32,70%**
Material Science	Multi	roberta-base	**NER**	**79,37%**	**79,09%**	**79,23%**
			RE	49,76%	29,64%	34,66%
	Single		NER	70,46%	75,46%	77,41%
			RE	43,28%	**40,62%**	**41,91%**

Multi: Multicategory; Single: Singlecategory;

Na primeira etapa, foram utilizados os corpora fora do contexto de C2, o **SciERC** (9), da área de Inteligência artificial, e o **Material Science** (24), da área de Ciência dos materiais. Ambos os textos dos corpora são expressos na língua inglesa. Inicialmente, os corpora foram copiados para outro local de armazenamento para serem manipulados. Em cada cópia, as anotações foram ajustadas em conformidade com a abordagem *Singlecategory*. No caso do SciERC, por exemplo, as categorias de entidades pré-definidas *task, method, evaluation metric, material, evaluation metric* e *generic* foram substituídas pela categoria "ENTITY". Para o ajuste fino, os hiperparâmetros originais (e.g. o *dropout* foi definido com 20%), os ML indicados em seus trabalhos foram recuperados, respectivamente, *allenai/scibert* e *roberta-base*, e as tarefas foram definidas como NER e RE. Além disso, foi utilizado o *pipeline en_core_web_sm*[2] da biblioteca spaCy. Esse *pipeline* utiliza um vocabulário pequeno e é indicado para lidar com textos em inglês, a partir da instância de um ML baseado na arquitetura *Transformer*.

Na Tabela 1, são apresentados os resultados alcançados no experimento utilizando os corpora SciERC e o *Material Science*. Cabe destacar que os resultados foram alcançados mediante a realização de ajustes finos sucessivos que foram registrados no experimento Ex_3, abordado a seguir, e estão disponíveis para consulta no GitHub[3]. Esses resultados mostraram-se promissores e indicam que a adoção da estratégia **Singlecategory**, além de oferecer maior flexibilidade, alcançou, em alguns casos, resultados superiores quando

[2] https://spacy.io/models/en

[3] <https://github.com/comp-ime-eb-br/S2C2-IME/blob/b681a8bb07a630c4636721707dccd71e4054375c/deliverables/idea-c2/experimentos/exp3/ex3_etapa1_fine_tuning.pdf>

comparados à abordagem *Multicategory*. No caso do SciERC, destacamos que o tipo *Singlecategory* foi superior na tarefa NER em todas as métricas. E somente alcançou um valor ligeiramente inferior na métrica *precision* na tarefa de RE. Por sua vez, ao analisarmos o corpus *Material Science*, os resultados de *Multicategory*, na tarefa NER, alcançaram resultados superiores em todas as métricas. Por sua vez, na tarefa RE, o tipo *Singlecategory* alcançou resultados superiores nas métricas *Recall* e *F1-score*. Dessa forma, as evidências apontam que a adoção da abordagem *Singlecategory* é propicia em alguns casos e permite gerar ML ajustados com resultados promissores.

Tabela 2 – Resultado das métricas do ajuste fino do IDEA-C2-LM (Singlecategory), utilizando um corpus de C2. Adaptado de Avelino et al.(5).

		Resultado das métricas		
	Rodada	Precisão	*Recall*	*F1-score*
NER	1	9,93%	17,19%	12,58%
	2	86,56%	86,48%	86,51%
RE	1	0,36%	56,48%	0,72%
	2	98,06%	98,37%	98,21%

Na Tabela 2, são apresentados os resultados da geração do IDEA-C2-ML, utilizando a abordagem *Singlecategory* a partir de um corpus composto de mais de 3 mil textos a área de C2 e utilizando o *pipeline* ***pt_core_news_sm***. Para tal, o experimento foi dividido em duas rodadas distintas, destacando os níveis de maturidade da abordagem em momentos diferentes da pesquisa. Em cada rodada, na realidade são executados refinamentos e testes sucessivos com valores e hiperparâmetros distintos com objetivo de gerar o ML ajustado com o melhor desempenho. A evolução dos resultados em cada teste, bem como a aplicação em outras instâncias de ML, são detalhadas no Ex_3, na seção 6.3, disponíveis para consulta através do GitHub[4], discriminando as datas das ocorrências dos testes, bem como os valores dos hiperparâmetros, os modelos de linguagem, os *pipelines* utilizados e os valores das métricas alcançadas em detalhes.

Na primeira rodada, o experimento foi executado em função da submissão de um artigo ao *Proceedings of the 16th International Conference on Computational Processing of Portuguese* (PROPOR 2024)[5]. Na oportunidade, em meados de 2023, a abordagem IDEA-C2 estava ainda incipiente, o corpus com poucas anotações e, consequentemente, os resultados também não estavam bons. Porém, como a abordagem *Singlecategory* é uma proposta diferente dos trabalhos do estado da arte, apostamos estrategicamente na submissão com o objetivo de obtermos um *feedback* dos revisores da conferência. Nesse sentido, o objetivo foi alcançado com êxito, destacando que o artigo ficou até a fase final da seleção, além é claro de uma boa revisão do nosso trabalho. Nessa rodada, o corpus

[4] <https://github.com/comp-ime-eb-br/S2C2-IME/blob/b681a8bb07a630c4636721707dccd71e4054375c/deliverables/idea-c2/experimentos/exp3/ex3_etapa2_fine_tuning.pdf>

[5] https://aclanthology.org/2024.propor-1.0/

utilizado possuía em torno de 150 anotações e os hiperparâmetros foram utilizados com seus valores padrão. Por isso que os resultados alcançaram resultados abaixo do esperado. No entanto, os valores dessa primeira rodada demonstraram uma tendência da abordagem *Singlecategory* de alcançar valores altos de *recall*, com 17,19% na tarefa NER e 56,48% na tarefa RE, em relação à precisão. Isso nos leva a crer que ao adotar esse tipo de abordagem, o ajuste do ML prioriza a sensibilidade ou cobertura do ML em detrimento da seletividade ou precisão, principalmente porque as anotações das categorias de entidades não são distribuídas. Esse comportamento evidencia que o ajuste do ML recuperou a maioria dos exemplos relevantes (poucos falsos negativos), classificando um número maior de instâncias como positivas.

Na segunda rodada, utilizamos o corpus com mais de 600 termos anotados e os hiperparâmetros foram ajustados através de refinamentos sucessivos. Os resultados alcançados representaram uma maior maturidade da pesquisa e foram baseados no artigo Avelino et al.(5), apresentado, em meados de 2024, na 26ª ICEIS (International Conference on Enterprise Information Systems). Ao analisarmos os resultados aqui expressos sob a ótica da abordagem *Singlecategory* podemos tirar algumas conclusões. Note que os resultados evidenciam que o *recall* mantém-se alto em ambas as tarefas, com 86,48% para NER e 98,37% para RE. Porém, esses valores estão mais equilibrados à métrica *precisão* em ambas as tarefas, com 86,56% para NER e 98,06% para RE. Dessa forma, esses resultados evidenciam que o ML ajustado, utilizando a abordagem *Singlecategory*, está calibrado e confiável para realizar as inferências necessárias quando os textos a ele forem submetidos, demonstrando que esse tipo de abordagem é viável nas tarefas NER e RE.

Portanto, de acordo com as evidências do experimento, a abordagem *Singlecategory* atende à proposição da hipótese H2, demonstrando que um metamodelo que utiliza metacategorias de entidades e relações é capaz de flexibilizar a anotação de um corpus para o ajuste fino de um ML nas tarefas NER e RE. Além dos resultados alcançados, demonstrou-se a flexibilidade da aplicação de C2RM em contextos distintos. É oportuno destacar que a abordagem de categoria única, além de válida, pode também ser útil para gerar um KG, principalmente quando a definição de categorias pode ser postergada e realizada no próprio KG.

6.2 Ex_2: Avaliação da anotação semiautomatizada no IDEA-C2

O experimento Ex_2 tem como objetivo validar a hipótese H3 que propõe o seguinte: "É possível que o metamodelo combinado com uma pré-anotação heurística e RS aplicado em um corpus pode gerar ML ajustados cujas métricas de avaliação são equiparáveis ao estado da arte." Esse experimento foi apresentado no trabalho de Avelino et al.(2) e avaliou a eficácia da estratégia de anotação do IDEA-C2, comparando a pré-anotação

semiautomatizada de um corpus com a anotação manual, como veremos adiante.

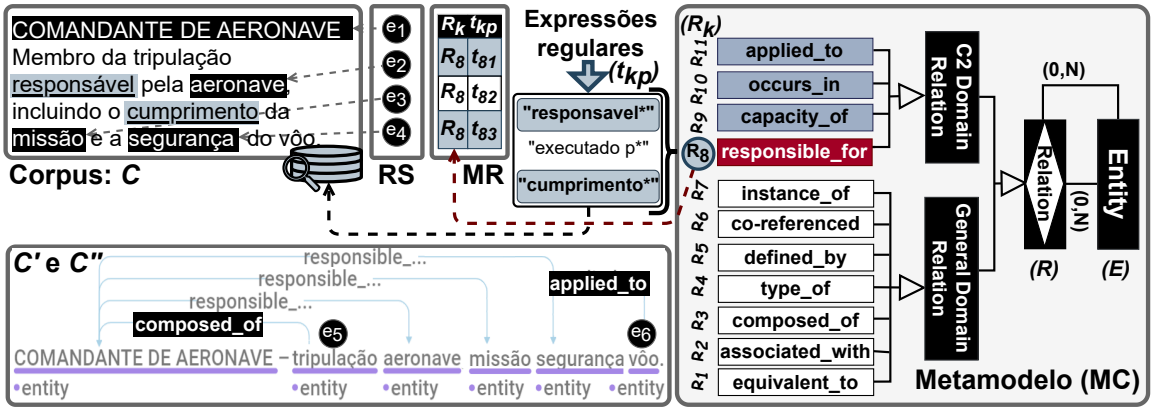


Figura 23 – Pré-anotação do corpus C no IDEA-C2. Adaptado de Avelino et al.(2).

Na Figura 23, é ilustrado um exemplo de pré-anotação do IDEA-C2-Tool que implementa os componentes da subcamada **Preparação** da arquitetura da abordagem IDEA-C2. No exemplo, é explorada a pré-anotação a partir de uma amostra do corpus C, constituído de mais de 3 mil trechos de textos Doutrinas Militares (DML) e do Glossário de Termos do EB (1, 117, 165, 172, 173). Em destaque, os conjuntos, MC e MR, que representam o metamodelo C2RM e o mapeamento das regras, respectivamente. Na parte inferior, o trecho de C é destacado com as entidades sublinhadas, em lilás, e as relações pré-anotadas e rotuladas de C', representadas por setas ligando as entidades.

A validação das pré-anotações de C' é realizada por amostragem a partir da criação dos subcorpora de textos, SM e SC'. O conteúdo desses subcorpora é formado por 10% de amostras, selecionadas aleatoriamente, de C e C', respectivamente. Além disso, o subcorpus SM foi distribuído a um usuário especialista a fim de realizar a anotação manual com o suporte da ferramenta Doccano. Como resultado da anotação manual, foi gerado o subcorpus SM'.

Na validação de C', o SM' é considerado como *gold standard* para comparar o conjunto de trechos pré-anotados de SC'. Na parte inferior da Figura 23, note que em C'' as anotações destacadas *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 foram pré-anotadas. Finalmente, essas anotações compõem o corpus curado C'' que foi utilizado para gerar o IDEA-C2-LM. Ademais, os resultados da validação de C' foram processados a fim de avaliar a eficácia da pré-anotação. Esses resultados são apresentados na Tabela 3 e estão disponíveis para consulta no GitHub[6].

Durante a validação da pré-anotação entre SC' e SM' foram considerados os valores das variáveis *True Positive (TP)*, *False Positive (FP)* e *False Negative (FN)*, cujos

[6] <https://github.com/comp-ime-eb-br/S2C2-IME/tree/main/deliverables/idea-c2/experimentos/exp2>

Tabela 3 – Pré-anotação baseada em Regras (SC') x Anotação manual (SM'). Adaptado de Avelino et al.(2).

		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%

resultados foram obtidos através da análise comparativa das instâncias anotadas de ambos os subcorpora. Assim, se o usuário especialista anotou em SM' um termo, por exemplo "aeronave" como *entity*, e_2, e a pré-anotação de SC' também o fez, então temos uma ocorrência de $TP = |(SC' \cap SM')|$. Caso o termo anotado pelo usuário especialista não tenha sido pré-anotado, por exemplo "tripulação", e_5, e "voo", e_6, considera-se $FN = |(SM' - SC')|$. O contrário também pode ocorrer, representando um $FP = |(SC' - SM')|$.

Similarmente, o mesmo tipo de validação é feita para as relações, porém de duas formas distintas. Inicialmente, são avaliadas as relações de entidades considerando os rótulos (avaliação chamada de "categorizadas"). Por exemplo, uma ocorrência TP foi considerada ao comparar a tripla da relação entre as entidades ("comandante de aeronave", e_1,***responsible_for***, "aeronave", e_2). Porém, ocorreu um FN na tripla da relação entre as entidades ("tripulação", e_5, ***composed_for***, "comandante de aeronave", e_1), a qual foi anotada somente pelo usuário especialista e sem um correspondente em SC'. Como o contrário também pode ocorrer, vamos supor que somente em SC' foi encontrada a tripla da relação ("comandante de aeronave", e_1, ***responsible_for***,"missão", e_3), então teríamos uma ocorrência de FP. Em um segundo momento, o mesmo procedimento de comparação é realizado entre as relações de entidades, porém independente do rótulo atribuído. Essas ocorrências são denominadas como "não categorizadas").

De acordo com os resultados das variáveis (*TP, FP e FN*), são calculadas as métricas de precisão, *recall* e *F1-Score*, como apresentado na Tabela 3. No geral, com um corpus contendo mais de 3 mil trechos de textos, 770 termos identificados como *entity* foram pré-anotados e os resultados alcançaram a precisão de 95% e 59% de *recall*. No caso das relações "não categorizadas", foram alcançados 76% de precisão e 41% de *recall*, e, respectivamente, 45% e 24%, nas "categorizadas". Os resultados das métricas evidenciam que o uso do RS combinado com o metamodelo e as regras heurísticas contribuiu para a identificação de entidades e relações não categorizadas na pré-anotação, principalmente dados os resultados de precisão e cobertura. Contudo, a pré-anotação ainda deve ser aperfeiçoada quando consideramos o rótulo utilizado entre as entidades relacionadas.

Portanto, de acordo com as evidências do experimento, a pré-anotação atende à proposição da hipótese H3, demonstrando que o metamodelo combinado com uma pré-anotação heurística e RS aplicado em um corpus pode gerar ML ajustados cujas métricas de avaliação são equiparáveis ao estado da arte. Apesar deste experimento ter se

concentrado na pré-anotação, o experimento Ex_3 valida a geração do ML ajustado a partir do corpus C'' aqui gerado. Além dos resultados abordados, a estratégia de pré-anotação contribuiu com a redução de esforços em relação à anotação manual. 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 IDEA-C2 pode contribuir com a redução do esforço e favorecer a construção de um corpus.

Além da redução de esforço, um outro benefício é a mudança na atuação do usuário especialista, que passa a revisar e incrementar a qualidade do corpus anotado, pois a pré-anotação já identifica e pré-anota as entidades e relações. Além disso, como a rotatividade de pessoal é um dos desafios das FAs, a adoção de regras de pré-anotação pode contribuir na retenção e padronização de entendimento dos usuários especialistas. Principalmente porque o conhecimento passa a ser internalizado nas regras elaboradas, minimizando a dependência do indivíduo.

6.3 Ex_3: Avaliação do ajuste fino de Modelo de linguagem em diferentes *pipelines*

O experimento Ex_3 tem como objetivo validar as hipóteses H1 e H3. A hipótese H1 propõe o seguinte: "É possível gerar um KG a partir de textos doutrinários, orientado por um metamodelo e apoiado por um ML ajustado no contexto militar." Já a hipótese H3 propõe o seguinte: "É possível que o metamodelo combinado com uma pré-anotação heurística e RS aplicado em um corpus pode gerar ML ajustados cujas métricas de avaliação são equiparáveis ao estado da arte." Para atingir o objetivo mencionado, os artefatos IDEA-C2-KG e IDEA-C2-LM, respectivamente, ligados às hipótese H1 e H3, foram gerados a partir da implementação dos componentes das subcamadas **Execução** e **Treinamento** da arquitetura da abordagem IDEA-C2, detalhados na seção 5.1. Assim, os componentes **Geração de Knowledge Graph** e **Pipeline de Treinamento** foram instanciados a partir de um processo iterativo com refinamentos sucessivos, utilizando diferentes parâmetros e tipos de ML, a fim de gerar tanto o IDEA-C2-KG quanto o ML ajustado (IDEA-C2-LM) com o melhor desempenho.

Inicialmente, foram testados diferentes hiperparâmetros, tendo como base os valores padrão indicados pela biblioteca SpaCy. Em seguida, a cada iteração foram inseridos novos valores de hiperparâmetros com foco no desempenho do ML. Ao analisar o resultado, ajustes pontuais foram realizados. Com isso, foi possível identificar limitações, corrigir inconsistências e revisar a geração do ML ajustado ao domínio de C2. Cabe destacar que os diários de bordo com todos os detalhes das iterações do experimento estão disponíveis para consulta através do GitHub[7], discriminando as datas das ocorrências dos testes, bem

[7] <https://github.com/comp-ime-eb-br/S2C2-IME/tree/b681a8bb07a630c4636721707dccd71e4054375c/

como os valores dos hiperparâmetros, os modelos de linguagem, os *pipelines* utilizados e os valores das métricas alcançadas em detalhes.

Para a formação do corpus, foi utilizado um conjunto de textos no domínio militar obtido de forma aleatória com base no Glossário de Termos do EB (1) e alguns trechos de Doutrinas Militares (DML) (1, 165, 123, 66). O corpus é composto em 3.394 sentenças de textos anotados no Doccano com cerca de 22 mil palavras. Para a anotação dos textos, foi utilizado o conjunto de regras de pré-anotação, apresentado na seção 4.3, totalizando cerca de 14 mil termos anotados. Para o ajuste fino, o corpus foi dividido em 80% para treino[8] e validação[9] e 20% para teste[10]. Embora esse experimento já tenha sido apresentado nos trabalhos de Avelino et al.(5) e Avelino et al.(2), no decorrer da pesquisa outras iterações foram realizadas, novos resultados foram obtidos e estão disponíveis para consulta no GitHub[11].

Na Tabela 4, são apresentados os resultados finais alcançados de cada iteração, dividido por tarefa (NER e RE), incluindo o tipo de abordagem (AB), o *pipeline* (PL), subdividido em *small* (SM), *middle* (MD) e *large* (LG), as métricas e os hiperparâmetros. Em destaque, os valores em negrito indicam os *pipelines* que alcançaram as melhores performances por tarefa e abordagem. A coluna "AB" descreve o tipo de abordagem utilizada no ajuste fino, contendo os valores "SG" para *Singlecategory* e "MT" para *Multicategory*. A coluna "métricas" reúne os valores de precisão, recall e F1-score alcançados. Por fim, a coluna "hiperparâmetros" descreve o tipo e o valor de cada hiperparâmetro utilizado para configurar o *pipeline*. Cabe ressaltar que apesar de a tabela de comparação conter ambas as abordagens, neste experimento é discutida somente a abordagem *Singlecategory*. Todavia, a discussão acerca dos resultados de *Multicategory* é apresentada em detalhes no experimento Ex_6, na seção 6.6.

Ao analisar os valores da Tabela 4, verificamos que os *pipelines* **pt_core_news_sm** e **pt_core_news_md** alcançaram os melhores resultados em ambas as tarefas. Por um lado, o **pt_core_news_sm** quando aplicado na abordagem *Singlecategory* (SG), na tarefa NER, apresentou um desempenho consistente, alcançando valores de métricas aproximados ao *Multicategory*, evidenciando um ML equilibrado e capaz de reconhecer entidades com baixa propensão a erros de classificação. Apesar de 304 épocas ser relativamente alto, o que poderia até indicar um *overfitting*, acreditamos que o alcance desse valor tenha ocorrido em função do corpus utilizado ser pequeno, impactado pelo elevado

deliverables/idea-c2/experimentos/exp3>

[8] <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/deliverables/idea-c2/outputs/Dataset_exporta_anotacao-train.jsonl>

[9] <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/deliverables/idea-c2/outputs/Dataset_exporta_anotacao-dev.jsonl>

[10] <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/deliverables/idea-c2/outputs/Dataset_exporta_anotacao-test.jsonl>

[11] <https://github.com/comp-ime-eb-br/S2C2-IME/tree/main/deliverables/idea-c2/experimentos/exp3>

Tabela 4 – Comparação dos *pipelines* utilizados na geração do IDEA-C2-LM. Adaptado dos trabalhos de Avelino et al.(5) e Avelino et al.(2).

	AB	**PL**	**Métricas**			**EP**	**Hiperparâmetros**				
			PR	**RC**	**F1**		**DP**	**BS**	**MX**	**TH**	**MS**
NER	SG	**SM**	**86,56%**	**86,48%**	**86,51%**	**304**	20	128	4096	-	20000
		MD	84,42%	79,07%	81,66%	159					
		LG	85,44%	80,24%	82,76%	323					
	MT	**MD**	**87,61%**	**85,24%**	**86,41%**	**295**					
RE	SG	**SM**	**98,06%**	**98,37%**	**98,21%**	**66**		500	0	0,5	1000
		MD	91,93%	81,28%	86,28%	49		250		0,3	
		LG	91,97%	87,24%	86,54%	46					
	MT	**MD**	**86,67%**	**86,55%**	**86,41%**	**134**	10	1000	400	0,5	

NER: Reconhecimento de Entidades Nomeadas; RE: Extração de relações; PL: *Pipeline*;
AB: Abordagem; SG: *Singlecategory*; MT: *Multicategory*;
SM: *Small*; MD: *Middle*; *Large*; PR: Precisão; RC: *Recall*; F1: F1-Score;
EP: Épocas; DP: *Dropout*; BS:*Batch Size*; MX: *Max Length*;
TH: *Threshold*; MS: *Max Steps*.

número de *Max Steps* de 20000. Talvez, esse valor de *Max Steps* tenha garantido um maior refinamento das camadas do ML e uma possível estabilização do *F1-Score*. Inclusive o valor de 20% de *Dropout* se mostrou adequado para evitar um possível *overfitting*, corroborado pelo resultado alcançado do ajuste fino do ML.

Em contrapartida, o *Batch Size* e o *Max Length*, respectivamente, com 128 e 4096, apesar de serem valores altos, não afetaram o desempenho do ajuste fino do ML. Nesse último, por exemplo, acreditamos que o *Max Length* não tenha interferido no resultado, pois o BERT é limitado a 512 *tokens* (12). Entretanto, esses valores poderiam ser revisitados e testados em iterações futuras com outros números. Por outro lado, ainda na tarefa NER, o *pipeline* **pt_core_news_md** alcançou o melhor resultado quando aplicado na abordagem *Multicategory* (MT). Ao investigar esses resultados, verificamos que ambos os *pipelines* alcançaram valores aproximados. Contudo, as evidências apontam que **pt_core_news_md** pode ser mais indicado para cenários de aplicação em que há distribuição de classes em função de possuir um vetor de *embeddings* de 300 dimensões. Isso foi evidenciado no maior valor de precisão que alcançou 87,61%, bem como pela queda natural do *recall* em função da seletividade de representações que atingiu 85,24%. Mesmo assim, o resultado do *F1-score* de 86,41% ficou ligeiramente abaixo no **pt_core_news_sm**, que alcançou 86,51%, indicando uma estabilidade em ambos os *pipelines*.

Estendendo a análise dos valores da Tabela 4, agora com o foco na tarefa RE, aplicado na abordagem *Singlecategory*, observamos que os resultados alcançados das métricas foram altos. Cabe destacar que o *recall* foi ligeiramente superior à precisão, sugerindo que o ML ajustado é capaz de identificar quase todas as relações do treinamento. Porém, acredita-se que esse resultado tenha sido influenciado por haver somente uma categoria. Ao combinar os hiperparâmetros (*dropout* de 20%, *batch size* de 500 e *threshold*

de 0,5) nos leva a crer que foi uma escolha adequada, inclusive, quando comparado com os resultados dos outros *pipelines*. Além disso, o resultado de 66 épocas, indicado na coluna EP, sugere que o número limitado de relações favoreceu o aprendizado do ML. Esse favorecimento pode ser reforçado quando relacionamos o valor do *dropout* com o resultado das épocas, pois é possível observar que não houve problemas de *overfitting*. Como os resultados são satisfatórios, presume-se que o BERTimbau se ajusta ao domínio e às estruturas relacionais do corpus, alcançando bons resultados na tarefa de extração de relações.

Entretanto, nos testes do IDEA-C2-ML, notamos que a extração de relações é influenciada também pela estrutura gramatical em que ML foi ajustado. Por exemplo, foram submetidas ao IDEA-C2-LM duas sentenças de textos (s_1 e s_2), respectivamente com os seguintes valores: s_1: "Acampamento é uma forma de estacionamento em que a tropa se instala..."; e s_2: "O acampamento da tropa ocorreu em terras inimigas.". Note que s_1 é uma sentença bem parecida com um trecho de texto utilizado no ajuste fino. Já s_2 é uma sentença formulada aleatoriamente e desconhecida do ML. Vamos explorar esses textos a seguir.

Quadro 12 – Exemplo de submissão de s_1 e s_2 a IDEA-C2-LM.

	NER	**RE**
s_1	acampamento; estacionamento; tropa	(t_1) acampamento – *associated_with* – tropa [99,37%]; (t_2) acampamento – *type_of* – estacionamento [97,85%]; (t_3) tropa – *responsible_for* – acampamento [1,45%]; (t_4) tropa – *type_of* – acampamento [0,72%];
s_2	acampamento; tropa;	(t_5) tropa – *responsible_for* – acampamento [11%]

No Quadro 12, são apresentadas algumas variações de inferências resultantes da submissão de s_1 e s_2 ao IDEA-C2-LM. Ao analisar o quadro, na tarefa NER, o IDEA-C2-LM reconheceu todas as entidades conforme esperado. Na tarefa RE, houve alguns acertos, como nos casos das triplas t_1, t_2 e t_3. Porém, em t_3, apesar de inferir corretamente, o grau recomendado pelo ML foi muito baixo. Além disso, ao considerarmos t_4, observamos que o ML inferiu a entidade nomeada "tropa" como *type_of* de "acampamento". Essa inferência é imprecisa e com o grau de confiança muito baixo, devendo ser melhor investigada. Por outro lado, ao analisar s_2, mesmo sendo uma sentença desconhecida, o ML extraiu a relação corretamente. Mesmo assim, o resultado alcançado foi com um desempenho relativamente baixo. Dessa forma, depreende-se da análise do exemplo que o reconhecimento de entidades alcançou resultados promissores. No entanto, as relações entre as entidades ainda é sensível à estrutura em que ela se encontra. Assim, observa-se que o IDEA-C2-LM, na tarefa RE, pode ser aprimorado e melhor ajustado a partir de mais iterações e novos refinamentos sucessivos. Ademais, as entidades e relações obtidas no IDEA-C2-LM podem ser representadas através da geração do IDEA-C2-KG, como veremos adiante.

```
PREFIX cnt: <http://idea-c2.org/data/#>
CONSTRUCT {?s1 ?p1 ?o1. ?s2 ?p2 ?o2. ?s3 ?p3 ?o3}
WHERE {
    {SELECT * WHERE {?s1 ?p1 ?o1. FILTER(?s1=cnt:acampamento)}}
    {SELECT * WHERE {?s2 ?p2 ?o2. FILTER(?o2=cnt:estacionamento)}}
    {SELECT * WHERE {?s3 ?p3 ?o3. FILTER(?s3=cnt:tropa)}}}
```

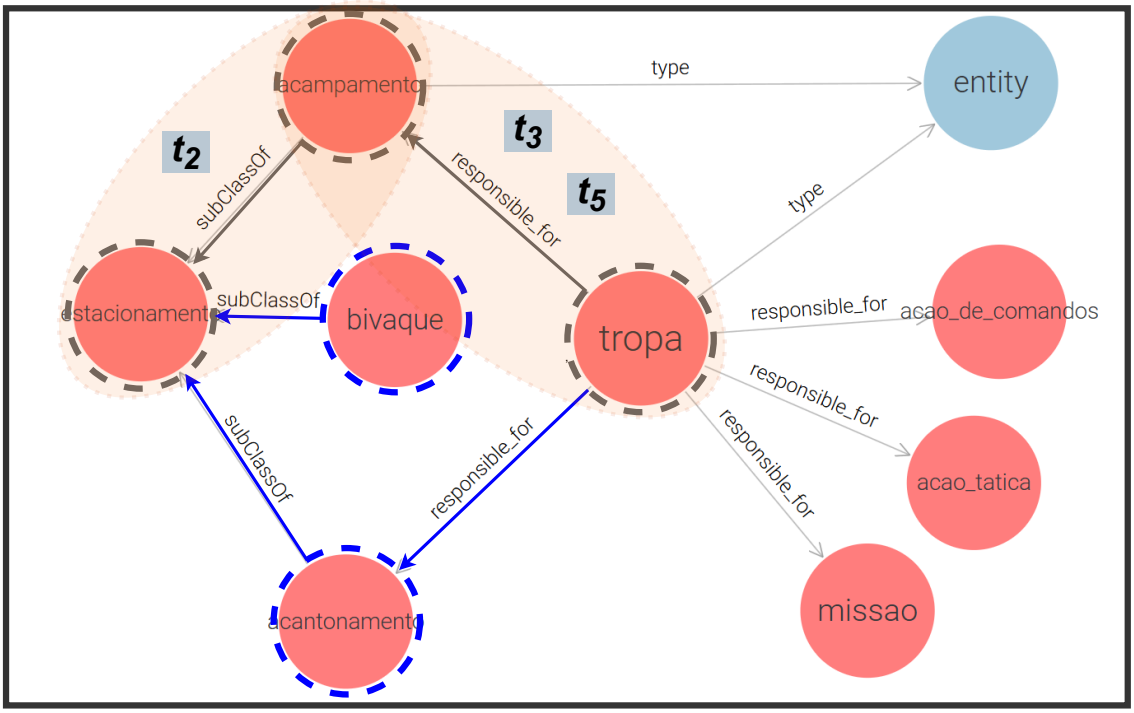


Figura 24 – Extrato do IDEA-C2-KG baseado nas inferências do IDEA-C2-LM.

Na Figura 24, é ilustrado um extrato do grafo IDEA-C2-KG através de uma consulta SPARQL. Esse grafo foi gerado a partir de textos doutrinários, orientado pelo C2RM com base no resultado das entidades reconhecidas após a submissão de s_1 e s_2 ao IDEA-C2-LM. Para enriquecer o IDEA-C2-KG, durante a geração do grafo somente são inseridos novos recursos. Note que são destacadas em vermelho as triplas, t_2, t_3 e t_5, confirmando que os recursos de dados espelham as entidades reconhecidas e as relações extraídas do IDEA-C2-LM. Contudo, como o IDEA-C2-KG é formado após a curadoria da anotação e enriquecido através das interações com IDEA-C2-LM, outros recursos estão disponíveis para análises. Por exemplo, os nós **bivaque** e **acantonamento** são relacionados ao nó **estacionamento** através da relação *rdfs:subClassOf*. Logo, é possível identificar que além de **acampamento**, também há outros tipos de **estacionamento**, como **acantonamento** e **bivaque**. Porém, somente **acantonamento** e **acampamento** estão associados a uma **tropa** através da relação *responsible_for*. Além disso, no canto direito, é possível visualizar alguns outros nós associados à **tropa**, como **missao**, **acao_tatica** e **acao_de_comandos**.

Portanto, de acordo com as evidências no experimento, a geração do IDEA-C2-LM

atende à proposição da hipótese H3, demonstrando que ele foi concebido com o apoio do metamodelo C2RM combinado com uma pré-anotação heurística e RS aplicado em um corpus. Cabe ressaltar que a pré-anotação foi abordada em detalhes no experimento Ex_2. Neste experimento, o foco foi explorar a geração do IDEA-C2-LM através de um processo iterativo com refinamentos sucessivos a fim de alcançar o melhor desempenho do ML ajustado. Além disso, notamos que as evidências com a geração do IDEA-C2-KG atendem à proposição da hipótese H1, demonstrando que é possível gerar um KG a partir de textos doutrinários, orientado por um metamodelo e apoiado por um ML e ajustado ao contexto militar. Os resultados alcançados na geração do IDEA-C2-KG são promissores, permitindo que o grafo seja enriquecido através das interações com o IDEA-C2-LM. Ademais, como abordado, as evidências com a exploração do KG favorecem a obtenção do conhecimento, permitindo que o especialista do domínio aprofunde ou complemente um conhecimento já adquirido no IDEA-C2-LM.

6.4 Ex_4: Avaliação do IDEA-C2-DM (DM^D) com o fragmento da CROMO-MOS (Operação militar ofensiva)

O experimento Ex_4 tem como objetivo validar a hipótese H4 que propõe o seguinte: "É possível apoiar a construção de um DM no contexto militar a partir da submissão de textos a um ML ajustado combinando com a exploração dos dados em um KG." Para alcançar esse objetivo o experimento foi estruturado em três etapas descritas a seguir.

Na primeira etapa, foi construído um DM manualmente, do tipo *gold standard*, denominado DM^1. Esse DM foi elaborado por um especialista do domínio a partir de um conjunto de textos, $CT = \{st_1, st_2, ...st_m\}$, como exemplo um minimundo, e um conjunto listado de Questões de Competência (QC) associadas, $QC = \{qc_1, qc_2...qc_n\}$, que explora objetos, atores e ações baseadas em doutrinas no domínio militar brasileiro. Esta etapa será apresentada na subseção 6.4.1. Na segunda etapa, um outro especialista do domínio, obtém a mesma lista de QC e CT da primeira etapa e com o apoio do IDEA-C2-Tool, ele constrói um DM, denominado por DM^D (IDEA-C2-DM). Esta etapa será apresentada na subseção 6.4.2. Por fim, na terceira etapa, apresentada na subseção 6.4.3, descrevemos em três estágios, a avaliação do IDEA-C2-DM, quantificando os acertos do DM^D em comparação com o DM^1, observando classes, relações e seus tipos correspondentes.

6.4.1 Etapa 1: Obtenção do minimundo, QC e DM^1

No Quadro 13, é apresentado um conjunto de textos, representado por CT, que explora um minimundo no domínio militar, originalmente discutido no trabalho de Silva(3). Esse minimundo descreve o contexto de uma Operação militar baseado em DML (174, 173), em especial as Operações ofensivas, descrevendo as suas características e aplicações com o

foco na troca de informações entre os subordinados. Na parte inferior, são listadas três QC, representadas pelo conjunto QC, as quais explicitam perguntas sobre o universo das Operações ofensivas que o Modelo de Domínio (DM) deve ser capaz de responder.

Quadro 13 – Minimundo do cenário de Operação Ofensiva. Adaptado de Silva(3).

Minimundo	
Uma Operação Militar (Military OPeration) possui diferentes subtipos, como Operações Básicas e Complementares (Basic and Complentary Operations). As Operações básicas são operações que, por si só, podem atingir os objetivos determinados por uma autoridade em uma situação de guerra ou não-guerra. As operações complementares, por outro lado, destinam-se a ampliar, melhorar e/ou complementar as operações básicas. Uma Operação Ofensiva (Offensive Operation) é uma das Operações Básicas que se inicia e se desdobra em quatro operações distintas: (i) Reunião de Preparação (Assembly Area), quando o comandante se reúne com seus subordinados para trocar informações e transmitir ordens; (ii) Marcha para o Combate (Movement to Contact), quando as forças participantes marcham em direção às forças inimigas; (iii) Ataque Coordenado (Organized Offensive), momento em que ocorre a ação de atacar as forças inimigas; e (iv) Aproveitamento do êxito e Perseguição (Exploitation and Follow Up), uma vez que o ataque foi bem sucedido, ações para consolidar a vitória e perseguir o inimigo poderão ser realizadas. Alguns participantes podem executar uma operação de Controle de Tráfego (Traffic Control) ou de Reconhecimento (Initial Reconnaissance) durante uma operação de Marcha para o Combate. O rádio cognitivo pode perceber essas informações táticas via canal de comunicação com os sistemas C2.	
Questões de Competência	
qc_1	No contexto tático, o que é uma operação ofensiva e como ela é constituída?
qc_2	Que elementos operacionais executam ou participam da Operação Ofensiva?
qc_3	Qual é a ordem temporal em que operações que fazem parte da Operação Ofensiva acontecem?

Do texto extraído de Silva(3), utilizou-se para o experimento o modelo de domínio bem fundamentado, denominado CROMO-MOS. Esse DM foi construído por um especialista de domínio com base nos construtos da *Unified Foundational Ontology (UFO)*, como ilustrado na Figura 25 e responde às questões do conjunto QC descritas no Quadro 13. Ao analisar a CROMO-MOS identificam-se as principais entidades ou classes do domínio (e.g. *Military operation, Basic Operation, Offensive Operation*, dentre outras.) e os relacionamentos entre essas classes representados por hierarquias, agregações e associações simples. Por exemplo, em qc_1, questiona-se o que é uma operação ofensiva e como ela é constituída. Ao avaliar as classes expressas na CROMO-MOS, é possível inferir que uma Operação Ofensiva é um subtipo de Operação Básica, que se desdobra em quatro operações distintas: (i) Reunião de Preparação (*Assembly Area*); (ii) Marcha para o Combate (*Movement to Contact*); (iii) Ataque Coordenado (*Organized Offensive*); e (iv) Aproveitamento do êxito e Perseguição (*Exploitation and Follow Up*). Nesse sentido, independente do compromisso ontológico assumido, conforme descrito em Silva(3), com a CROMO-MOS é possível responder qc_1 a qc_3.

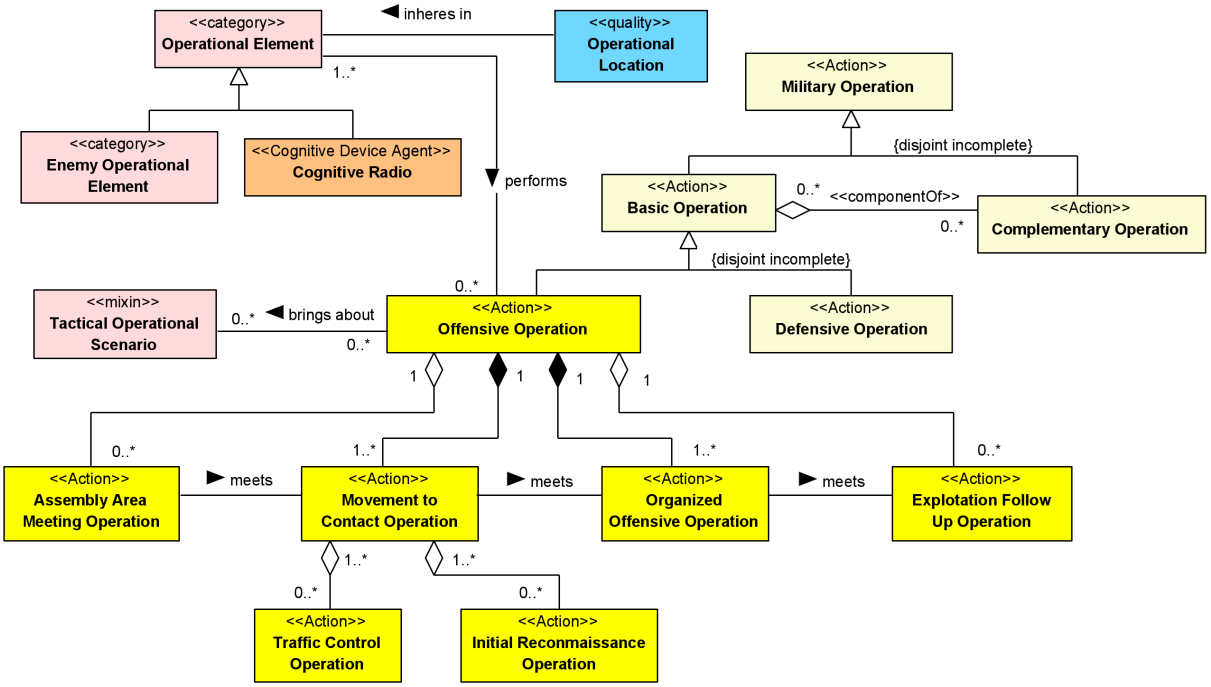


Figura 25 – DM CROMO-MOS: Operação Ofensiva. Adaptado de Silva(3).

Os esforços empreendidos pelo especialista do domínio para construir manualmente a CROMO-MOS envolveram atividades para entender o negócio, investigar pelo menos duas doutrinas volumosas (173, 174) e realizar um levantamento de requisitos. Comumente, um levantamento envolve atores do domínio que podem possuir visões distintas do cenário explorado. O esforço empreendido para elaborar um DM de modo tradicional costuma ser custoso além de demandar tempo para amadurecimento e entendimento do negócio e, muitas vezes contar com a disponibilidade dos usuários-chaves. Nesse aspecto, é justamente onde IDEA-C2 pode se inserir e tentar contribuir, como veremos na próxima subseção.

6.4.2 Etapa 2: Modelagem conceitual com o suporte do IDEA-C2

Segundo Elmasri e Navathe(63), para construir um Modelo de Domínio (DM), inicialmente, o especialista identifica os objetos do mundo real (e.g. Operação Militar) que possui existência própria e indivisível (e.g. o nome da operação). Em seguida, o especialista analisa a relevância do objeto no contexto do domínio, bem como busca características semânticas através dos relacionamentos entre esses objetos (e.g. Operação ofensiva é um tipo de Operação militar). Além disso, ele busca elucidar as Questões de Competência (QC) que o DM deve responder (175). Por fim, o especialista reúne todas as informações e elabora graficamente o DM.

No contexto do IDEA-C2, a identificação e a relação dos objetos do domínio é realizada através da interação do especialista com IDEA-C2-Tool. Como abordado, há duas formas de interação. Na primeira, o especialista interage diretamente com o IDEA-

C2-LM através dos textos do domínio. Por sua vez, na segunda, o especialista interage com IDEA-C2-KG com o objetivo de refinar as informações do domínio, enriquecendo seu conhecimento, assim como a construção do seu DM. Cabe ressaltar que não há ordem pré-estabelecida, i.e., o especialista pode interagir de modo independente, buscando as informações que melhor desejar.

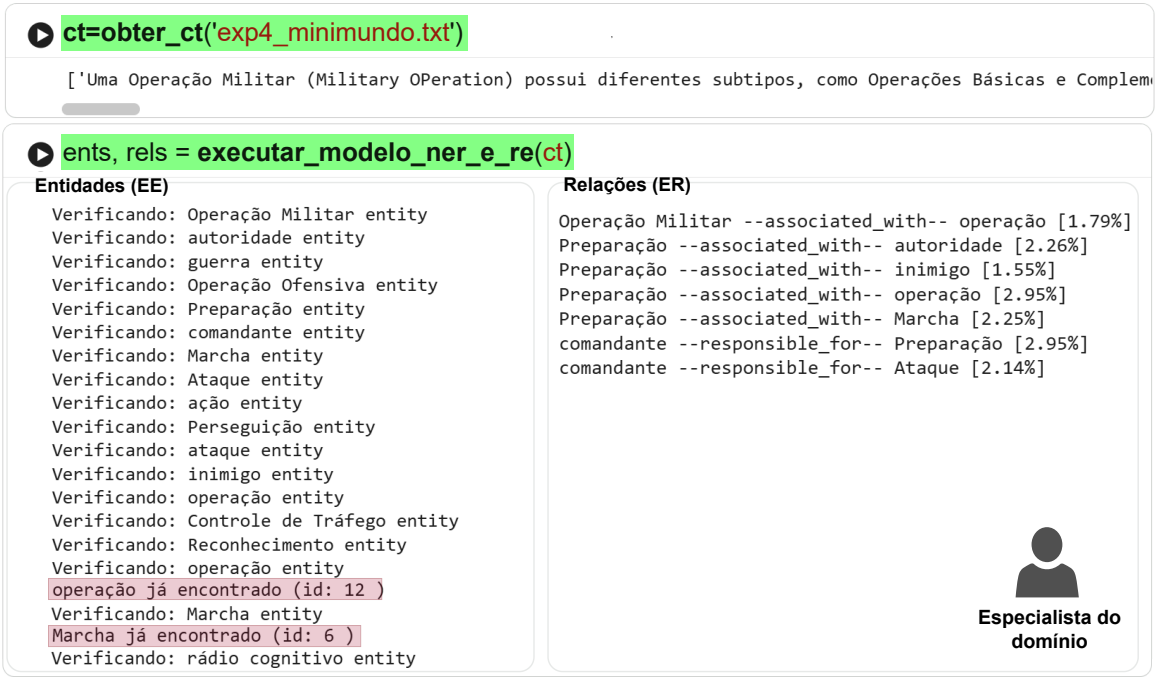


Figura 26 – IDEA-C2-Tool: Obter e Executar IDEA-C2-LM (Ex_4). Imagem do autor.

Nesse sentido, para realizar o experimento, de posse do minimundo (CT) e das questões de competência correspondentes (QC), o especialista do domínio faz uso do protótipo IDEA-C2-Tool (Figura 26) e submete CT através do procedimento ***obter_ct()*** do caderno IDEA-ETAPA 5-RodaModeloNEReRE - Experimento 4[12] a fim de interagir com IDEA-C2-LM. Posteriormente, o especialista executa a rotina ***executar_modelo_ner_e_re(*** CT ***)***, passando como parâmetro CT. Cabe esclarecer que no IDEA-C2-Tool, o conteúdo processado nesta rotina é apresentado tanto em formato de lista (Figura 26) quanto de grafo (Figura 27) para facilitar a visualização das entidades e relações identificadas. Contudo, esse grafo ainda não representa os recursos armazenados no IDEA-C2-KG.

Ao processar o texto submetido, o IDEA-C2-Tool reconheceu dezesseis entidades nomeadas (e.g. Operação militar, comandante, etc.), apresentadas na lista **Entidades (EE)** da Figura 26, as quais foram armazenadas no conjunto EE. Além disso, foram extraídas sete relações entre as entidades identificadas (e.g. comandante – *responsible_for* – ataque), apresentadas na lista **Relações (ER)**, as quais foram armazenadas no conjunto

[12] <https://github.com/comp-ime-eb-br/S2C2-IME/tree/main/deliverables/idea-c2/experimentos/exp4>

ER. Ambos os conjuntos de dados são candidatos potenciais de classes e relacionamentos a serem representados no DM, cabendo ao especialista do domínio avaliar a pertinência no contexto do negócio. Como mencionado, no IDEA-C2-Tool, a avaliação desses conjuntos de dados também pode ser realizada através de uma visualização em formato de grafo, onde os nós representam as entidades identificadas e as arestas são as relações extraídas, como será apresentada a seguir.

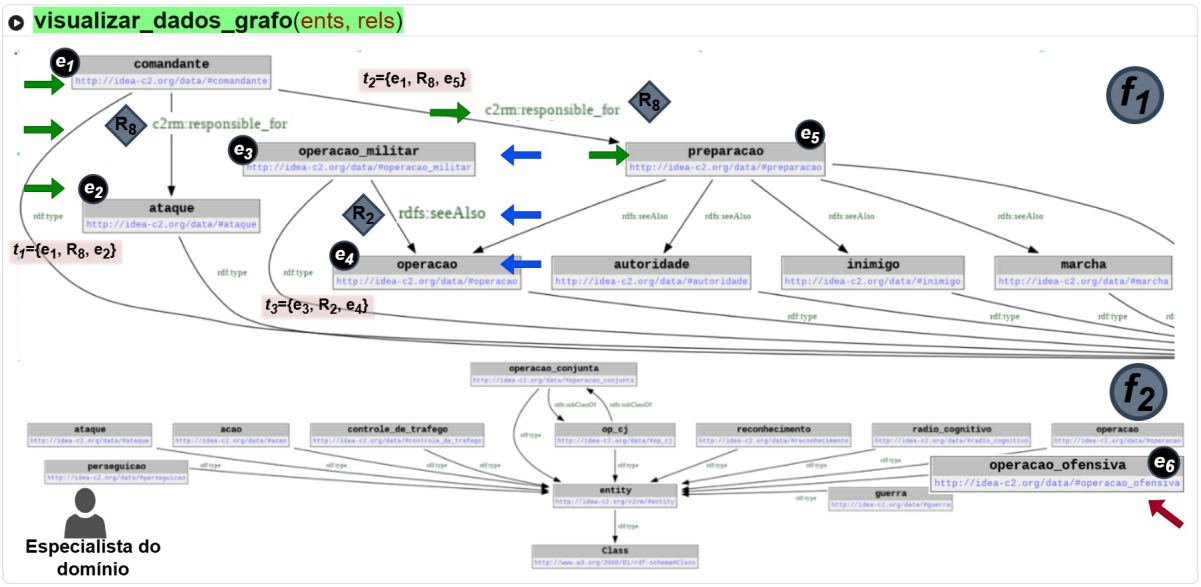


Figura 27 – Visualização em forma de grafo da interação com IDEA-C2-LM. Imagem do autor.

Na Figura 27, após o especialista executar a rotina ***visualizar_dados_grafo(ents, rels)***, os dados dos conjuntos, *EE* e *ER*, são transformados em formato de grafo, conforme ilustração. Esse tipo de recurso permite ao especialista navegar entre nós e arestas a fim de identificar potenciais classes e relações candidatas que no formato em lista são difíceis de visualizar. Note que dado o tamanho do grafo, foi necessário dividir em dois *frames* ($f1$ e $f2$). Ao analisar o grafo, o especialista identificou que os nós e as arestas são de fato elementos do domínio militar (e.g. comandante, operação militar, inimigo, etc.), indicando aspectos semânticos inerentes ao contexto da operação ofensiva. Entretanto, como IDEA-C2-LM é baseada na abordagem DD, é necessário fazer uma análise aprofundada para determinar o quanto o artefato pode apoiar o especialista do domínio. Dessa forma, para facilitar a análise, destacamos na Figura 27, as três triplas (t_1, t_2 e t_3) usadas para exemplificar as possíveis análises que o especialista do domínio poderia realizar.

Nas triplas, t_1 e t_2, destacadas em setas verdes, na Figura 27, por exemplo, o especialista identificou que ambas as triplas expressam que o nó "comandante", e_1, é associado através da propriedade (*responsible_for*) aos nós "ataque", e_2, e "preparação", e_5. Além disso, o especialista identificou que o nó, e_5, é associado ao nó "operacao", e_4, através da propriedade *rdfs:seeAlso*. Assim, o especialista ao analisar semanticamente ambas

as triplas, t_1 e t_2, presumiu que IDEA-C2-LM inferiu corretamente o contexto aplicado. Na mesma linha de raciocínio, o especialista ao analisar a tripla, t_3, identificou que os nós "operacao_militar", e_3, e "operacao", e_4, destacados em setas azuis, são associados através da propriedade *rdfs:seeAlso*. Assim, ele pôde deduzir que essas triplas são relacionadas ao domínio militar e expressam as características do minimundo (CT) apresentado. Apesar disso, como há relações entre os pares de nós, (e_3, e_4) e (e_4, e_5), através da propriedade *rdfs:seeAlso*[13], esses nós podem suscitar diversas interpretações, como veremos a seguir.

Por exemplo, na tripla t_3, o especialista pode identificar o nó ***operacao***, e_4, como uma entidade mais genérica e ***operacao_militar***, e_5, como uma entidade especializada, expressando, assim, uma hierarquia. Por outro lado, o especialista pode também identificá-las como termos sinônimos ou até mesmo agregados. Independente do que o especialista defina, a relação *rdfs:seeAlso* não é adequada no domínio analisado, pois existem aspectos semânticos que podem qualificar melhor a relação entre os nós da tripla t_3. Na mesma linha de raciocínio, o especialista ao analisar o nó **operacao_ofensiva**, e_6, ligado ao conjunto QC em questão, pode deduzir que IDEA-C2-LM conseguiu até identificá-lo no contexto. Porém, o IDEA-C2-LM não retornou as relações com os outros nós. Dessa forma, esses dois últimos casos analisados merecem uma investigação aprofundada no IDEA-C2-KG dadas as lacunas de conhecimento que o ML não conseguiu explicitar.

Um complemento proposto pela abordagem IDEA-C2 para aprimorar a elaboração do DM, é investigar as lacunas de conhecimento do IDEA-C2-LM por meio dos nós e arestas explorados no IDEA-C2-KG. Para tanto, o especialista no domínio elabora consultas ao IDEA-C2-KG para identificar classes do modelo de domínio de acordo com as QCs. Primeiro, as QCs gerais podem ser traduzidas em consultas exploratórias escritas em SPARQL, tais como "o que é aplicado a quê", e podem ser enviadas ao KG. Essas consultas retornam um conjunto de entidades e relações que são candidatas potenciais para representar classes e relacionamentos no modelo de domínio em construção, como apresentado na seção 4.3, mais especificamente no subprocesso **Realizar modelagem conceitual**, como veremos adiante.

No fragmento do IDEA-C2-KG, ilustrado na Figura 28, por exemplo, o especialista executou uma consulta que retorna os nós diretamente relacionados ao nó **operacao_ofensiva**, e_1, destacado em pontilhado preto. Ao analisar os nós relacionados ao e_1, o especialista considerou-o como uma classe do modelo de domínio em construção. Assim, outras classes potenciais associadas ao e_1 também foram identificadas, sugerindo que novas análises podem ser realizadas para responder as questões do conjunto QC. Por exemplo, os nós, de e_2 a e_5, destacados em pontilhados azuis, são relacionados através da propriedade *rdfs:subClassOf*. Como essa propriedade expressa uma hierarquia do tipo *General Domain*, esses nós podem ser mapeados como classes candidatas (c_2 a c_5) de acordo com a sua

[13] A propriedade *rdfs:seeAlso* expressa baixo valor semântico.

```
PREFIX cnt: <http://idea-c2.org/data/#>
CONSTRUCT {?s1 ?p1 ?o1. ?s2 ?p2 ?o2.}
WHERE
{{SELECT * WHERE {?s1 ?p1 ?o1. FILTER(?s1=cnt:operacao_ofensiva)}}
    {SELECT * WHERE {?s2 ?p2 ?o2. FILTER(?s2=cnt:operacao_ofensiva)}}
    FILTER(?o1=?o2)}
```

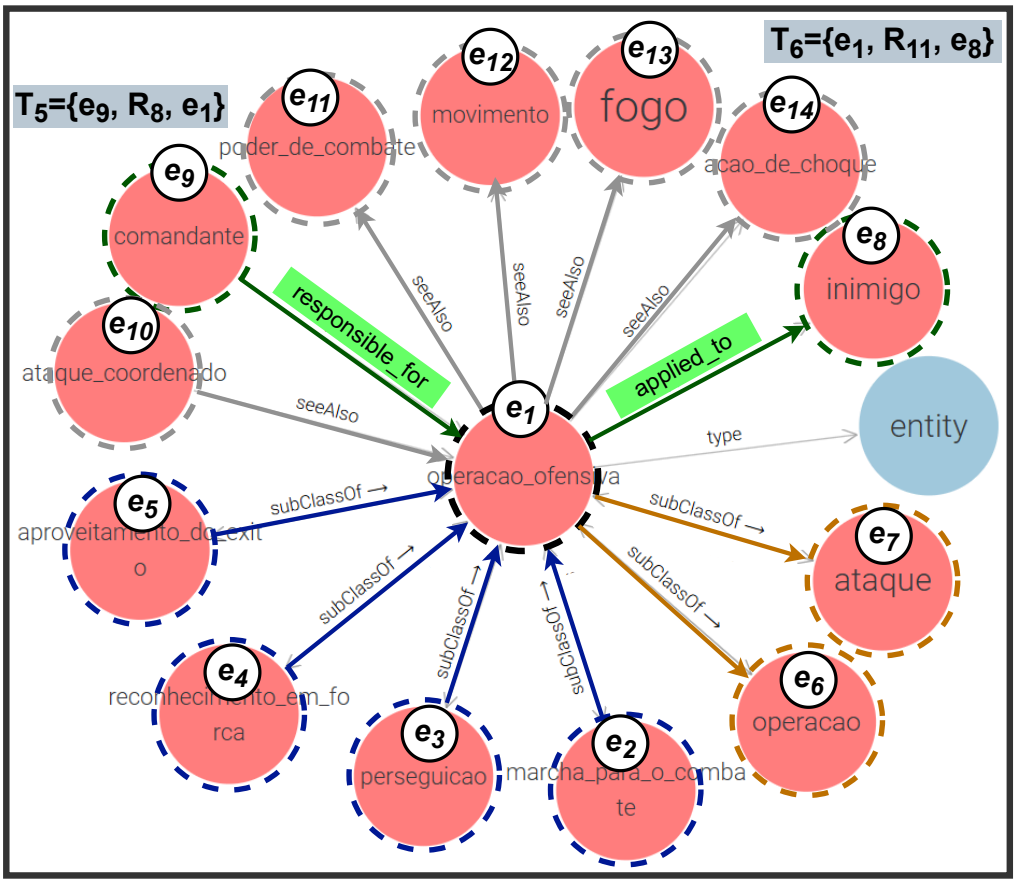


Figura 28 – Consulta sobre o nó **operacao_ofensiva**. Imagem do autor.

semântica, assim como e_1 pode ser mapeada em c_1 no DM, como ilustrado na Figura 29.

De modo análogo, ainda na Figura 28, o especialista ao analisar os nós, e_6 e e_7, pontilhados em laranja, identificou que eles também são relacionados a e_1 através da propriedade *rdfs:subClassOf*. Porém, essas relações são expressas em direção contrária em função de e_1 ser subclasse de e_6 e e_7. Além disso, ele identificou que há duas relações semânticas de e_1 com os nós, e_8 e e_9, pontilhados em verde, diretamente do domínio militar ou *C2 Domain* que merecem ser exploradas. Ao analisar a relação *c2rm:responsible_for*(e_9, e_1), o especialista identificou que o **comandante** é o responsável pela **operação ofensiva**. Por sua vez, ao analisar a relação *c2rm:applied_to*(e_1, e_8), ele identificou que uma **operação ofensiva** é aplicada sobre um **inimigo**. Com base nessas análises, o especialista identificou possíveis nós e relações no IDEA-C2-KG que foram mapeados como classes e relações

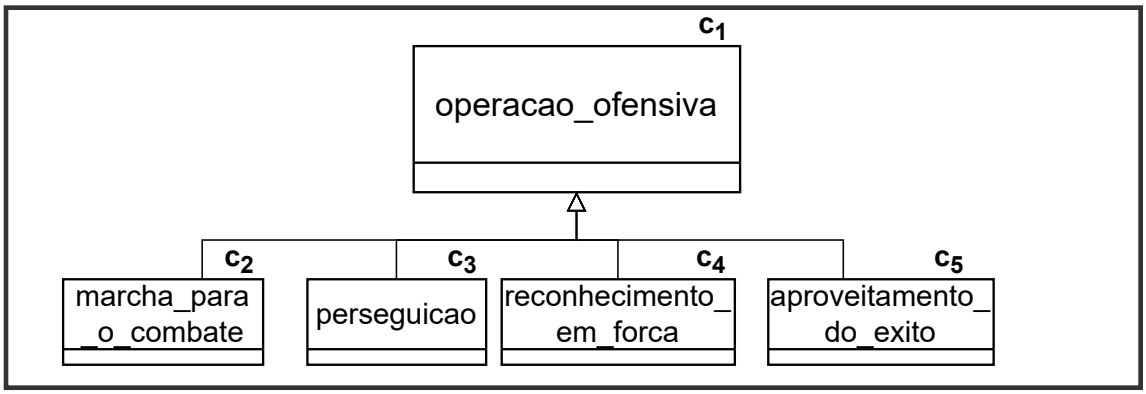


Figura 29 – Construção do DM - Hierarquia de Operação ofensiva (e_1). Imagem do autor.

correspondentes no DM em construção, representadas por c_1, c_6, c_7, c_8 e c_9, como ilustrado na Figura 30. Cabe observar que existem relações de e_1 com os nós de e_{10} a e_{14}, através da propriedade *rdfs:seeAlso*, que merecem ser investigadas por meio de outras consultas ao IDEA-C2-KG, como veremos a seguir.

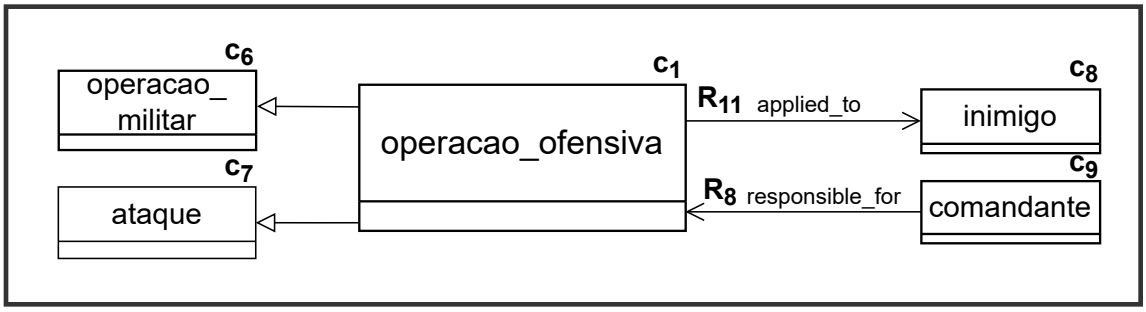


Figura 30 – Construção do DM - Classes e relacionamentos com **operacao_ofensiva** (e_1). Imagem do autor.

No *Frame*[1] da Figura 31, é ilustrada à esquerda a consulta elaborada pelo especialista com o objetivo de investigar as relações adjacentes em comum entre os nós, e_1 e e_{14}, que podem enriquecer a construção do DM. Ao analisar o grafo resultado desta consulta, à direita da ilustração, o especialista observou que os nós, e_8 e e_{10}, destacados em pontilhados verdes, possuem relações comuns aos nós e_1 e e_{14}. Assim, o especialista pôde inferir que os nós, e_1 e e_{14}, possuem algumas semelhanças, apesar de não estarem relacionados diretamente. Dessa forma, a partir dessa inferência o especialista pode aprofundar seu conhecimento através de novas consultas ao IDEA-C2-KG. Por exemplo, ao investigar os nós e_1 e e_{10}, relacionados através da propriedade *rdfs:seeAlso*, o especialista suspeita que pode haver outros nós e relações indiretamente relacionados, como veremos a seguir.

No *Frame*[2] da Figura 31, o especialista elaborou outra consulta ao IDEA-C2-KG com o objetivo de investigar a relação *rdfs:seeAlso*(e_1, e_{10}) através dos nós, e_1 e e_{10},

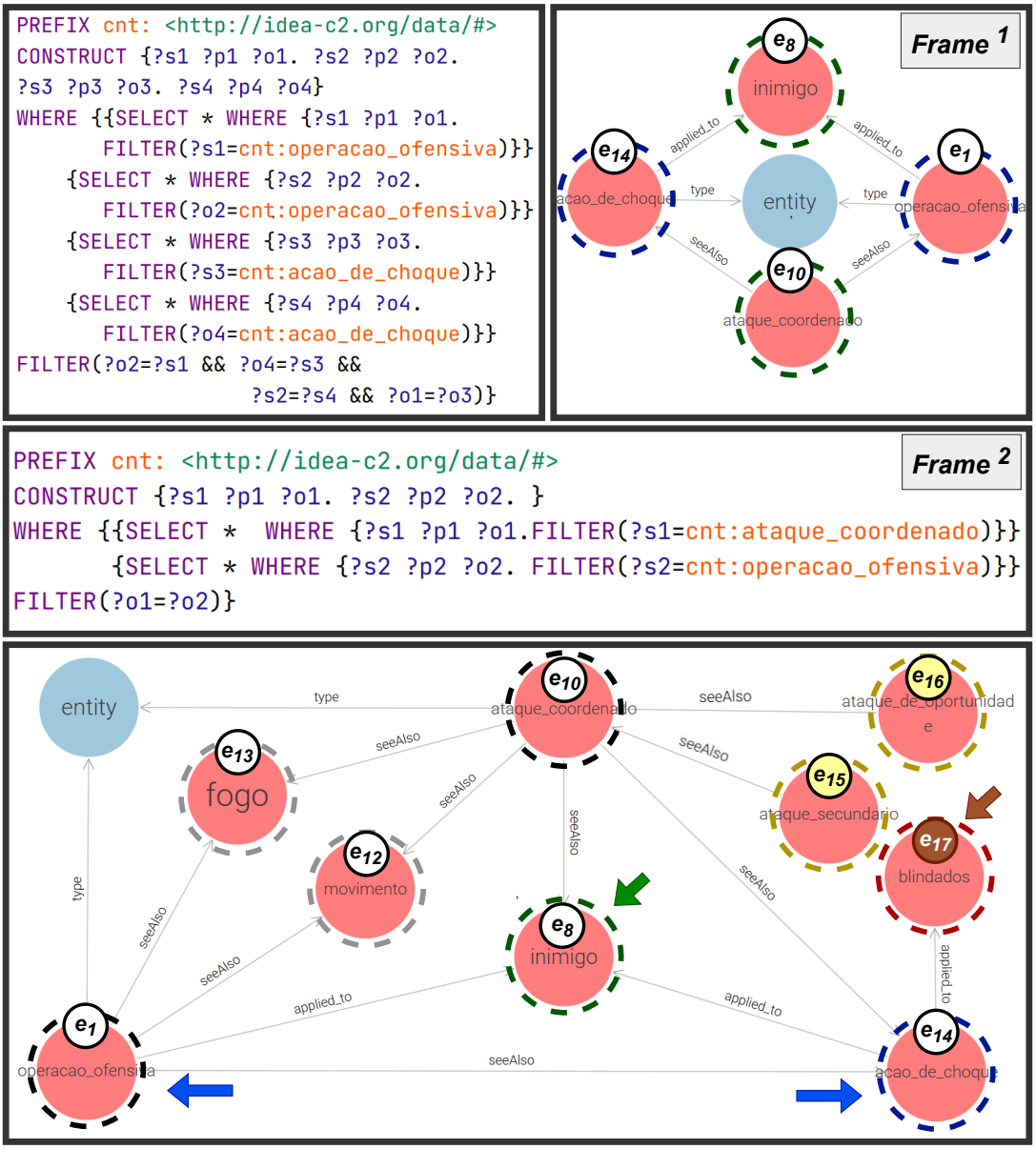


Figura 31 – Consulta sobre os nós **operacao_ofensiva** e **acao_de_choque**. Imagem do autor.

destacados em pontilhado preto, cujas relações adjacentes podem enriquecer a construção do DM. Inicialmente, o especialista identificou que ambos os nós, e_{12} e e_{13}, pontilhados em cinza, são relacionados aos nós, e_1 e e_{10}. Em seguida, ele identificou que os nós, e_{15} e e_{16}, são relacionados ao nó e_{10}, indicando que há dois "ataques" que podem também ser associados contra o **inimigo**, e_8. Porém, como todos esses nós analisados são relacionados através da propriedade *rdfs:seeAlso*, merecem uma investigação aprofundada. Por outro lado, o especialista ao analisar o nó **blindados**, e_{17}, destacado em pontilhado vermelho, identificou-o como uma potencial classe, c_{17}, a ser mapeada no DM, tendo em vista a sua relação com o nó e_{14} através da propriedade *applied_to*.

Portanto, a investigação de nós e relações no IDEA-C2-KG é realizada iterativamente à medida em que novos nós são identificados. As relações com maior valor semântico (e.g. *c2rm:responsible_for* e *rdfs:subClassOf*) facilitam a identificação e o mapeamento de classes candidatas no DM. Como apresentado, o especialista através das consultas e análises realizadas no IDEA-C2-KG, identificou o total de 18 entidades que foram armazenadas no conjunto EE', bem como 20 triplas, contendo o par de entidades e suas relações, as quais foram extraídas e armazenadas no conjunto ER'. Os conjuntos EE' e ER' foram utilizados no mapeamento das classes do DM, como veremos a seguir.

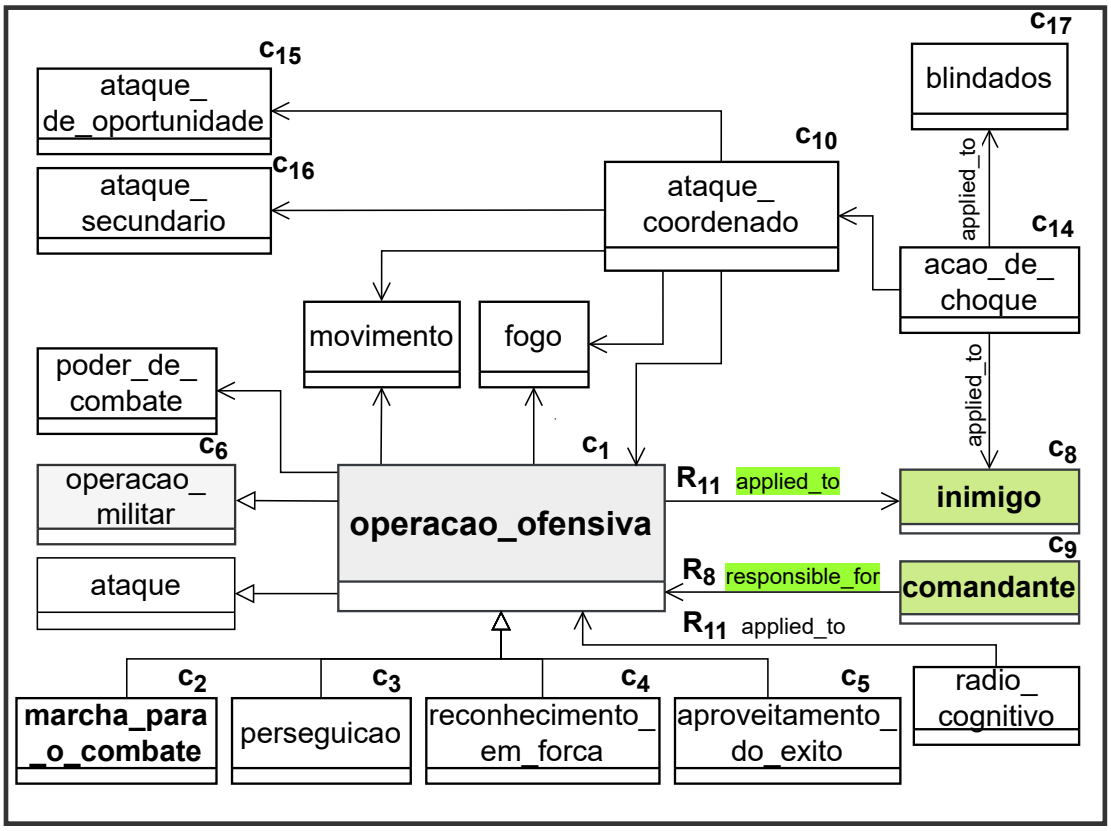


Figura 32 – IDEA-C2-DM (DM^D) baseado no minimundo (CT). Imagem do autor.

Com base nos artefatos IDEA-C2-LM e IDEA-C2-KG, o especialista no domínio construiu o IDEA-C2-DM, denominado como DM^D, ilustrado na Figura 32. Ao longo da exploração do KG, as entidades dos conjuntos EE e EE' foram mapeadas para as classes (c_n) no DM^D, assim como as suas relações, ER e ER', são mapeadas em relacionamentos. Por exemplo, a partir do conjunto EE, ilustrado na Figura 27, as entidades e_1 e e_6 foram mapeadas para as classes do modelo de domínio c_1 e c_6, destacadas em cinza. Da mesma forma que os nós, e_8 e e_9, ilustrados na Figura 28, foram mapeados para as classes, c_8 e c_9, destacadas em verde. Observe que algumas entidades do KG são deixadas de fora do mapeamento, por exemplo, **guerra**, uma vez que parece não estar conectada a nenhuma outra entidade. Cabe ressaltar que, embora o especialista conte com o suporte do KG, ele

é livre para tomar as decisões de design ao conceber seu modelo de domínio.

Nesta subseção, foram apresentadas as interações do especialista do domínio com os artefatos IDEA-C2-LM e IDEA-C2-KG como suporte à elaboração do IDEA-C2-DM, DM^D. Na próxima subseção, DM^D será avaliado a partir da comparação de seus objetos com a CROMO-MOS, DM^1.

6.4.3 Etapa 3: Avaliação do IDEA-C2-DM (DM^D vs. DM^1)

Neste experimento, como foi mencionado, dois especialistas foram envolvidos. Um deles apresentou um DM construído a partir de um minimundo, utilizando essencialmente a abordagem TD, denominado DM^1. Por sua vez, outro especialista utilizou a abordagem IDEA-C2, combinando as abordagens DD e TD, para construir um DM, denominado DM^D.

Consideramos o DM^1 como *gold standard* por ter sido construído com base em um estudo de longa duração, exigindo mais tempo e envolvendo mais fontes. O DM^D foi gerado por outro especialista que seguiu a abordagem IDEA-C2. Embora tenhamos considerado DM^1 como o *gold standard*, é possível argumentar que é controverso considerar uma modelagem conceitual como a verdade absoluta, pois ela está sujeita a variabilidade de conceitos dentro do domínio e a diferentes visões sobre um mesmo fato (104). No entanto, o objetivo aqui não é descobrir qual a melhor modelagem, mas é avaliar o quão coincidente foram as modelagens, i.e., se ao usar a abordagem IDEA-C2 foi possível contemplar os elementos da modelagem DM^1. Ao término, na Tabela 5, são demonstrados os resultados da comparação dos elementos do DM^D em relação ao DM^1.

Para avaliar o experimento, foram definidos três estágios que englobam desde a obtenção dos DMs, incluindo a definição dos objetos de comparação e os resultados alcançados. Inicialmente, quantificamos os acertos do DM^D em comparação com o DM^1, observando classes, relacionamento e seus tipos correspondentes. Para realizar a comparação, consideramos cada DM como uma tupla, $DM = (DM_C, DM_R)$, onde DM_C é o conjunto de classes de DM e DM_R é o conjunto de seus relacionamentos. Ambos os conjuntos são definidos da seguinte forma:

- $DM_C^D = \{c_1, ..., c_n\}$
- $DM_R^D = \{(c_i, R_j, c_k) \mid c_i, c_k \in DM_C^D\}$
- $DM_C^1 = \{c_1^1, ..., c_m^1\}$
- $DM_R^1 = \{(c_p^1, R_q^1, c_r^1) \mid c_p^1, c_r^1 \in DM_C^1\}$

No primeiro estágio, os conjuntos de classes, DM_C^D e DM_C^1, são comparados, par a par, i.e., cada par $(c_i, c_p^1) \in (DM_C^D \times DM_C^1)$ é analisado. Quando ambas as classes de um

par representam o mesmo conceito, ele é quantificado como um *verdadeiro positivo* (TP), como exemplo a classe **operacao_ofensiva**. Assim, o total de ocorrências verdadeiras positivas é dado por $TP = |(DM_C^D \cap DM_C^1)|$[14]. Quando existem classes no DM^1 que não estão no DM^D, por exemplo, a classe **Operational Element**, que indica um falso negativo (FN). O total de falsos negativos é dado por $FN = |(DM_C^1 - DM_C^D)|$. Da mesma forma, que o oposto ao FN pode ocorrer. Por exemplo, a classe **ataque** que está em DM_C^D e não está em DM_C^1, o que indica um falso positivo (FP). Então, o total de falsos positivos é dado por $FP = |(DM_C^D - DM_C^1)|$. Além disso, TP, FP e FN são usados para calcular as métricas da comparação, apresentadas na Tabela 5.

No segundo estágio, os conjuntos de relações, DM_R^D e DM_R^1, são comparados da seguinte maneira. Inicialmente, os pares de classes relacionados são comparados, desconsiderando seu tipo de relação. Cada par $((c_i, c_k), (c_p^1, c_r^1))$ é analisado, onde $(c_i, R_j, c_k) \in DM_R^D$ e $(c_p^1, R_q^1, c_r^1) \in DM_R^1$, assumindo que os pares $(c_i, c_p^1), (c_k, c_r^1)$ foram previamente classificados como verdadeiros positivos. Por exemplo, o fato de o par de classes, **marcha_combate** e **operacao_ofensiva**, estar relacionado em ambos os DMs indica um TP. Além disso, FN e FP são dados por $FN = |(DM_R^1 - DM_R^D)|$ e $FP = |(DM_R^D - DM_R^1)|$, respectivamente. Um exemplo de FN envolve o par de classes, **radio_cognitivo** e **operational_element**, que aparece apenas em DM^1. Isso corresponde a uma hierarquia entre as classes, onde **radio_cognitivo** é subclasse de **operational_element**. Em contrapartida, um exemplo de FP ocorre entre a classe **operacao_ofensiva**, identificada tanto em DM^D quanto em DM^1, e **fogo** que aparece apenas em DM^D.

Finalmente, no terceiro estágio, cada par dessas classes relacionadas é analisado novamente com o foco no tipo da relação, ou seja, (R_j, R_q^1). Por exemplo, as classes **marcha_combate** e **operacao_ofensiva** estão relacionadas entre si por meio de diferentes tipos de relação, ou seja, uma composição é usada em DM_R^1, diferentemente da hierarquia que é usada em DM_R^D. Nesse caso, é considerada uma ocorrência de FP. Por outro lado, se as relações forem do mesmo tipo, isso indica uma ocorrência de TP. Por fim, uma ocorrência de FN é contada quando uma relação é modelada em DM_R^1 e não há relação correspondente em DM_R^D.

Tabela 5 – Comparação entre IDEA-C2-DM (DM^D) e DM^1

Conjuntos	**TP**	**FP**	**FN**	**PR**	**RC**	**F1**
$DM_C^D \times DM_C^1$ **(classes)**	7	9	11	44%	39%	41%
$DM_R^D \times DM_R^1$ **(classes relacionadas)**	3	13	17	10%	15%	12%
$DM_R^D \times DM_R^1$ **(tipos de relacionamentos)**	0	16	41	0%	0%	0%

[14] Operações de conjunto como $\cap$ e $-$ são aplicadas aqui no significado semântico dos elementos do conjunto.

A Tabela 5 mostra os resultados da comparação entre DM^D e DM^1. Ao analisar os números, pode-se deduzir que o IDEA-C2 alcançou resultados promissores ao combinar as abordagens DD e TD para desenvolver um modelo de domínio. Considerando a comparação envolvendo apenas classes individualmente, a taxa de precisão alcançou 44% com uma cobertura de 39%. Em contrapartida, ao considerar os pares de classes, as taxas reduziram, alcançando 10% de precisão e 15% de cobertura, indicando que ao envolver as relações na avaliação do experimento, as diferentes visões de um mesmo fato influenciam a semântica expressa na relação. Além disso, os resultados da comparação envolvendo as relações (pares de classes com seus tipos de relação) que não tiveram nenhuma concordância, em virtude de não haver TP entre ambos os DMs. Presumimos que um dos indícios desse resultado está nos desafios relacionados às atividades de anotação e curadoria dos textos. Essas atividades são necessárias para ajustar o ML não apenas ao domínio específico, mas também a diferentes contextos de forma mais ampla. Isso afeta negativamente a geração do KG e faz com que o especialista perca tempo buscando recursos mais expressivos no IDEA-C2-KG, principalmente os nós relacionados através da propriedade *rdfs:seeAlso*. Embora os resultados não tenham sido favoráveis quanto às relações, ao analisarmos as relações de FP, que foram sugeridas pelo DM^D, elas podem ser úteis no aprimoramento do DM^1, e que por algum motivo tenham ficado de fora da análise do primeiro especialista.

Outro ponto interessante na modelagem é a especialização das classes, de c_2 a c_5, em relação a classe c_1 no DM^D. Diferentemente da modelagem de DM^D, o especialista em DM^1 utilizou a composição, por exemplo, entre as classes, **Movement Contact Operation** e **Offensive Operation**, em detrimento de uma hierarquia. Acreditamos que essa decisão foi fundamentada em uma estratégia de modelagem recomendada em Gamma et al.(176). Nessa publicação, os autores afirmam que as composições podem reduzir o acoplamento, favorecer o reúso e aumentar a flexibilidade, minimizando a rigidez da hierarquia de classes (176). Outra suposição é que o especialista do domínio pode ter entendido que a "operação ofensiva" seja um evento composto de outros eventos que ocorrem ao longo do tempo. Independente disso, como mencionado, o próprio Kent(104) afirmava que não há somente diferentes visões de um mesmo fato, há também formas distintas de representar o mesmo fato, pois um DM retrata uma realidade simplificada de um domínio sob uma perspectiva particular do especialista.

Dessa forma, o papel da abordagem IDEA-C2 é oferecer os subsídios para apoiar o especialista na elaboração do DM, porém a decisão é inteiramente dele. Mediante a isso, é arriscado afirmar que um DM está certo ou errado. Na realidade, um DM é o resultado de escolhas de técnicas e estratégias de modelagem sob uma perspectiva e não é uma verdade absoluta. Cabe ressaltar que as análises aqui apresentadas não cessam as possíveis formas de representação de um DM. Outras questões também poderiam ser discutidas, como a inclusão de c_{15} e c_{16}, desdobrando dois ataques empreendidos em um ataque coordenado, c_{10}. Contudo, buscamos destacar somente alguns pontos relevantes entre ambos os DM

com foco na entidade **operacao_ofensiva**.

Portanto, de acordo com as evidências do experimento, acreditamos que a abordagem IDEA-C2 atende à proposição da hipótese H4, demonstrando que é possível apoiar a construção de um DM no contexto militar ao submeter textos expressos em linguagem natural a um ML ajustado, combinando com a exploração dos dados em um KG. Os resultados alcançados são promissores e indicam que o DM construído com o suporte do IDEA-C2, combinando as abordagens DD e TD, pode ser comparado com um outro DM construído de maneira tradicional, utilizando a abordagem TD. Mesmo assim, o IDEA-C2 não se propõe a substituir a abordagem tradicional. Na realidade, o IDEA-C2 desempenha um papel complementar e relevante na construção de um DM. Principalmente, ao se valer da quantidade abrangente de elementos obtidos através da abordagem DD, refinado pela abordagem TD através da exploração e conceituação desses elementos no KG. Além disso, os resultados sobre os tipos de relação podem ser aperfeiçoados, motivando novos experimentos, como veremos na seção 6.5.

6.5 Ex_5: Avaliação de DM^n elaborados em relação ao DM^D

O experimento Ex_5, análogo ao anterior, visa validar a hipótese H4 que propõe o seguinte: "É possível apoiar a construção de um DM no contexto militar a partir da submissão de textos a um ML ajustado combinando com a exploração dos dados em um KG." Diferentemente de Ex_4, este experimento envolveu um número razoável de participantes que conceberam modelos de domínios distintos, representados por DM^n, onde n identifica o DM do participante, à luz de um minimundo.

Para a realização deste experimento ser viável, simplificamos a interação dos participantes à nossa abordagem através de um extrato do IDEA-C2-KG, previamente preparado e focado no minimundo. Esse extrato do KG foi oferecido à metade dos participantes do experimento, o qual foi gerado com base no minimundo processado pelo modelo de linguagem. A outra metade dos participantes construiu o DM de maneira tradicional, sem nenhum suporte do IDEA-C2. O objetivo neste experimento é oferecer um artefato simplificado para apoiar uma parte dos participantes na construção de seus DM^n e, posteriormente, comparar com aqueles não utilizaram IDEA-C2. Dessa forma, estruturou-se o experimento em duas subseções que lidam desde a caracterização e objetivo, apresentado na subseção 6.5.1, até as avaliações através da análise dos resultados quantitativos e qualitativos alcançados, apresentado na subseção 6.5.2.

6.5.1 Caracterização do experimento

O objetivo do experimento Ex_5 é avaliar como o IDEA-C2 pode apoiar a elaboração de um DM a partir dos recursos do IDEA-C2-KG, o qual foi construído com base no ajuste

fino do IDEA-C2-LM. Para tal, foi elaborado um minimundo no domínio militar, a partir da Doutrina de Operações Conjuntas - MD30-M-011 (119), explorando o processo de criação de uma Operação Conjunta (Op Cj). Esse minimundo foi distribuído a especialistas de diversas áreas e níveis de experiências para elaborem um DM que atenda às Questões de Competência (QC) desse domínio, como apresentado na seção B.1.

Os participantes foram divididos aleatoriamente em dois grupos de forma igualitária. O primeiro grupo recebeu e utilizou o fragmento do IDEA-C2-KG para dar suporte aos participantes na construção de cada DM^n, ilustrado na Figura 36. O segundo grupo de participantes utilizou somente seus conhecimentos sobre o domínio, sem nenhum suporte para construir DM^n. A convocação dos participantes se deu por e-mail, como ilustrado na Figura 39. No e-mail, foi incluído um link para um formulário Web[15], ilustrado na Figura 40. O formulário foi elaborado compondo perguntas que mapeiam o perfil do participante, capturam a sua percepção sobre a modelagem conceitual e questionam aspectos relevantes do experimento, como apresentado na seção B.6.

Cabe destacar que dos 35 participantes convidados por e-mail, 28 aceitaram a convocação, alcançando 88% de participação. Ao término, os participantes submeteram seus respectivos DM^n, disponíveis no repositório[16] do experimento, para avaliação e comparação de seus objetos com DM^D. É importante salientar que os dados dos participantes bem como seus modelos de domínio não possuem nenhuma identificação ou marca pessoal, resguardando, assim, o sigilo das informações prestadas.

6.5.2 Avaliação do experimento

Para avaliar este experimento, foi utilizado IDEA-C2 para apoiar a construção de um modelo de domínio[17], DM^D, observando as questões de competência e o minimundo disponível no experimento. Assim, assume-se DM^D como um modelo de domínio padrão, do tipo *gold standard*, a fim de servir de base de comparação de seus objetos (classes e os pares de classes relacionadas), como ilustrado na Figura 37. Análogo ao experimento anterior, para realizar a comparação, consideramos cada DM como uma tupla $DM = (DM_C, DM_R)$, onde DM_C é o conjunto de classes de DM e DM_R é o conjunto das suas relações.

No Ex_5, foram definidos três estágios, similar ao Ex_4, porém com algumas adaptações. Nesses estágios, os objetos de $DM^D \times DM^n$ são comparados, onde n representa o identificador do DM elaborado por cada participante. No primeiro estágio, os conjuntos de classes, DM^D_C e DM^n_C, são comparados, par a par, i.e., cada par $(c_i, c^n_p) \in (DM^D_C \times DM^n_C)$ é analisado. Quando ambas as classes de um par representam o mesmo conceito, ele

[15] Link de acesso: https://forms.gle/YQqgfw64Cf6t8R9w6

[16] <https://github.com/comp-ime-eb-br/S2C2-IME/tree/main/deliverables/idea-c2/experimentos/exp5/modelagens>

[17] <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/deliverables/idea-c2/experimentos/exp5/E5-DMD.pdf>

é quantificado como um *verdadeiro positivo* (TP), como por exemplo a classe **operação conjunta** presente tanto em DM^D (Figura 37) quanto em DM^{24} (Figura 38). Cabe destacar que DM^{24}, utilizada como exemplo, foi elaborada por um dos participantes do experimento. O somatório de ocorrências verdadeiras positivas é dado por $\sum_C TP_C^n = \sum_{nC} |(DM_C^D \cap DM_C^n)|$[18]. Diferentemente do experimento anterior, não são calculadas as outras variáveis, FP e FN, tampouco as métricas de precisão e *recall*, porque neste experimento é avaliado o suporte do IDEA-C2 na construção do DM, interessando somente o TP. Por fim, calcula-se o percentual de objetos comuns de DM^n em relação ao total de classes de DM^D, através de $P_{TP_C^n} = 100 \cdot \frac{|DM_C^D \cap DM_C^n|}{\sum |DM_C^D|}$, apresentado na Tabela 6.

No segundo estágio, os conjuntos de relações, DM_R^D e DM_R^n, são comparados da seguinte maneira. Inicialmente, os pares de classes relacionados são comparados, desconsiderando seu tipo de relação. Cada par $((c_i, c_k), (c_p^n, c_r^n))$ é analisado, onde $(c_i, R_j, c_k) \in DM_R^D$ e $(c_p^n, R_q^n, c_r^n) \in DM_R^n$, assumindo que os pares $(c_i, c_p^n), (c_k, c_r^n)$ foram previamente classificados como verdadeiros positivos. Assim, o somatório de ocorrências verdadeiras positivas é dado por $\sum_C TP_R^n = \sum_{n_R} |(DM_D^R \cap DM_R^n)|$. Por exemplo, análogo ao primeiro estágio, o fato de o par de classes, **operação militar** e **operacao conjunta**, estar relacionado em ambos os DMs, DM^D e DM^{24}, indica um TP. Aqui também não são calculados FN e FP, tampouco a precisão e o recall. Além disso, o percentual de objetos comuns entre os pares de classes relacionados de DM_R^n e DM_R^D é dado por $P_{TP_R^n} = 100 \cdot \frac{|DM_R^D \cap DM_R^n|}{\sum |DM_R^D|}$, como apresentado na Tabela 6.

Tabela 6 – Comparação entre $DM^D \times DM^n$ elaborados com e sem o suporte do IDEA-C2

	Sem suporte do IDEA-C2					**Com suporte do IDEA-C2**			
	$DM_C^D \times DM_C^n$ **(Classes)**		$DM_R^D \times DM_R^n$ **(Relações)**			$DM_C^D \times DM_C^n$ **(Classes)**		$DM_R^D \times DM_R^n$ **(Relações)**	
DM^n	TP_C^n	$P_{TP_C^n}$	TP_R^n	$P_{TP_R^n}$	DM^n	TP_C^n	$P_{TP_C^n}$	TP_R^n	$P_{TP_R^n}$
1	29	50%	27	44%	4	23	40%	23	38%
2	22	38%	25	41%	**7**	**29**	**50%**	**35**	**57%**
3	15	26%	9	15%	8	25	43%	22	36%
5	21	36%	27	44%	11	21	36%	23	38%
6	20	34%	27	44%	13	27	47%	32	52%
9	23	40%	21	34%	17	18	31%	14	23%
12	25	43%	29	48%	18	21	36%	17	28%
15	23	40%	25	41%	22	25	43%	23	38%
16	22	38%	24	39%	23	13	22%	13	21%
19	26	45%	27	44%	**24**	**35**	**60%**	**39**	**64%**
20	16	28%	19	31%	**26**	**37**	**64%**	**36**	**59%**
21	17	29%	17	28%	**27**	**33**	**57%**	**38**	**62%**
25	18	31%	17	28%	29	25	43%	26	43%
28	**30**	**52%**	**33**	**54%**	30	19	33%	22	36%

(i) Somatório (TP_C^n) e percentual ($P_{TP_C^n}$) de classes comuns.
(ii) Somatório (TP_R^n) e percentual ($P_{TP_R^n}$) de relações comuns.

[18] Operação de conjunto como ∩ aplicada aqui no significado semântico dos elementos do conjunto.

A Tabela 6 apresenta os resultados quantitativos alcançados da comparação entre DM^n e DM^D. Inicialmente, os dois grupos que elaboraram cada DM^n são classificados em virtude do suporte ou não do IDEA-C2. Em seguida, é calculado o grau de objetos comuns por conjunto de classes, $(DM_C^D \times DM_C^n)$, e classes relacionadas, $(DM_R^D \times DM_R^n)$. Por fim, são destacados em cinza, os DM^n que alcançaram o maior número de objetos comuns entre DM^n em relação a DM^D.

No geral, ao analisar os números, pode-se deduzir que os participantes que utilizaram IDEA-C2 como suporte para elaborar os DMs alcançaram resultados superiores em comparação ao grupo que não utilizou a nossa abordagem. Dentre os participantes que não utilizaram IDEA-C2, somente o modelo de domínio, DM^{28}, alcançou o resultado superior a 50% de objetos comuns de classes e relações quando comparado a DM^D. Em contrapartida, os resultados alcançados pelos participantes que utilizaram IDEA-C2 superaram quatro vezes aqueles que não a utilizaram, destacando os modelos de domínio, DM^7, DM^{24}, DM^{26} e DM^{27}.

Considerando a comparação envolvendo apenas classes individualmente, no grupo que utilizou IDEA-C2 como suporte, houve participante que chegou a superar 60% de objetos comuns, como foi o caso de DM^{26}. Além disso, considerando os pares de classes, os resultados alcançados também são promissores. Em destaque o modelo de domínio, DM^{24}, que alcançou 64% de objetos comuns entre os pares de classes e relações. Ambos os resultados demonstram que IDEA-C2 pode contribuir na construção de DMs, oferecendo aos usuários um grafo de conhecimento rico (Figura 36) que através de seus recursos pode permitir a exploração navegacional, assim como favorece a descoberta de relações semânticas dentro de um determinado domínio. Cabe mencionar que os dados de comparação, em detalhes, são armazenados em uma planilha disponível para consulta[19].

No terceiro estágio, foi realizada a avaliação qualitativa, envolvendo vinte e oito participantes, por meio de um questionário, como apresentado no seção B.6. Ao analisar o setor de atuação, identificou-se que 82% atuam no setor público, englobando militares e civis, respectivamente, com 78% e 22%. O setor privado corresponde a 18% dos participantes. Ao analisar o grau de escolaridade, 53% são graduados, 21% são mestres, 10% são doutores e o restante são alunos de graduação. Quando analisamos o tempo de experiência, identificamos que 64% dos participantes possuem até 5 anos de atuação na área de Tecnologia da Informação. E o restante possui mais de 10 anos de experiência. Entretanto, quando questionamos o tempo de experiência em modelagem de dados conceitual, percebemos que houve uma diferença considerável em relação ao tempo de atuação profissional. Uma parte dos participantes, cerca de 68%, responderam que possuem menos de 2 anos de experiência em modelagem conceitual. Isso nos levou a investigar qual foi o resultado

[19] <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/deliverables/idea-c2/experimentos/exp5/E5-An%C3%A1lises.xlsx>

médio desses participantes que possuem menor tempo de experiência, principalmente em relação aos objetos comuns dos DM^n em relação ao DM_D. Como resultado, identificamos que o grupo menos experiente alcançou um percentual abaixo de 39%, envolvendo classes e relações. Esse resultado ficou bem abaixo dos resultados de DM^7, DM^{24}, DM^{26} e DM^{27}, que alcançaram mais de 60% entre as classes e relações.

Como o minimundo explorado no experimento é direcionado ao universo das operações conjuntas, indagamos os participantes o seguinte: "O minimundo apresentado expressa o cenário de maneira clara e concisa?" Nesse quesito, 14% concordaram totalmente com uma das afirmativas. Destacamos algumas das opiniões registradas, como por exemplo: "*Sinceramente, não tive maiores dificuldades porque já conhecia o domínio do negócio, mas eu ainda não tinha feito essa modelagem englobando os quatro níveis decisórios. Meu conhecimento concentrava-se no nível operacional*". E a maioria, cerca de 42%, "concordou parcialmente", destacando uma das afirmativas: "*O cenário é muito amplo, o que necessita conhecimento e pesquisa sobre o assunto, mesmo com o minimundo exposto, uma vez que ele não compreende todas as peculiaridades do assunto*". Além disso, nem discordou nem concordou alcançou 25%, acompanhado de 1% que discordaram parcialmente, e 3% discordaram totalmente. Ao analisar em detalhes esses números, observamos que mais da metade dos participantes tiveram uma impressão positiva sobre o minimundo abordado.

Quanto à elaboração do modelo de domínio, indagamos aos participantes o seguinte: "Durante a modelagem ocorreram situações de dúvida quanto à representação de uma entidade ou relacionamento?" A esse respeito, 75% dos participantes responderam que "concordam totalmente e parcialmente". Nesse caso, destacamos uma das afirmativas: "*O minimundo é complexo e exige uma modelagem com muitas entidades*". Isso nos leva a crer que apesar da exposição do minimundo ter sido bem aceita pela maioria, mesmo assim ainda apresenta dificuldades na identificação das classes e relações. Além disso, indagamos os participantes sobre a possibilidade de construir o DM em 60 minutos. A despeito do tempo, 50% dos entrevistados responderam que "discordam totalmente e parcialmente". Em destaque a afirmativa: "*Não tenho tanto conhecimento sobre o assunto e pouca prática. Em concordância, durante a modelagem sempre surgem dúvidas de como fazer tal coisa e isso faz com que 60 minutos, sem ninguém pra tirar dúvida, não seja tão produtivo assim*".

Quanto ao apoio do IDEA-C2-KG, indagamos os participantes o seguinte: "Você utilizou o grafo (IDEA-C2-KG) como apoio?" Essa indagação foi dividida em cinco questões. Nas duas primeiras questões, abordamos sobre as entidades e relações do grafo retratarem o domínio do minimundo. Nesse quesito, 56% "concordam parcialmente e totalmente", demonstrando a relevância do artefato no experimento. Porém, ao considerar somente as entidades, 42% responderam que "nem discordam nem concordam". Essa mesma resposta foi dada por 28% dos participantes quando consideraram apenas as relações. Com o objetivo de corroborar com essa avaliação, destacamos a seguinte afirmativa: "*...O grafo*

ajuda, mas achei que o grafo em alguns pontos tinham coisas a mais e em outros pontos poderiam ser expandidos (tipo como no GraphDB quando a gente clica 2 vezes e o grafo vai sendo expandido)".

Na terceira questão, foi indagado aos participantes o seguinte: "Compreendi claramente os nós e arestas do grafo em relação ao contexto do minimundo." Nesse aspecto, a maioria dos participantes, cerca de 71%, responderam que "concordam plenamente e parcialmente", acompanhado de 28% que "nem discordam nem concordam". Assim, é razoável deduzir que a visualização gráfica do IDEA-C2-KG contribuiu positivamente na tarefa de modelagem realizada pelos participantes. Outras duas indagações sobre o grafo foram realizadas. Uma delas questiona o seguinte: "No geral, o grafo contribuiu com a elaboração do modelo conceitual do minimundo." Nesse ponto, as respostas foram bem distribuídas e não houve unanimidade, pois 14% responderam que "discordam totalmente" e o mesmo percentual se repetiu para "discordo parcialmente" e "concordo parcialmente". Ademais, 28% responderam que "nem discordam nem concordam" e de igual modo "concordo totalmente.

Por fim, os participantes foram indagados sobre o seguinte: "Recomendo o uso de grafo de conhecimento em apoio à elaboração de modelos conceituais? Cerca de 57% responderam que "concordam parcialmente", acompanhados de 28% que "concordam totalmente". Por outro lado, uma porção mínima de 14% respondeu que "discorda parcialmente". Em suma, a avaliação qualitativa dos participantes, demonstra que o uso do grafo foi positiva para a maioria que participou do experimento. Com base nas respostas, algumas melhorias podem ser introduzidas. Uma delas envolve a interação dinâmica do IDEA-C2-KG, assim como a submissão de consultas ao grafo para explorar outros recursos.

Portanto, de acordo com as evidências do experimento, acreditamos que a abordagem IDEA-C2, principalmente quanto ao uso do IDEA-C2-KG como apoio à construção de DM, atende à proposição da hipótese H4. Demonstramos que é possível apoiar a construção de um DM no contexto militar ao submeter textos expressos em linguagem natural a um ML ajustado, combinando com a exploração dos dados em um KG. Os resultados quantitativos e qualitativos alcançados são promissores e confirmaram que a aplicação do IDEA-C2-KG no apoio à construção de DMs em comparação aos métodos tradicionais foi positiva. Em particular, as relações de C2RM demonstraram ser úteis de modo que podem ser exploradas no KG, permitindo que os especialistas de domínio possam aumentar o seu grau de conhecimento sobre o negócio, minimizando o tempo gasto e o acesso às doutrinas.

6.6 Ex_6: Avaliação da geração do IDEA-C2-LM incorporada com a taxonomia do MAISC^2

O experimento Ex_6 estende o Ex_1 e tem como objetivo validar a hipótese H2 que propõe o seguinte: "Um metamodelo que permite metacategorizar as entidades e relações pode flexibilizar a anotação de um corpus para o ajuste fino de um ML nas tarefas NER e RE.". A extensão mencionada neste experimento diz repeito à geração de um ML ajustado ao contexto, utilizando a abordagem *Multicategory*. Para tal, é explorado um cenário de Comando e Controle (C2) a partir da instanciação da taxonomia MAISC^2 (4), como abordado na subseção 6.6.1. Diferentemente de Ex_1 que validou a estratégia *Singlecategory*, neste experimento os construtos de alto nível de C2RM são preservados e um novo Recurso Semântico (RS) é instanciado.

A partir da nova instância, as categorias de entidades são incorporadas com o objetivo de aprimorar o subprocesso **Anotar corpus**. Nesse subprocesso, as categorias e os termos instanciados do MAISC^2 são comparados ao mapeamento existente nas regras do IDEA-C2. Caso exista um mapeamento no IDEA-C2 correspondente, ele é substituído por aquele indicado em MAISC^2. Caso contrário, ele é incluído automaticamente. Por exemplo, no IDEA-C2 o termo "aeronave" é uma instância de *Entity* de C2RM. Contudo, esse mesmo termo em MAISC^2 é uma instância de *vehicle*. Nesse caso, IDEA-C2 passa a adotar o mapeamento do MAISC^2 na pré-anotação do corpus, como abordado em detalhes na subseção 6.6.2.

O novo corpus é submetido ao subprocesso **Ajustar modelo de linguagem** a fim de gerar um novo IDEA-C2-LM, voltado à abordagem *Multicategory*, nas tarefas NER e RE. Além disso, a performance alcançada na abordagem *Multicategory* é avaliada comparativamente à abordagem *Singlecategory*, como abordado na subseção 6.6.3. Ao IDEA-C2-LM *Multicategory* são submetidos textos do cenário de aplicação para avaliar os resultados da interação com ML, comparando-o ao IDEA-C2-LM *Singlecategory*, como abordado na subseção 6.6.4. Cabe salientar que este experimento se limita a explorar o cenário, porém está fora do escopo a reprodução das demais funcionalidades do MAISC^2, como a classificação de prioridade, a entrega de mensagens e a geração de estatísticas.

6.6.1 Caracterização e Taxonomia do MAISC^2

Este experimento explora o cenário de troca de mensagens em um ambiente operativo militar de C2, que pode envolver complexidades de comunicação em função das diferentes Forças Armadas, a partir da abrangência e objetivos estratégicos empregados em uma operação militar (4). Nesse sentido, destacamos o trabalho de Mosafi et al.(4) que explora um cenário típico de C2 através de um chat tático que apoia seus usuários na troca de informações, ajudando a compreensão das mensagens trocadas e aprimorando a

interpretação das informações.

Por exemplo, supondo a comunicação fictícia, expressa em linguagem natural, entre um soldado e um comandante:"*MSG1: SOLDADO: Comandante, avistei uma aeronave e um pelotão inimigo. Solicito autorização para engajar.*" "*MSG2: COMANDANTE: Soldado, autorizada a ação de operação ofensiva sobre o inimigo.*" Nesse caso, espera-se que as entidades nomeadas sejam reconhecidas e classificadas de acordo com a sua categoria. Apesar de hipotético e simplificado o exemplo, essas comunicações em uma operação militar são volumosas e envolvem diversos atores e níveis hierárquicos. Um exemplo é a Operação de Garantia da Lei e da Ordem que ocorreu, em 2024, no Rio de Janeiro, durante a cúpula de líderes do G20, onde foram empregados 44 mil militares em atividades de segurança com os outros órgãos do Estado (46).

Quadro 14 – Taxonomia de categorias do MAISC^2. Adaptado de Mosafi et al.(4).

Categoria	**Acrônimo**	**Exemplo de possíveis termos**
Action	ACT	atacar, recuar, resgatar, comunicar
Direction	DRT	norte, sul, leste, oeste
Device	DVC	notebook, walkie-talkie, antena, lanterna
Event	EVT	incêndio, tempestade, missão, resgate
Place	PLC	planície, bairro, avenida, região
Agent	AGT	policial, médico, sargento, consultor, comandante
Supplies	SPL	alimentos, medicamentos, munição, água
Unit	UNT	quartel-general, pelotão, esquadrão
Vehicle	VHC	avião, carro, motocicleta, tanque
Weapon	WEP	pistola, baioneta, faca, metralhadora

A taxonomia do MAISC^2 foi definida a partir de um conjunto de categorias específicas, como apresentado no Quadro 14, com o objetivo de apoiar a classificação dos termos utilizados no contexto de C2 (4). As categorias de entidades são definidas de acordo com a aplicação no domínio militar, como exemplo: i) *Action* (ACT) que determina uma ação de ataque, retirada, resgate, etc.; ii) *Direction* (DRT) que indica a localização (norte, sul, leste e oeste); iii) *Device* (DVC) que determina o tipo de equipamento (e.g. notebook, antena, etc.); iv) *Place* (PLC) que indica lugar (e.g. região, avenida, etc.); v) *Vehicle* (VHC) que indica um meio (e.g. aeronave, tanque, etc.). Além dessas categorias, há outras não mencionadas.

Essa taxonomia estrutura um conjunto de termos utilizados no cenário de C2, definindo uma hierarquia agregada para cada um deles. Cada agregador pode ser identificado como uma categoria de entidade. Como IDEA-C2 é uma abordagem flexível e permite a instanciação de RS, é oportuno investigar como essas categorias podem ser incorporadas em nossa abordagem e quais resultados podem ser obtidos. Todavia, IDEA-C2 foi concebida como uma abordagem *singlecategory*, principalmente em função da geração do KG, sendo necessário realizar algumas adaptações para não somente permitir a instanciação de outro RS, mas também incorporar ao IDEA-C2 as características da abordagem *multicategory*.

Dessa forma, IDEA-C2 pode se tornar uma abordagem híbrida e aplicada em diversos propósitos de apoio às atividades no domínio militar, como será abordado na próxima subseção.

6.6.2 Incorporação do MAISC^2 no subprocesso Anotar Corpus

A incorporação de um novo RS ao IDEA-C2 requer adaptações no subprocesso **Anotar corpus**. Inicialmente, deve ser incluído um objeto de entrada, D, para permitir a obtenção da taxonomia do MAISC^2, dada por $D = \{D_1, D_2...D_n\}$. Além disso, em **Definir regras de pré-anotação**, uma atividade descrita em Avelino et al.(2), na tarefa NER, originalmente, as categorias são identificadas como *entity*, onde o conjunto de entidades é definido por $E = \{e_1, e_2...e_n\}$, pois são instâncias de *Entity* de C2RM. Ao considerar que D foi instanciado, i.e., utilizando a abordagem *multicategory*, IDEA-C2 incorpora as categorias de entidades, representadas por $\forall x\,(D_i(x) \rightarrow D_j(x))$, onde cada subclasse $D_i(x)$ é associada a uma superclasse, $D_j(x)$. Assim, para cada termo $D_i(x)$, admite-se como categoria de entidade $D_j(x)$. Caso não haja instanciação de D, mantém-se a categoria genérica *Entity*, como ilustrado a seguir.

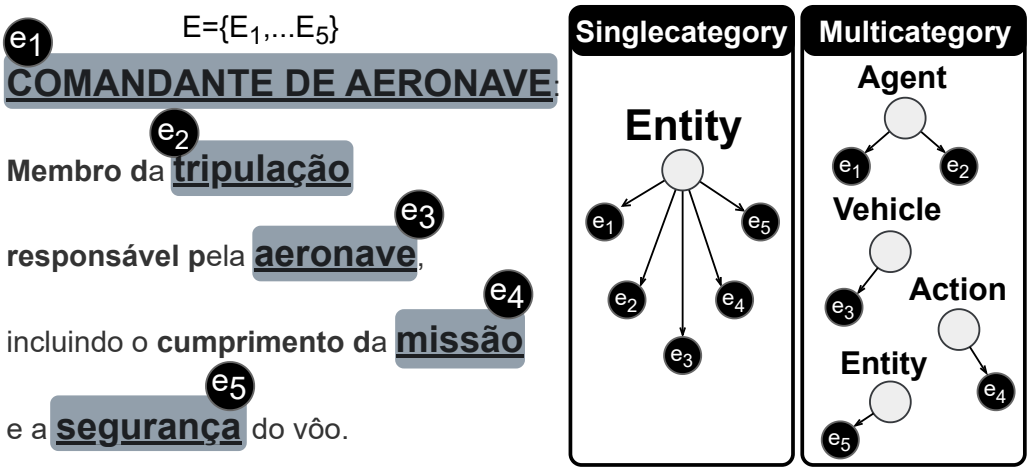


Figura 33 – Representação híbrida de categorias de entidades. Imagem do autor.

Na Figura 33, ao considerarmos o trecho do corpus C, "*COMANDANTE DE AERONAVE: Membro da tripulação responsável pela aeronave, incluindo o cumprimento da missão e a segurança do vôo.*", são ilustrados dois quadros, em formato de grafo, representando o tipo de abordagem (*singlecategory* e *multicategory*), com exemplos de categorias de entidades assumidas.

No quadro *Singlecategory*, o conjunto E é constituído de termos de RS, representando as instâncias de categoria única diretamente de *Entity*, formado por $E = \{\text{comandante\_de\_aeronave}, \text{tripulação}, \text{aeronave}, \text{missão}, \text{segurança}\}$. No quadro *Multicategory*, são ilustradas as instâncias do conjunto de categorias definidas por D, onde $D_i(x)$

e $D_j(x)$ podem assumir os valores, respectivamente, *comandante de aeronave* e *agent*, denotado por $\forall x\,(comandante\_de\_aeronave \rightarrow agent)$. Cada valor de $D_i(x)$ é definido como regra de categoria de entidade a ser identificada no corpus C, como representado nos subgrafos $Agent$, $Vehicle$, $Action$ e $Entity$. Esses valores são armazenados no conjunto $E = \{E_1, ...E_5\}$. Note que na ausência de categorias de entidades de D em relação a C, a regra de mapeamento é estabelecida conforme a abordagem *singlecategory*, como ilustrado no exemplo de E_5, representado por $\forall x\,(Missão \rightarrow Entity)$.

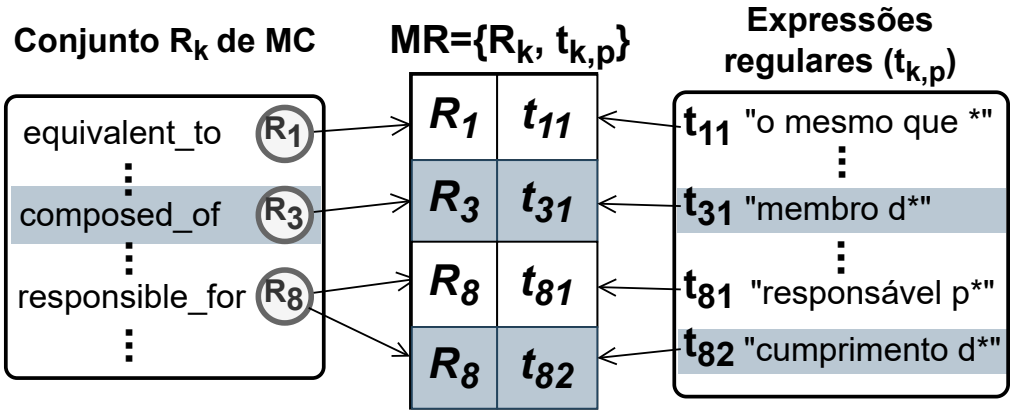


Figura 34 – Exemplo de regras de pré-anotação no corpus C. Imagem do autor.

Por outro lado, na tarefa RE, com base nas especializações de C2RM são elaboradas regras de pré-anotação, $MR = \{Rk, tk, p\}$, onde Rk é uma especialização de R e tk, p representa uma expressão regular associada à Rk (2). Cada expressão regular é elaborada a partir de padrões textuais que é explorada na atividade **Pré-anotar entidades e relações**, uma atividade de Avelino et al.(2), que avalia e anota os termos entre pares de categorias de entidades em uma mesma sentença.

Na Figura 34, são ilustrados três exemplos dos pares ordenados (E_1, E_2), (E_1, E_3) e (E_1, E_4) extraídos do exemplo da Figura 33. Ao explorar os padrões dos textos entre os valores dos pares ordenados, encontram-se expressões como ("membro d*", "responsável p*" e "cumprimento d*") que indicam expressões regulares que darão origem às regras para anotar os textos do corpus C, utilizando as especializações de R, como exemplos: *composed_of* para "membro d*", além de *responsible_for* para "responsável p*" e "cumprimento d". Ao término, é acionada a atividade **Pré-anotar entidades e relações** que gera o corpus C', permitindo que os usuários especialistas possam realizar a curadoria e, posteriormente, gerar o corpus C'' para executar o subprocesso **Ajustar modelo de linguagem**, abordado na próxima subseção.

6.6.3 Ajuste fino do IDEA-C2-LM (*Multicategory*)

No subprocesso **Ajustar modelo de linguagem**, o corpus C'' foi submetido com o objetivo de ajustar os pesos do ML pré-treinado no domínio militar, nas tarefas de NER e RE, agora utilizando a abordagem *Multicategory*. Para tal, assumiu-se a mesma configuração base do experimento Ex_1, com destaque ao BERTimbau (36), ao *pipeline* pt_core_news_sm do SpaCy, parametrizado, respectivamente, com os valores: Batch_size de 128 e 500; Max_Length de 4096 e 250; e Threshold de 0,5 para RE. Cabe ressaltar que durante o experimento, foram testados outros *pipelines*, como pt_core_news_md e pt_core_news_lg, no entanto seus resultados foram inferiores ao pt_core_news_sm. Além disso, o código-fonte do experimento, bem como os resultados da execução estão disponíveis em repositório público[20].

Tabela 7 – Comparação do ajuste fino do IDEA-C2-LM (*Singlecategory* vs. *Multicategory*)

Modelo de linguagem	**Tarefa**	**Tipo de abordagem**	**P**	**R**	**F1**
BERTimbau	NER	Single	86%	82%	84%
		Multi	**87%**	**85%**	**86%**
	RE	**Single**	**94%**	**89%**	**91%**
		Multi	86%	87%	86%

Single: *Singlecategory*; Multi: *Multicategory*;
P: Precisão; R: *Recall*; F1: *F1-Score*.

Na Tabela 7, são apresentados os resultados do ajuste fino do ML instanciado, especificando o tipo de tarefa (NER e RE) e o tipo de abordagem utilizado. Ao analisar a tarefa NER, em destaque, observa-se que os valores das métricas da abordagem *Multicategory* foram ligeiramente superiores, não havendo distorções no número de *steps* por rodada de treinamento, cujo os valores, de 304 na *Single* e 385 na *Multi*, foram aproximados. Por outro lado, ao avaliar os resultados na tarefa RE, apesar das métricas de precisão e *recall* e F1-score serem inferiores na abordagem *Multicategory*, houve uma diferença de quase 50% no número de *steps* (134 e 66) quando comparamos com a *Singlecategory*. Portanto, isso nos leva a crer que a abordagem *Multicategory* pode gerar modelos com maior capacidade de precisão de resultados, provavelmente influenciado pela maior distribuição dos pesos por categoria e suas relações.

Na Figura 35, é ilustrada a matriz de confusão que mede a performance do aprendizado do IDEA-C2-LM através do conjunto de categorias de entidades instanciado do MAISC[2] (4). Na matriz, são comparados os pares de cada categoria através da relação dos valores apresentados dentro (previsão correta) e fora (erro de classificação) da diagonal principal. Note que as categorias ***action, unit, agent, place, weapon*** possuem uma alta taxa de verdadeiros positivos, atingindo valores acima de 80%. Em contrapartida, a

[20] <https://github.com/comp-ime-eb-br/S2C2-IME/tree/main/deliverables/idea-c2/experimentos/exp6>

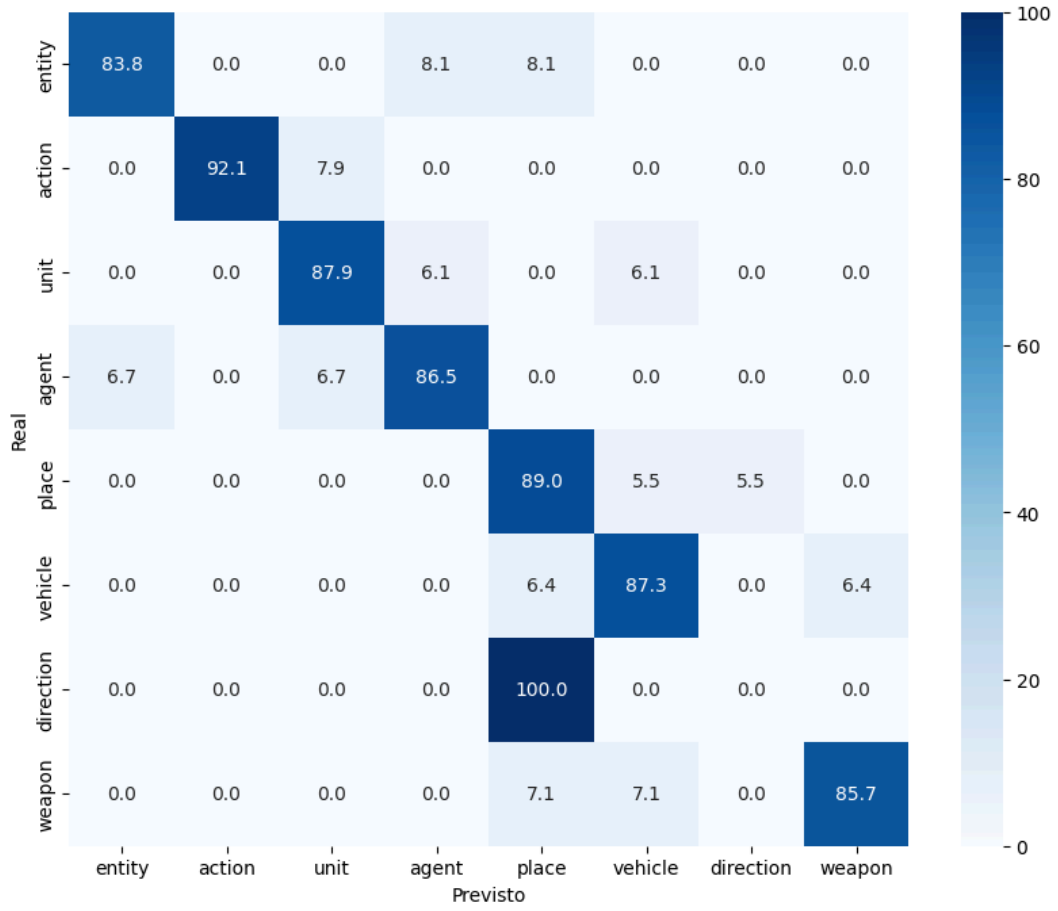


Figura 35 – Matriz de confusão das categorias de entidades na tarefa NER. Imagem do autor.

categoria ***direction*** não obteve um bom resultado, provavelmente em função de poucos exemplos de anotação. Além disso, houve algumas distorções de padrões de confusões semânticas, talvez em função da proximidade de algumas categorias na matriz de *embeddings*, como nos casos de (*entity* - *agent*) e (*vehicle* - *weapon*). Portanto, como o desempenho do ajuste fino pode ser analisado em função de vários fatores, uma linha de ação possível a ser investigada seria o aperfeiçoamento das regras do algoritmo de pré-anotação, especialmente na distinção das categorias e termos no corpus.

Como pôde ser observado, o resultado do ajuste fino um de ML pode sofrer influências diversas. A variação de hiperparâmetros, o tipo de tarefa desejada, a qualidade de anotação do corpus, bem como as estratégias de abordagens (*single* e *multi category*) devem ser ajustadas e testadas continuamente através de rodadas exaustivas. Além disso, as evidências ajudam a confirmar que na maioria dos casos a *Singlecategory* pode gerar ML ajustados ao domínio com maior cobertura (*recall*), porém com menor precisão em função

de possuir somente uma única categoria. Contudo, em nosso experimento, as evidências nos levam a crer que a abordagem *singlecategory* é propícia para a geração de *Knowledge Graph (KG)* dada a generalização dos resultados, postergando a identificação das entidades para a exploração no próprio KG.

Em contrapartida, a abordagem *Multicategory*, além da precisão e da diversidade de categorias, pode ser aplicada em contextos que a exatidão do modelo deve ser o diferencial do cenário de aplicação, como exemplo, a detecção de segurança ao avaliar se uma mensagem possui ou não um *spam*. No entanto, outros experimentos ainda devem ser conduzidos, inclusive as especializações *General domain relation* do metamodelo C2RM podem ser expandidas, por exemplo, com a introdução de relações antônimas para demonstrar a semântica entre duas entidades opostas. Por fim, com o IDEA-C2-LM ajustado devem ser submetidos textos para avaliar os seus resultados, como será abordado na próxima subseção.

6.6.4 Interação com IDEA-C2-LM no cenário de C2

Nesta subseção, com base nos textos obtidos através da troca de mensagens hipotética ocorrida entre o soldado e o comandante, abordado na subseção 6.6.1, é conduzida a interação com IDEA-C2-LM, *single* e *multi category*, com o objetivo de comparar as entidades identificadas e as relações extraídas (Tabela 8).

Tabela 8 – Comparação da interação do IDEA-C2-LM (*Singlecategory* x *Multicategory*)

IDEA-C2	**NER**	**RE**
Single	[0, 'Comandante', 'entity'], [1, 'aeronave', 'entity'], [2, 'pelotão', 'entity'], [3, 'Soldado', 'entity'], [4, 'ação', 'entity'], [5, 'operação ofensiva', 'entity']]	[0, 'Comandante', 'associated_with', 'operação ofensiva']
Multi	[0, 'Comandante', 'agent'], [1, 'aeronave', 'vehicle'], [2, 'pelotão', 'unit'], [3, 'engajar', 'entity'], [4, 'Soldado', 'entity'], [5, 'ação', 'entity'], [6, 'operação ofensiva', 'action'], [7, 'inimigo', 'unit']	[0, 'Comandante', 'associated_with', 'operação ofensiva'], [1, 'Soldado', 'associated_with', 'operação ofensiva']

Na Tabela 8, ao analisar os resultados alcançados, as evidências nos levam a crer que o IDEA-C2-LM (multi) obteve melhor desempenho, pois reconheceu um número maior de entidades nomeadas e relações. Mesmo assim, cabe destacar que ao inferir "engajar" como uma entidade nomeada, percebe-se uma possível alucinação do IDEA-C2-LM (multi). Entretanto, em uma avaliação preliminar, as demais entidades nomeadas foram reconhecidas no texto e categorizadas de acordo com o conjunto de categorias utilizadas no

ajuste fino. Além disso, o IDEA-C2-LM (multi) obteve melhores resultados na inferência das relações entre essas entidades. No entanto, ao avaliar o rótulo semântico inferido, nota-se que o modelo utilizou *associated_with* que representa uma relação genérica, i.e., com baixa expressividade semântica. Já o IDEA-C2-LM (single) obteve resultados satisfatórios, essencialmente por servir como apoio à geração de KG, favorecendo a inferência de novas entidades e relações diretamente no KG.

Portanto, de acordo com as evidências do experimento, acreditamos que ao instanciar a taxonomia MAISC^2, preservando as metacategorias de C2RM, para gerar um ML ajustado utilizando a abordagem *Multicategory*, conseguimos atender à proposição da hipótese H2. Demonstramos que um metamodelo que permite metacategorizar as entidades e relações pode flexibilizar a anotação de um corpus para o ajuste fino de um ML nas tarefas NER e RE. Inclusive quando comparamos a geração do IDEA-C2-LM a partir de ambas as abordagens (*single* e *multi category*), notamos como a adoção de um metamodelo pode flexibilizar ao ponto de suportar a instância de outra taxonomia. Os resultados são promissores e a abordagem *multicategory* se destacou por melhorar a precisão e a cobertura do IDEA-C2-LM na tarefa NER. Porém, ainda há pontos de melhoria necessários para evoluir o trabalho, como veremos em detalhes na seção 6.7.

6.7 Análise crítica dos experimentos

Nos experimentos conduzidos nesta tese, buscamos não somente validar as hipóteses, mas também identificar oportunidades de melhoria durante o desenvolvimento da pesquisa. Para cada experimento foram demandados esforços na direção de alcançar resultados para medir o desempenho do trabalho.

O experimento Ex_1 demonstrou ser possível gerar ML ajustados a um domínio através de uma única categoria, comprovando que a hipótese H2, além de ser possível, também é viável. Como os resultados iniciais não foram satisfatórios, foi necessário aprofundar os estudos sobre os hiperparâmetros a fim de melhorar o ajuste fino do ML. Porém, uma possível evolução deste experimento seria adotar outros ML, como o mBERT (Multilingual BERT)[21], o AlBERTina (BERT em Português)[22] e o RoBERTa em Português[23]. Essa adoção pode colaborar com novas análises e resultados para avaliar o desempenho do ML ajustado em comparação com BERTimbau. Por fim, outra alternativa possível para evoluir o desempenho do IDEA-C2-LM (NER e RE), é adotar o paradigma "LLM-as-a-Judge". Esse paradigma permite incrementar o ML, avaliando a completude, as alucinações, as violações de restrição do domínio e a semântica das triplas de relações, como abordado no trabalho de Laskar et al.(177).

[21] https://huggingface.co/google-bert/bert-base-multilingual-cased
[22] https://huggingface.co/PORTULAN/albertina-900m-portuguese-ptbr-encoder-brwac
[23] https://huggingface.co/collections/eduagarcia/roberta-legal-portuguese

Em relação ao experimento Ex_2, nota-se que a anotação semiautomatizada da abordagem IDEA-C2, combinando regras de expressão regular heurísticas com Recursos Semânticos (RS), alcançou resultados promissores, principalmente quando levamos em consideração a minimização de esforços na curadoria do corpus. Entretanto, é possível evoluir o experimento, adotando análise de *parser* para detectar padrões nos textos quando há desdobramento de termos. Por exemplo: o termo "ação de" quando acompanhado de outro termo subsequente, pode indicar um tipo de "ação", como nos casos de "ação de choque" e "ação de comandos". Pode ainda adotar regras, utilizando *POS Tagging* que preveja contexto sintáticos, por exemplo, <*termo1*> + [**VERBO**]+ <*termo2*>, como abordado no trabalho de Avelino, Cordeiro e Cavalcanti(167). Por fim, pode ainda transformar cada regra em um conjunto a ser utilizado como uma *label function*, inclusive obtendo resultados de sua aplicação na rotina de anotação e curadoria.

O experimento Ex_3 alcançou resultados promissores de desempenho ao testar diferentes *pipelines* (pt_core_news_sm, pt_core_news_md e pt_core_news_lg). Assim, ao recuperarmos os resultados de cada *pipeline*, as diferenças são ligeiramente superiores em pt_core_news_sm e pt_core_news_md a depender do tipo de abordagem utilizada (*Single* e *Multicategory*). Contudo, novos experimentos podem ser conduzidos na direção de utilizar outro *pipeline*, mantendo logicamente a biblioteca SpaCy. Nesse sentido, o *pipeline* spaCy-LLM[24] pode ser introduzido e avaliado na arquitetura da abordagem IDEA-C2, utilizando, por exemplo, o Sabiá-7B[25], um ML pré-treinado em 7 bilhões de *tokens* a partir do subconjunto ClueWeb22, um corpus volumoso formado por textos em língua portuguesa (178). Dessa forma, justifica-se a adoção do Sabiá-7b a fim de testar no experimento a sua capacidade de inferência nos textos do domínio militar, principalmente em tarefas de extração semântica.

Os experimentos Ex_4 e Ex_5 são complementares, porém o Ex_4 tem um comportamento de estudo de caso. Ambos os experimentos alcançaram resultados satisfatórios no que diz respeito à precisão das classes. Contudo, novas investigações podem ser conduzidas para refinar as relações, principalmente na exploração do KG. Um caminho viável é realizar inferências sobre os recursos do IDEA-C2-KG e utilizá-las para enriquecer as relações contextuais nos próprios textos do corpus com o objetivo de evoluir o ciclo iterativo de ajuste fino do IDEA-C2-LM, assemelhado ao aprendizado por reforço. Ainda assim, as evidências indicam que a hipótese de combinar as abordagens DD e TD foi uma estratégia eficaz para a construção de DM. Por um lado, a interação com o IDEA-C2-LM é limitada em função de suas características subsimbólicas. Por outro lado, o IDEA-C2-LM demonstra alta capacidade para sugerir um conjunto amplo de entidades e relações a partir dos textos submetidos, tarefa que seria mais difícil para um ser humano. Essa capacidade possibilita que o usuário avalie e explore, no âmbito do IDEA-C2-KG, outras entidades e

[24] https://github.com/explosion/spacy-llm
[25] https://huggingface.co/maritaca-ai/sabia-7b

relações existentes, agora empregando sua cognição e capacidade analítica, i.e., elementos característicos da abordagem TD.

Finalmente, no experimento Ex_6, a incorporação da taxonomia do MAISC^2 na abordagem IDEA-C2 foi oportuna no sentido de torná-la uma abordagem híbrida (*single* e *multicategory*). Os resultados alcançados utilizando a abordagem *multicategory* são promissores, principalmente na tarefa de reconhecimento de entidades nos textos. Contudo, é necessário aperfeiçoar a tarefa de extração de relações com o intuito de entregar melhores resultados inferenciais na interação com IDEA-C2-LM. Além disso, como a abordagem *multicategory* restringe o conjunto de entidades e relações de acordo com as categorias predefinidas, a geração de KG é limitada a um escopo menor, sendo possível criar instâncias distintas do IDEA-C2-KG^n.

Entretanto, neste experimento, as múltiplas instâncias do IDEA-C2-KG não foram objeto de exploração. Assim, novas investigações no experimento podem ser conduzidas na direção de recuperar as diferentes instâncias do IDEA-C2-KG com o objetivo de agregar um KG mais abrangente, utilizando, por exemplo, a técnica de enriquecimento de *datasets* (116). Dessa forma, é possível aperfeiçoar a identificação de classes, bem como a capacidade inferencial com novas relações contextuais.

Nesta seção, foram apresentados os experimentos elaborados para testar e validar as hipóteses deste trabalho. A execução dos experimentos conduziram a pesquisa a obter resultados e refinamentos necessários à sua continuidade. Com base nisso, no próximo capítulo são discutidos os resultados alcançados e os próximos desafios que podem ser explorados futuramente.

7 CONCLUSÃO E CONSIDERAÇÕES FINAIS

Este trabalho de pesquisa apresentou o IDEA-C2, uma abordagem híbrida e supervisionada que combina características das abordagens Theory-Driven (TD) e Data-Driven (DD) para dar suporte à elaboração de modelos de domínio. O IDEA-C2 utiliza o metamodelo Command and Control Relations Model (C2RM), constituído de construtos de aplicação genéricas e específicas no domínio de C2, combinando Recursos Semânticos (RS) com um método supervisionado à distância para anotar um corpus. Essa característica proporciona flexibilidade à abordagem, uma vez que as categorias das entidades do domínio não são predeterminadas (*singlecategory*). Por outro lado, como a abordagem IDEA-C2 é híbrida, ela também permite a incorporação de vocabulários ou taxonomias ao C2RM, definindo um conjunto de categorias específicas (*multicategory*).

Como os Modelos de Linguagem (ML) pré-treinados são subsimbólicos, característico da abordagem DD, sua aplicação a domínios específicos é, por vezes, questionada. No IDEA-C2, como alternativa, o ajuste fino do ML pré-treinado é realizado, usando um corpus anotado e curado por usuários especialistas, com o apoio de um pré-anotador semiautomatizado que minimiza a intervenção humana. Em nossos experimentos, o pré-anotador reduziu o esforço de anotação, alcançando, respectivamente, uma precisão de 95% nas entidades e 76% nas relações. Esses esforços culminaram na geração de um ML ajustado, nas tarefas NER e RE, denominado IDEA-C2-LM. Os resultados do IDEA-C2-LM são promissores, pois alcançaram 86% de precisão e cobertura na abordagem *singlecategory*, bem como 88% de precisão e 86% de cobertura na abordagem *multicategory*, como apresentado nas seções 6.1, 6.2, 6.3 e 6.6 através dos experimentos Ex_1, Ex_2, Ex_3 e Ex_6.

Apesar dos resultados do IDEA-C2 serem promissores, a extração de conhecimento sobre o IDEA-C2-LM é restrita às respostas e ao raciocínio do ML, limitando as inferências ao ajuste fino realizado sobre ML ajustado. Além disso, algumas dessas inferências não são diretamente explicáveis, pois os ML são caixas-pretas e suas decisões resultam de interações matemáticas subsimbólicas. Muitas vezes, o raciocínio de um ML é complexo em função dos bilhões de parâmetros, dificultando o acesso e a explicabilidade de suas decisões. Como alternativa, este trabalho permitiu aos especialistas do domínio realizarem inferências sobre o IDEA-C2-LM. Essas inferências serviram de base de conhecimento e foram estruturadas em um grafo RDF, denominado IDEA-C2-KG. Além dessas inferências, o IDEA-C2-KG foi constituído de um conjunto de dados utilizado no ajuste fino do IDEA-C2-LM, totalizando mais de 5 mil entidades e 30 mil relações, disponível para os especialistas do domínio realizarem consultas exploratórias sobre seus recursos.

Os resultados das interações com o IDEA-C2-LM e IDEA-C2-KG são promissores,

principalmente quando aplicados no suporte ao desenvolvimento de um Modelo de Domínio (DM), evidenciados nos experimentos Ex_4 e Ex_5, abordado nas seções 6.4 e 6.5. Ambos os experimentos foram executados em ambientes controlados e envolveram pessoas com formações e níveis de experiência heterogêneos. Um dos experimentos avaliou os objetos dos DMs elaborados por um grupo de usuários, que utilizou uma abordagem tradicional, comparando com outro grupo que foi apoiado pelo IDEA-C2. Nesse experimento, os resultados foram favoráveis em 40% dos casos analisados para o grupo apoiado por IDEA-C2, alcançando valores superiores a 50% de acerto entre classes e relacionamentos. Cabe destacar que os resultados alcançados no experimento Ex_5 foram influenciados pelas interações e decisões de modelagem de cada participante. Nesse sentido, não há garantias de que esse experimento alcançasse os mesmos resultados com outros participantes, principalmente em função de os modelos de domínio construídos estarem sujeitos a diferentes visões sobre fatos expressos no minimundo proposto no experimento.

O protótipo IDEA-C2-Tool foi desenvolvido a partir do macroprocesso do IDEA-C2 com o objetivo de executar os experimentos e avaliar os resultados alcançados. Além disso, o PREAnoTeTool, um subproduto do IDEA-C2-Tool, foi criado a partir da funcionalidade de anotação do corpus com o objetivo de pré-anotar os textos de modo semiautomatizado, minimizando, assim, a necessidade de intervenção humana na curadoria. Em resumo, o IDEA-C2-Tool recupera o corpus curado com o intuito de gerar o Modelo de Linguagem (ML) ajustado ao domínio, assim como o Knowledge Graph (KG) com os recursos de dados disponíveis para operações, consultas e inferências. Por fim, o IDEA-C2-Tool permite interações de usuários especialistas no domínio no apoio à elaboração de um DM.

Portanto, a partir dos experimentos e das evidências dos resultados alcançados, demonstrou-se que a abordagem IDEA-C2 contribuiu com os usuários especialistas na obtenção do conhecimento a partir da geração dos artefatos IDEA-C2-LM e IDEA-C2-KG. Por meio desses artefatos, a abordagem apoiou os especialistas do domínio na elaboração de um DM (IDEA-C2-DM). Dessa forma, os resultados apontam a utilidade do IDEA-C2 na aplicação dos experimentos, demonstrando a viabilidade da abordagem em estudos de casos distintos e favorecendo as análises sobre os recursos do KG. Ademais, a abordagem pode agilizar a construção de um modelo conceitual no domínio de C2, ou até apoiar a revisão de modelagens já existentes com o objetivo de aumentar a interoperabilidade entre os aplicativos de C2.

7.1 Contribuições

Este trabalho confirmou as hipóteses estabelecidas inicialmente através do desenvolvimento da abordagem IDEA-C2, assim como pela implementação dos protótipos, IDEA-C2-Tool e PREAnoTeTool, e sobre a avaliação dos resultados através da aplicação

em seis experimentos distintos que explorou diferentes estudos de casos. Por isso, é possível citar as contribuições a seguir:

- **C1**: Um processo semiautomatizado e supervisionado de geração de KG, apoiado por um ML ajustado ao domínio militar a partir de textos doutrinários com metacategorias de C2RM;
- **C2**: Um corpus anotado com base nas metacategorias de C2RM, disponível para treinar Modelos de Linguagem (ML) nas tarefas de NER e RE;
- **C3**: Um processo de triplificação para gerar o KG, baseado no grafo RDF, a partir das metacategorias de C2RM;
- **C4**: Um processo de pré-anotação baseado em regras heurísticas de expressão regular para minimizar a anotação e apoiar a curadoria do corpus;
- **C5**: Um processo cíclico de refinamento de ajuste de ML com base em novos textos;
- **C6**: Uma sistemática para apoiar a elaboração de Modelo de Domínio (DM) combinando as abordagens Data-Driven (DD) e Theory-Driven (TD) a partir das interações com IDEA-C2-LM e IDEA-C2-KG;
- **C7**: Implementação do processo da abordagem adaptável, configurável, aplicado ao contexto de C2 para realizar experimentos que permita validar e avaliar a abordagem;
- **C8**: Implementação de dois protótipos (PREAnoTeTool e IDEA-C2-Tool) baseado nos casos de uso do cenário de aplicação para realização de experimentos que permite validar e avaliar a abordagem;
- **C9**: Ajustes do ML combinando Recursos Semânticos (RS) com a incorporação de taxonomias e vocabulários aplicado ao domínio militar (*multicategory).*

7.2 Limitações, dificuldades encontradas e melhorias

O trabalho de pesquisa desenvolvido nesta tese atendeu o escopo e prazo definidos. Comumente, o trabalho apresenta algumas limitações que podem servir de objeto de estudo de outros trabalhos futuros. Apesar deste trabalho ter utilizado seis experimentos com objetivos distintos, é importante submeter o IDEA-C2 a outros estudos de casos a fim de avaliar seus resultados. Os protótipos desenvolvidos não levaram em consideração aspectos relacionados à performance tampouco a usabilidade. Além disso, questões associadas à interface gráfica da aplicação também não foram consideradas. Entretanto, o código fonte está disponível no Github[1] e pode ser adaptado e evoluído.

[1] <https://github.com/comp-ime-eb-br/S2C2-IME/tree/main/deliverables/idea-c2>

Durante o desenvolvimento deste trabalho uma das principais dificuldades foi a obtenção do corpus no domínio militar em língua portuguesa. Para suprir essa dificuldade inicial, definiu-se como um dos objetivos a criação do corpus, porém outro desafio é justamente definir quais textos poderiam compor o corpus. Assim, buscou-se textos representativos do domínio baseado nas doutrinas. Contudo, como é vasto o número de doutrinas, a seleção se deu por meio de amostras dos textos com base nos assuntos em estudo. Superados os desafios de criação do corpus, observou-se que um RS poderia contribuir com a técnica de supervisão à distância. No entanto, em um primeiro momento, não foram encontrados RS no domínio militar, como por exemplo uma ontologia. Para superar essa escassez, utilizou-se o Glossário de Termos do Exército Brasileiro (1) que elenca um conjuntos de termos e definições aplicadas ao domínio militar. Além disso, a seleção do Glossário foi oportuna em função da abrangência de assuntos e temas, diferentemente de uma única doutrina militar que é restrita a um contexto específico.

Por fim, outro aspecto relevante foi em relação ao ajuste fino dos ML. Inicialmente, foram aplicados esforços para estabelecer uma infraestrutura capaz de suportar altas demandas de processamento. Algumas tentativas foram realizadas em ambiente local para avaliar o desempenho através de pequenos experimentos. Como o resultado foi aquém das necessidades da pesquisa, buscou-se uma alternativa de solução em nuvem, principalmente pela praticidade e escalabilidade. Nesse caso, adotou-se o Google Colaboratory Pro, uma versão robusta e financiada que atendeu todas as demandas para realizar o ajuste fino dos ML, pois oferece recursos variados com GPUs de alta velocidade e boa capacidade de armazenamento e processamento.

7.3 Trabalhos futuros

Algumas das discussões sobre possíveis melhorias são apresentadas amplamente na seção 6.7. Contudo, destacamos alguns pontos relevantes que podem ser estendidos em pesquisas futuras. Como a abordagem limitou-se a extrair dos textos as entidades e relações atribuindo a cada uma delas uma categoria, poderia ser proposta a extração de textos anotados a partir da exploração das inferências no IDEA-C2-KG. A ideia central é criar um ciclo iterativo e incremental de refinamentos do IDEA-C2-LM de modo híbrido, i.e., partindo tanto do texto anotado quanto da exploração do KG.

Durante o desenvolvimento da pesquisa, o IDEA-C2 foi expandido para comportar a abordagem *multicategory* a partir da incorporação de um recurso semântico, impactando positivamente na anotação do corpus e na geração do IDEA-C2-LM. Entretanto, é possível expandir também o artefato IDEA-C2-KG com base nas multicategorias incorporadas, permitindo que sejam gerados KG distintos. Cada um desses KG pode ser utilizado para enriquecimento de *datasets* através de operação de interligação e *matching*. Outro ponto

importante a ser destacado é a possibilidade de utilização de IDEA-C2 em outros domínios, inclusive em prospecção tecnológica, ciência de materiais e ciência da computação através de textos dos trabalhos já referenciados. Oportunamente, cabe salientar que devem ser encontrados alguns desafios, principalmente envolvendo o recurso semântico de apoio à pré-anotação, bem como o mapeamento das regras de expressão regular.

Cabe destacar que o apoio à elaboração do IDEA-C2-DM ficou limitado à modelagem conceitual tradicional. Porém, é possível incorporar ao IDEA-C2-DM uma abordagem ontológica e bem fundamentada, por exemplo, através da ontologia de fundamentação UFO[2](179). Para tal, acreditamos que um caminho viável é investir no mapeamento entre o metamodelo C2RM e os fragmentos da UFO (Ontology of Endurants (UFO-A) e Ontology of Perdurants (UFO-B)), qualificando as instâncias do construto *entity* através da estrutura taxonômica que envolve os indivíduos existentes no tempo com todas as suas partes, por exemplo, com os *Endurants types* e *Perdurant types*.

Finalmente, um dos pilares desta tese é o apoio na anotação do corpus. Os resultados dos experimentos evidenciaram uma minimização do esforço de anotação com valores razoáveis. Contudo, há pouco tempo, a comunidade vem adotando o paradigma "LLM-as-a-Judge" que pode ser aplicado na abordagem IDEA-C2 para avaliar comparativamente os resultados da atividade de pré-anotação com o novo paradigma. Ao ser incorporado ao IDEA-C2, sugere-se validar as categorias utilizadas no corpus, a atribuição de escala de valores com os níveis de confiança entre as anotações e até o aperfeiçoamento do pré-anotador da abordagem.

7.4 Agradecimentos

Agradeço ao IME - Instituto Militar de Engenharia e à FINEP/DCT/FAPEB (nº 2904/20-01.20.0272.00) pelo apoio ao Projeto "Sistemas de Sistemas de Comando e Controle".

[2] https://nemo.inf.ufes.br/

REFERÊNCIAS

1 BRASIL. Glossário de termos e expressões para uso no Exército. *Exército. Estado-Maior*, Brasília, DF, 2018. Disponível em: <https://bdex.eb.mil.br/jspui/bitstream/1/1148/1/Gloss%c3%a1rio%20EB%202018.pdf>.

2 AVELINO, J.; ROSA, G.; DANON, G.; CORDEIRO, K.; CAVALCANTI, M. C. Preanote: Uma abordagem de anotação de corpus para o ajuste fino de large language model pré-treinado. In: *Anais do XXXIX Simpósio Brasileiro de Bancos de Dados*. Porto Alegre, RS, Brasil: SBC, 2024. p. 806–812. ISSN 2763-8979. Disponível em: <https://sol.sbc.org.br/index.php/sbbd/article/view/30750>.

3 SILVA, M. A. A. da. *Combinando técnica e doutrina por meio de conceitos ontológicos em proveito de sistemas de comunicações cognitivos para cenários operacionais militares*. 151 p. Tese (Doutorado) — Instituto Militar de Engenharia, Rio de Janeiro, 2023.

4 MOSAFI, F. F. D. S.; COSENZA, M. S.; VASCONCELOS, L. V. C.; PIRES, L. F.; DUARTE, J. C.; CAVALCANTI, M. C. R. Performance evaluation of monolithic and microservice architectures for natural language processing in command and control applications. *IEEE Access*, v. 13, p. 155447–155461, 2025.

5 AVELINO, J.; ROSA., G.; DANON., G.; CORDEIRO., K.; C. Cavalcanti., M. Knowledge graph generation from text using supervised approach supported by a relation metamodel: An application in c2 domain. In: INSTICC. *Proceedings of the 26th International Conference on Enterprise Information Systems - Volume 1: ICEIS*. [S.l.]: SciTePress, 2024. p. 281–288. ISBN 978-989-758-692-7. ISSN 2184-4992.

6 MARR, B. *20 fatos sobre a internet que você (provavelmente) não sabe*. 2015. <https://forbes.com.br/sem-categoria/2015/10/20-fatos-sobre-a-internet-que-voce-provavelmente-nao-sabe/>. (Acessado em 10/12/2022).

7 CODD, E. F. A relational model of data for large shared data banks. *Communications of the ACM*, ACM, v. 13, n. 6, p. 377–387, 1970.

8 DIALANI, P. *The Future of Data Revolution will be Unstructured Data*. 2020. Accessed: 2023-05-04. Disponível em: <https://www.analyticsinsight.net/the-future-of-data-revolution-will-be-unstructured-data/>.

9 LUAN, Y.; HE, L.; OSTENDORF, M.; HAJISHIRZI, H. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, 2018. p. 3219–3232. Disponível em: <https://aclanthology.org/D18-1360>.

10 KAUFMAN, D. *A inteligência artificial irá suplantar a inteligência humana?* ESTAÇÃO DAS LETRAS E CORES EDI, 2019. ISBN 9788568552902. Disponível em: <https://books.google.com.br/books?id=Fh-WDwAAQBAJ>.

11 FORBES. *ChatGPT tem recorde de crescimento da base de usuários*. 2023. <https://forbes.com.br/forbes-tech/2023/02/chatgpt-tem-recorde-de-crescimento-da-base-de-usuarios/>. (Acessado em 10/04/2024).

12 DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. BERT: Pre-training of deep bidirectional transformers for language understanding. 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)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019. p. 4171–4186. Disponível em: <https://aclanthology.org/N19-1423>.

13 RADFORD, A.; NARASIMHAN, K. Improving language understanding by generative pre-training. In: . [S.l.: s.n.], 2018.

14 FORBES. *Meta pagará quase US$ 15 Bi por participação na Scale AI*. 2025. <https://forbes.com.br/forbes-tech/2025/06/meta-pagara-quase-us-15-bi-por-participacao-na-scale-ai-diz-the-information/>. (Acessado em 10/09/2025).

15 GUIZZARDI, G.; PASTOR, O.; STOREY, V. C. Thinking Fast and Slow in Software Engineering . *IEEE Software*, IEEE Computer Society, Los Alamitos, CA, USA, v. 40, n. 06, p. 139–142, nov. 2023. ISSN 1937-4194. Disponível em: <https://doi.ieeecomputersociety.org/10.1109/MS.2023.3306132>.

16 SABA, W. S. Stochastic llms do not understand language: Towards symbolic, explainable and ontologically based llms. In: ALMEIDA, J. P. A.; BORBINHA, J.; GUIZZARDI, G.; LINK, S.; ZDRAVKOVIC, J. (Ed.). *Conceptual Modeling*. Cham: Springer Nature Switzerland, 2023. p. 3–19. ISBN 978-3-031-47262-6.

17 SHANI, C.; SOFFER, L.; JURAFSKY, D.; LECUN, Y.; SHWARTZ-ZIV, R. *From Tokens to Thoughts: How LLMs and Humans Trade Compression for Meaning*. 2025. Disponível em: <https://arxiv.org/abs/2505.17117>.

18 YANG, J.; HAN, S. C.; POON, J. A survey on extraction of causal relations from natural language text. *Knowledge and Information Systems*, Springer, v. 64, n. 5, p. 1161–1186, 2022.

19 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.; ZIMMERMANN, A. Knowledge graphs. *ACM Computing Surveys*, Association for Computing Machinery, New York, NY, USA, v. 54, n. 4, jul 2021. ISSN 0360-0300. Disponível em: <https://doi.org/10.1145/3447772>.

20 NOY, N.; GAO, Y.; JAIN, A.; NARAYANAN, A.; PATTERSON, A.; TAYLOR, J. Industry-scale knowledge graphs: Lessons and challenges: Five diverse technology companies show how it's done. *Queue*, Association for Computing Machinery, New York, NY, USA, v. 17, n. 2, p. 48–75, apr 2019. ISSN 1542-7730. Disponível em: <https://doi.org/10.1145/3329781.3332266>.

21 LEE, J.; YOON, W.; KIM, S.; KIM, D.; KIM, S.; SO, C. H.; KANG, J. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, v. 36, n. 4, p. 1234–1240, 09 2019. ISSN 1367-4803. Disponível em: <https://doi.org/10.1093/bioinformatics/btz682>.

22 GUPTA, T.; ZAKI, M.; KRISHNAN, N. M. A.; Mausam. Matscibert: A materials domain language model for text mining and information extraction. *npj Computational Materials*, v. 8, n. 1, p. 102, May 2022. ISSN 2057-3960. Disponível em: <https://doi.org/10.1038/s41524-022-00784-w>.

23 ROSA, G. F.; AVELINO, J. O.; CAVALCANTI, M. C.; DUARTE, J. C. TForMIX: A method that combines LLM and Multidimensional Modeling for Technological Foresight. *IEEE Access*, v. 13, p. 153320–153339, 2025.

24 WESTON, L.; TSHITOYAN, V.; DAGDELEN, J.; KONONOVA, O.; TREWARTHA, A.; PERSSON, K. A.; CEDER, G.; JAIN, A. Named entity recognition and normalization applied to large-scale information extraction from the materials science literature. *Journal of Chemical Information and Modeling*, v. 59, n. 9, p. 3692–3702, 2019. PMID: 31361962. Disponível em: <https://doi.org/10.1021/acs.jcim.9b00470>.

25 LIU, P.; QIAN, L.; ZHAO, X.; TAO, B. The construction of knowledge graphs in the aviation assembly domain based on a joint knowledge extraction model. *IEEE Access*, v. 11, p. 26483–26495, 2023.

26 ZHAO, Q.; HUANG, H.; DING, H. Study on military regulations knowledge construction based on knowledge graph. In: *2021 7th International Conference on Big Data and Information Analytics (BigDIA)*. [S.l.: s.n.], 2021. p. 180–184.

27 CHAUDHRI, V. K.; CHENG, B.; OVERTHOLTZER, A.; ROSCHELLE, J.; SPAULDING, A.; CLARK, P.; GREAVES, M.; GUNNING, D. Inquire biology: A textbook that answers questions. *AI Magazine*, v. 34, n. 3, p. 55–72, 2013.

28 BBC. *Os milhares de trabalhadores em países pobres que abastecem programas de inteligência artificial como o ChatGPT*. 2023. <https://www.bbc.com/portuguese/articles/c3gze230pj1o>. (Acessado em 02/06/2024).

29 ZHOU, J.; LI, X.; WANG, S.; SONG, X. Ner-based military simulation scenario development process. *The Journal of Defense Modeling and Simulation*, SAGE Publications Sage UK: London, England, v. 20, n. 4, p. 563–575, 2022.

30 FRIES, J. A.; STEINBERG, E.; KHATTAR, S.; FLEMING, S. L.; POSADA, J.; CALLAHAN, A.; SHAH, N. H. Ontology-driven weak supervision for clinical entity classification in electronic health records. *Nature communications*, Nature Publishing Group UK London, v. 12, n. 1, p. 2017, 2021.

31 MINTZ, M.; BILLS, S.; SNOW, R.; JURAFSKY, D. 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*. Suntec, Singapore: Association for Computational Linguistics, 2009. p. 1003–1011. Disponível em: <https://aclanthology.org/P09-1113>.

32 SANH, V.; DEBUT, L.; CHAUMOND, J.; WOLF, T. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

33 LIU, Y.; OTT, M.; GOYAL, N.; DU, J.; JOSHI, M.; CHEN, D.; LEVY, O.; LEWIS, M.; ZETTLEMOYER, L.; STOYANOV, V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

34 BELTAGY, I.; LO, K.; COHAN, A. SciBERT: A pretrained language model for scientific text. In: INUI, K.; JIANG, J.; NG, V.; WAN, X. (Ed.). *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, 2019. p. 3615–3620. Disponível em: <https://aclanthology.org/D19-1371>.

35 ROSTAM, Z. R. K.; KERTéSZ, G. Fine-tuning large language models for scientific text classification: A comparative study. In: *2024 IEEE 6th International Symposium on Logistics and Industrial Informatics (LINDI)*. [S.l.: s.n.], 2024. p. 000233–000238.

36 SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. Bertimbau: Pretrained bert models for brazilian portuguese. In: CERRI, R.; PRATI, R. C. (Ed.). *Intelligent Systems*. Cham: Springer International Publishing, 2020. p. 403–417. ISBN 978-3-030-61377-8.

37 SANTOS, L.; BIANCHI, R.; COSTA, A. Finbert-pt-br: Análise de sentimentos de textos em português do mercado financeiro. In: *Anais do II Brazilian Workshop on Artificial Intelligence in Finance*. Porto Alegre, RS, Brasil: SBC, 2023. p. 144–155. ISSN 0000-0000.

38 SILVEIRA, R.; PONTE, C.; ALMEIDA, V.; PINHEIRO, V.; FURTADO, V. Legalbert-pt: A pretrained language model for the brazilian portuguese legal domain. In: *Anais da XII Brazilian Conference on Intelligent Systems*. Porto Alegre, RS, Brasil: SBC, 2023. p. 268–282. ISSN 2643-6264. Disponível em: <https://sol.sbc.org.br/index.php/bracis/article/view/28420>.

39 NEWS, D. *Project Maven to Deploy Computer Algorithms to War Zone by Year's End*. 2017. <https://www.war.gov/News/News-Stories/Article/Article/1254719/project-maven-to-deploy-computer-algorithms-to-war-zone-by-years-end/>. (Acessado em 09/04/2024).

40 CABALLERO, W. N.; JENKINS, P. R. On large language models in national security applications. *Stat*, v. 14, n. 2, p. e70057, 2025. E70057 sta4.70057. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/sta4.70057>.

41 LI, L.; ZHANG, H.; LI, C.; YOU, H.; CUI, W. Evaluation on chatgpt for chinese language understanding. *Data Intelligence*, v. 5, n. 4, p. 885–903, 11 2023. ISSN 2641-435X. Disponível em: <https://doi.org/10.1162/dint_a_00232>.

42 SÁ, H. A.; GIRARDI, R.; DUARTE, J. C.; GALDINO, J. F. Tendências da inteligência artificial aplicada à defesa: forças, fraquezas, oportunidades e ameaças para o brasil. *Boletim de Conjuntura (BOCA)*, v. 21, n. 62, p. 01–33, 2025.

43 BRASIL. Portaria nº 4.617, de 04 de abril de 2021 do MCTIC. *Ministério da Ciência, Tecnologia e Inovações*, Brasília, DF, 2021. Disponível em: <https://antigo.mctic.gov.br/mctic/opencms/legislacao/portarias/Portaria_MCTI_n_4617_de_06042021.html>.

44 BRASIL. Portaria nº 4.979, de 13 de julho de 2021 do MCTIC. *Ministério da Ciência, Tecnologia e Inovações*, Brasília, DF, 2021. Disponível em: <https://antigo.mctic.gov.br/mctic/opencms/legislacao/portarias/Portaria_MCTI_n_4979_de_13072021.html>.

45 CIêNCIA, T. e. I. Ministério da. *Estratégia Brasileira de Inteligência Artificial - EBIA*. 2025. <https://www.gov.br/mcti/pt-br/acompanhe-o-mcti/transformacaodigital/inteligencia-artificial>. (Acessado em 23/07/2025).

46 BRASIL. Relatório de gestão do ministério da defesa. *Ministério da Defesa*, Brasília, DF, 2024. Disponível em: <https://www.gov.br/defesapt-/pt-br/acesso-a-informacao/transparencia-e-prestacao-de-contas/relatorio-de-gestao/relatorio-de-gestao-2024/arquivos/relatorio_gestao_md_2024_novo-1.pdf>.

47 NEWS, D. *Governo Federal - Painel Estatístico de Pessoal*. 2025. <https://painel.pep.planejamento.gov.br/>. (Acessado em 09/09/2025).

48 DOMINGO, A.; PARMAR, M. Functional analysis of cyberspace operations. In: IEEE. *MILCOM 2018-2018 IEEE Military Communications Conference (MILCOM)*. [S.l.], 2018. p. 673–678.

49 BRASIL. Lei nº 13.954, de 16 de dezembro de 2019 - dispõe sobre a reestruturação das carreiras militares. *Ministério da Defesa*, Brasília, DF, 2019. Disponível em: <https://www.planalto.gov.br/ccivil_03/_ato2019-2022/2019/lei/l13954.htm>.

50 BRASIL. Política Nacional de Defesa. *Ministério da Defesa*, Brasília, DF, 2020. Disponível em: <https://www.gov.br/defesa/pt-br/assuntos/copy_of_estado-e-defesa/pnd_end_congresso_1.pdf>.

51 BRASIL. Portaria nº 1.122, de 19 de março de 2020 do MCTIC. *Ministério da Ciência, Tecnologia e Inovações*, Brasília, DF, 2020. Disponível em: <https://www.in.gov.br/en/web/dou/-/portaria-n-1.122-de-19-de-marco-de-2020-249437397>.

52 BRASIL. Estratégia Brasileira de Inteligência Artificial (EBIA). *Ministério da Ciência, Tecnologia e Inovações*, Brasília, DF, 2020. Disponível em: <https://www.gov.br/mcti/pt-br/acompanhe-o-mcti/transformacaodigital/arquivosinteligenciaartificial/ebia-documento_referencia_4-979_2021.pdf>.

53 BRASIL. Plano Etratégico da Marinha (PEM 2040) 2020 - 2040. *Marinha do Brasil*, Brasília, DF, 2020. Disponível em: <https://www.marinha.mil.br/sites/all/modules/pub_pem_2040/book.html>.

54 HAN, X.; ZHU, H.; YU, P.; WANG, Z.; YAO, Y.; LIU, Z.; SUN, M. FewRel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, 2018. p. 4803–4809. Disponível em: <https://aclanthology.org/D18-1514>.

55 KONCEL-KEDZIORSKI, R.; BEKAL, D.; LUAN, Y.; LAPATA, M.; HAJISHIRZI, H. Text Generation from Knowledge Graphs with Graph Transformers. 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)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019. p. 2284–2293. Disponível em: <https://aclanthology.org/N19-1238>.

56 LUAN, Y.; WADDEN, D.; HE, L.; SHAH, A.; OSTENDORF, M.; HAJISHIRZI, H. A general framework for information extraction using dynamic span graphs. 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)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019. p. 3036–3046. Disponível em: <https://aclanthology.org/N19-1308>.

57 SOARES, L. B.; FITZGERALD, N.; LING, J.; KWIATKOWSKI, T. Matching the blanks: Distributional similarity for relation learning. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, 2019. p. 2895–2905. Disponível em: <https://aclanthology.org/P19-1279>.

58 ALI, M.; SPECK, R.; ZAHERA, H. M.; SALEEM, M.; MOUSSALLEM, D.; NGOMO, A.-C. N. Multilingual relation extraction: A survey. *IEEE Access*, v. 13, p. 151907–151933, 2025.

59 SOMMERVILLE, I. *Software Engineering*. 10th. ed. [S.l.]: Pearson, 2015. ISBN 0133943038.

60 YOURDON, E. *Análise estruturada moderna*. [S.l.]: Campus Rio de Janeiro, 1990.

61 CASELI, H. M.; NUNES, M. G. V. (Ed.). *Processamento de Linguagem Natural: Conceitos, Técnicas e Aplicações em Português*. [S.l.]: BPLN, 2023. <https://brasileiraspln.com/livro-pln>. ISBN 978-65-00-80693-9.

62 TOUVRON, H.; LAVRIL, T.; IZACARD, G.; MARTINET, X.; LACHAUX, M.-A.; LACROIX, T.; ROZIèRE, B.; GOYAL, N.; HAMBRO, E.; AZHAR, F.; RODRIGUEZ, A.; JOULIN, A.; GRAVE, E.; LAMPLE, G. *LLaMA: Open and Efficient Foundation Language Models*. 2023. Disponível em: <https://arxiv.org/abs/2302.13971>.

63 ELMASRI, R.; NAVATHE, S. *Fundamentals of Database Systems*. 7th. ed. USA: Pearson, 2015. ISBN 0133970779, 9780136086208.

64 RUSSELL, S.; NORVIG, P. *Artificial Intelligence: A Modern Approach*. 3. ed. [S.l.]: Prentice Hall, 2010.

65 GéRON, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. 3rd. ed. Sebastopol, CA, USA: O'Reilly Media, 2022. ISBN 978-1-098-10806-9. Disponível em: <https://www.oreilly.com/library/view/hands-on-machine-learning/9781098108069/>.

66 BRASIL. Doutrina Militar Terrestre. *Exército. Estado-Maior*, Brasília, DF, 2014. Disponível em: <https://bdex.eb.mil.br/jspui/bitstream/123456789/93/5/REVOGADO-EB20-MF-10.102.pdf>.

67 LI, J.; SUN, A.; HAN, J.; LI, C. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, v. 34, n. 1, p. 50–70, 2022.

68 RYEN, V.; SOYLU, A.; ROMAN, D. Building semantic knowledge graphs from (semi-) structured data: a review. *Future Internet*, MDPI, v. 14, n. 5, p. 129, 2022.

69 LEHNERT, W.; CARDIE, C.; FISHER, D.; RILOFF, E.; WILLIAMS, R. University of Massachusetts: Description of the CIRCUS system as used for MUC-3. In: *Third Message Uunderstanding Conference (MUC-3): Proceedings of a Conference Held in San Diego, California, May 21-23, 1991*. [s.n.], 1991. Disponível em: <https://aclanthology.org/M91-1033>.

70 GRISHMAN, R. Information extraction: Techniques and challenges. In: PAZIENZA, M. T. (Ed.). *Information Extraction A Multidisciplinary Approach to an Emerging Information Technology*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997. p. 10–27. ISBN 978-3-540-69548-6.

71 FACELI K.; LORENA, A.; GAMA, J.; CARVALHO, A. *Inteligência Artificial - Uma Abordagem de Aprendizado de Máquina. Edição 1.* [S.l.]: LTC Editora, 2015. 378 f., 2015.

72 STENETORP, P.; PYYSALO, S.; TOPIĆ, G.; OHTA, T.; ANANIADOU, S.; TSUJII, J. brat: a web-based tool for NLP-assisted text annotation. In: *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*. Avignon, France: Association for Computational Linguistics, 2012. p. 102–107. Disponível em: <https://aclanthology.org/E12-2021>.

73 BAVISKAR, D.; AHIRRAO, S.; KOTECHA, K. Multi-layout invoice document dataset (midd): A dataset for named entity recognition. *Data*, v. 6, n. 7, 2021. ISSN 2306-5729. Disponível em: <https://www.mdpi.com/2306-5729/6/7/78>.

74 TKACHENKO, M.; MALYUK, M.; HOLMANYUK, A.; LIUBIMOV, N. *Label Studio: Data labeling software*. 2020–2022. Open source software available from https://github.com/heartexlabs/label-studio. Disponível em: <https://github.com/heartexlabs/label-studio>.

75 NAKAYAMA, H.; KUBO, T.; KAMURA, J.; TANIGUCHI, Y.; LIANG, X. *doccano: Text Annotation Tool for Human*. 2018. Software available from https://github.com/doccano/doccano. Disponível em: <https://github.com/doccano/doccano>.

76 AI, E. *Prodigy*. 2023. 10 ago. de 2023. Disponível em: <https://prodi.gy/>.

77 JSON.ORG. *JSON Lines Documentation for the JSON Lines text file format*. 2023. 31 nov. de 2022. Disponível em: <https://jsonlines.org/>.

78 SMIRNOVA, A.; CUDRé-MAUROUX, P. Relation extraction using distant supervision: A survey. *ACM Comput. Surv.*, Association for Computing Machinery, New York, NY, USA, v. 51, n. 5, nov 2018. ISSN 0360-0300. Disponível em: <https://doi.org/10.1145/3241741>.

79 RATNER, A.; BACH, S. H.; EHRENBERG, H.; FRIES, J.; WU, S.; RÉ, C. Snorkel: Rapid training data creation with weak supervision. *The VLDB Journal*, Springer, v. 29, n. 2-3, p. 709–730, 2020.

80 ABHISHEK, G.; INGOLE, H.; LATURIA, P.; DORNA, V.; MAHESHWARI, A.; RAMAKRISHNAN, G.; IYER, R. SPEAR : Semi-supervised data programming in python. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Abu Dhabi, UAE: Association for Computational Linguistics, 2022. p. 121–127. Disponível em: <https://aclanthology.org/2022.emnlp-demos.12>.

81 DONG, H.; SUÁREZ-PANIAGUA, V.; ZHANG, H.; WANG, M.; CASEY, A.; DAVIDSON, E.; CHEN, J.; ALEX, B.; WHITELEY, W.; WU, H. Ontology-driven and weakly supervised rare disease identification from clinical notes. *BMC Medical Informatics and Decision Making*, BioMed Central, v. 23, n. 1, p. 1–17, 2023.

82 EUZENAT, J.; SHVAIKO, P. et al. *Ontology matching*. [S.l.]: Springer, 2007. v. 18.

83 VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L.; POLOSUKHIN, I. Attention is all you need. In: GUYON, I.; LUXBURG, U. V.; BENGIO, S.; WALLACH, H.; FERGUS, R.; VISHWANATHAN, S.; GARNETT, R. (Ed.). *Advances in Neural Information Processing Systems*. [S.l.]: Curran Associates, Inc., 2017. v. 30.

84 GAO, T.; HAN, X.; ZHU, H.; LIU, Z.; LI, P.; SUN, M.; ZHOU, J. FewRel 2.0: Towards more challenging few-shot relation classification. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, 2019. p. 6250–6255. Disponível em: <https://aclanthology.org/D19-1649>.

85 HAN, X.; GAO, T.; YAO, Y.; YE, D.; LIU, Z.; SUN, M. Opennre: An open and extensible toolkit for neural relation extraction. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*. [S.l.: s.n.], 2019. p. 169–174.

86 YAO, Y.; YE, D.; LI, P.; HAN, X.; LIN, Y.; LIU, Z.; LIU, Z.; HUANG, L.; ZHOU, J.; SUN, M. DocRED: A large-scale document-level relation extraction dataset. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, 2019. p. 764–777. Disponível em: <https://aclanthology.org/P19-1074>.

87 LEVY, O.; SEO, M.; CHOI, E.; ZETTLEMOYER, L. Zero-shot relation extraction via reading comprehension. In: *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*. Vancouver, Canada: Association for Computational Linguistics, 2017. p. 333–342. Disponível em: <https://aclanthology.org/K17-1034>.

88 SPALA, S.; MILLER, N. A.; YANG, Y.; DERNONCOURT, F.; DOCKHORN, C. DEFT: A corpus for definition extraction in free- and semi-structured text. In: *Proceedings of the 13th Linguistic Annotation Workshop*. Florence, Italy: Association for Computational Linguistics, 2019. p. 124–131. Disponível em: <https://aclanthology.org/W19-4015>.

89 SPALA, S.; MILLER, N.; DERNONCOURT, F.; DOCKHORN, C. SemEval-2020 task 6: Definition extraction from free text with the DEFT corpus. In: *Proceedings of the Fourteenth Workshop on Semantic Evaluation*. Barcelona (online): International Committee for Computational Linguistics, 2020. p. 336–345. Disponível em: <https://aclanthology.org/2020.semeval-1.41>.

90 LECUN, Y.; BENGIO, Y.; AL et. Deep learning. *Nature*, Nature Publishing Group, v. 521, p. 436–444, 2015.

91 MIWA, M.; BANSAL, M. End-to-end relation extraction using LSTMs on sequences and tree structures. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, 2016. p. 1105–1116. Disponível em: <https://aclanthology.org/P16-1105>.

92 SALTON, G.; WONG, A.; YANG, C. S. A vector space model for automatic indexing. *Commun. ACM*, Association for Computing Machinery, New York, NY, USA, v. 18, n. 11,

p. 613–620, nov 1975. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/361219.361220>.

93 GOLDBERG, Y. A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research*, v. 57, p. 345–420, 2016.

94 MIKOLOV, T.; CHEN, K.; CORRADO, G.; DEAN, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

95 PETERS, M. E.; NEUMANN, M.; IYYER, M.; GARDNER, M.; CLARK, C.; LEE, K.; ZETTLEMOYER, L. Deep contextualized word representations. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, 2018. p. 2227–2237. Disponível em: <https://aclanthology.org/N18-1202>.

96 GUIZZARDI, G.; ALMEIDA, J. P.; GUIZZARDI, R. S.; BARCELLOS, M. P.; FALBO, R. Ontologias de fundamentação, modelagem conceitual e interoperabilidade semântica. *Applied Ontology*, , p. 1–10, 2011.

97 FOWLER, M. *Analysis patterns: reusable object models*. [S.l.]: Addison-Wesley Professional, 1997.

98 CHEN, P. P.-S. The entity-relationship model—toward a unified view of data. *ACM transactions on Database Systems (TODS)*, Association for Computing Machinery, New York, NY, USA, v. 1, n. 1, p. 9–36, mar. 1976. ISSN 0362-5915. Disponível em: <https://doi.org/10.1145/320434.320440>.

99 LARMAN, C. *Utilizando UML e Padrões*. Bookman, 2007. ISBN 9788577800476. Disponível em: <https://books.google.com.br/books?id=hzl2tmT8QkUC>.

100 Elmasri, R.; Fu, J.; Ji, F. Multi-level conceptual modeling for biomedical data and ontologies integration. In: *Twentieth IEEE International CBMS*. [S.l.: s.n.], 2007. p. 589–594.

101 ROUCES, J.; MELO, G. de; HOSE, K. Complex schema mapping and linking data: Beyond binary predicates. In: . [s.n.], 2016. Disponível em: <http://ceur-ws.org/Vol-1593/#article-05>.

102 BEEK, W.; SCHLOBACH, S.; HARMELEN, F. van. A contextualised semantics for owl:sameas. In: *The Semantic Web. Latest Advances and New Domains*. [S.l.: s.n.], 2016. ISBN 978-3-319-34129-3.

103 KENT, W. Fact-based data analysis and design. *Journal of Systems and Software*, Elsevier, v. 4, n. 2-3, p. 99–121, 1984.

104 KENT, W. *Data and Reality: A Timeless Perspective on Perceiving and Managing Information*. [S.l.]: Technics publications, 2012.

105 MULLER, R. J. *Database design for smarties: using UML for data modeling*. [S.l.]: Morgan Kaufmann, 1999.

106 BRAMBILLA, M.; CABOT, J.; WIMMER, M. *Model-Driven Software Engineering in Practice: Second Edition*. 2nd. ed. [S.l.]: Morgan Claypool Publishers, 2017. ISBN 1627057080.

107 GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J. *Design Patterns: Elements of Reusable Object-Oriented Software*. USA: Addison-Wesley Longman Publishing Co., Inc., 1995. ISBN 0201633612.

108 OMG, O. M. G. *Meta Object Facility*. 2023. Meta Object Facility. Disponível em: <https://www.omg.org/spec/MOF/2.5.1/PDF>. Acesso em: 15 nov 2023.

109 FIDALGO, R. D. N.; SOUZA, E. M. D.; ESPAÑA, S.; CASTRO, J. B. D.; PASTOR, O. Eermm: A metamodel for the enhanced entity-relationship model. In: ATZENI, P.; CHEUNG, D.; RAM, S. (Ed.). *Conceptual Modeling*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. p. 515–524. ISBN 978-3-642-34002-4.

110 FIDALGO, R. N.; ALVES, E.; ESPAÑA, S.; CASTRO, J.; PASTOR, O. Metamodeling the enhanced entity-relationship model. *Journal of Information and Data Management*, v. 4, n. 3, p. 406–406, 2013.

111 FIDALGO, R. N.; SILVA, E. A. Consistent design of relational databases using eercase. *Journal of Information and Data Management*, v. 13, n. 5, 2022.

112 SADALAGE, P. J.; FOWLER, M. *NoSQL Essencial: Um guia conciso para o mundo emergente da persistência poliglota*. [S.l.]: Novatec Editora, 2013.

113 HITZLER, P.; KRÖTZSCH, M.; PARSIA, B.; PATEL-SCHNEIDER, P. F.; RUDOLPH, S. et al. Owl 2 web ontology language primer. *W3C recommendation*, v. 27, n. 1, p. 123, 2009.

114 ANGLES, R.; ARENAS, M.; BARCELÓ, P.; HOGAN, A.; REUTTER, J.; VRGOČ, D. Foundations of modern query languages for graph databases. *ACM Computing Surveys (CSUR)*, ACM New York, NY, USA, v. 50, n. 5, p. 1–40, 2017.

115 W3C-CONSORTIUM, W. W.-W. W. *W3C Recommendation - Web Semântica*. 2017. 0. Disponível em: <http://www.w3c.br/Padroes/WebSemantica>.

116 SHERIF, M. A.; NGOMO, A.-C. N.; LEHMANN, J. Automating rdf dataset transformation and enrichment. In: SPRINGER. *European Semantic Web Conference*. [S.l.], 2015. p. 371–387.

117 BRASIL. Doutrina para o Sistema Militar de Comando e Controle. *Ministério da Defesa*, Brasília, DF, 2015. Disponível em: <https://www.gov.br/defesa/pt-br/arquivos/doutrina_militar/lista_de_publicacoes/md31a_ma_03a_douta_sismca_3a_eda_2015.pdf>.

118 BRASIL. Glossário das Forças Armadas. *Ministério da Defesa*, Brasília, DF, 2015. Disponível em: <https://bdex.eb.mil.br/jspui/bitstream/123456789/141/1/MD35_G01.pdf>.

119 DEFESA, M. da. *Publicações*. 2023. 17 jul. de 2014. Disponível em: <https://www.gov.br/defesa/pt-br/assuntos/estado-maior-conjunto-das-forcas-armadas/doutrina-militar/publicacoes-1/copy_of_publicacoes>.

120 BRASIL, M. do. *Publicações*. 2023. 17 jul. de 2014. Disponível em: <https://www.redebim.dphdm.mar.mil.br/>.

121 BRASILEIRO, E. *Publicações*. 2023. 17 jul. de 2014. Disponível em: <https://bdex.eb.mil.br/>.

122 BRASILEIRA, F. A. *Publicações*. 2023. 17 jul. de 2014. Disponível em: <https://www.sislaer.fab.mil.br/>.

123 BRASIL. Doutrina de Operações Conjuntas. *Ministério da Defesa*, Brasília, DF, 2020. Disponível em: <https://www.gov.br/defesa/pt-br/arquivos/legislacao/emcfa/publicacoes/doutrina/md30-m-01-vol-1-2a-edicao-2020-dou-178-de-15-set.pdf>.

124 BRASIL. Doutrina para o Sistema Militar de Comando e Controle. *Ministério da Defesa*, Brasília, DF, 2015. Disponível em: <https://www.gov.br/defesa/pt-br/arquivos/doutrina_militar/lista_de_publicacoes/md31a_ma_03a_douta_sismca_3a_eda_2015.pdf>.

125 BRASIL. Conceito de Operações do Sistema Militar de Comando e Controle. *Ministério da Defesa*, Brasília, DF, 2016. Disponível em: <https://bdex.eb.mil.br/jspui/bitstream/123456789/4760/1/EB20-MF-10.102.pdf>.

126 MOSAFI, F. F. D. S.; PIRES, L. F.; DUARTE, J. C.; CAVALCANTI, M. C. Towards a microservices architecture to support communication in c2 applications. In: *2024 19th Annual System of Systems Engineering Conference (SoSE)*. [S.l.: s.n.], 2024. p. 227–232.

127 BRASIL. Decreto de 28 de julho de 2017. *Diário Oficial da República Federativa do Brasil*, Brasília, DF, 2017. Disponível em: <https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2017/dsn/dsn14485.htm>.

128 BEZERRA, A. A. *O emprego de centro de adestramento nas certificações das OM da força terrestre*. 31 p. Dissertação (Monografia (Especialização em Ciências Militares)) — Escola de Comando e Estado-Maior do Exército, Rio de Janeiro, 2021. Disponível em: <https://bdex.eb.mil.br/jspui/handle/123456789/10002>.

129 RIBEIRO, M. C. *Adestramento de Estados-Maiores Conjuntos com Emprego de Simulação Construtiva*. 55 p. Dissertação (Monografia (Curso de Altos Estudos de Política e Estratégia (CAEPE))) — Escola Superior de Guerra, Rio de Janeiro, 2016. Disponível em: <https://repositorio.esg.br/bitstream/123456789/1114/1/TCC\%20MARCELO\%20CARVALHO\%20RIBEIRO.pdf>.

130 DONEDA, A. L. C.; OLIVEIRA, J. C. de. Helicopter visual signaling simulation: Integrating vr and ml into a low-cost solution to optimize brazilian navy training. In: *2020 22nd Symposium on Virtual and Augmented Reality (SVR)*. [S.l.: s.n.], 2020. p. 434–442.

131 WANG, Q.; YAVUZ, S.; LIN, X. V.; JI, H.; RAJANI, N. Stage-wise fine-tuning for graph-to-text generation. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop*. Online: Association for Computational Linguistics, 2021. p. 16–22. Disponível em: <https://aclanthology.org/2021.acl-srw.2>.

132 NUNDLOLL, V.; SMAIL, R.; STEVENS, C.; BLAIR, G. Automating the extraction of information from a historical text and building a linked data model for the domain of ecology and conservation science. *Heliyon*, Elsevier, p. e10710, 2022.

133 ZHANG, Q.; CHEN, Z.; PAN, H.; CARAGEA, C.; LATECKI, L. J.; DRAGUT, E. SciER: An entity and relation extraction dataset for datasets, methods, and tasks in scientific documents. In: AL-ONAIZAN, Y.; BANSAL, M.; CHEN, Y.-N. (Ed.). *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Miami, Florida, USA: Association for Computational Linguistics, 2024. p. 13083–13100. Disponível em: <https://aclanthology.org/2024.emnlp-main.726/>.

134 PAN, H.; ZHANG, Q.; DRAGUT, E.; CARAGEA, C.; LATECKI, L. J. DMDD: A large-scale dataset for dataset mentions detection. *Transactions of the Association for Computational Linguistics*, MIT Press, Cambridge, MA, v. 11, p. 1132–1146, 2023. Disponível em: <https://aclanthology.org/2023.tacl-1.64/>.

135 RIOS-ALVARADO, A. B.; MARTINEZ-RODRIGUEZ, J. L.; GARCIA-PEREZ, A. G.; GUERRERO-MELENDEZ, T. Y.; LOPEZ-AREVALO, I.; GONZALEZ-COMPEAN, J. L. Exploiting lexical patterns for knowledge graph construction from unstructured text in spanish. *Complex & Intelligent Systems*, Springer, v. 9, n. 2, p. 1281–1297, 2023.

136 DANG, L. D.; PHAN, U. T.; NGUYEN, N. T. Gena: A knowledge graph for nutrition and mental health. *Journal of Biomedical Informatics*, v. 145, p. 104460, 2023. ISSN 1532-0464. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1532046423001818>.

137 HEALTH, U. D. of; SERVICES, H. *NCBO BioPortal - Biomedical Ontology*. 2005. 10 jul. de 2005. Disponível em: <https://bioportal.bioontology.org/ontologies>.

138 AI, A. A. I. for. *scispacy: Modelos SpaCy para processamento de texto biomédico*. 2023. 20 nov. de 2023. Disponível em: <https://allenai.github.io/scispacy/>.

139 CONSORTIUM, F. *scispacy: Modelos SpaCy para processamento de texto biomédico*. 2015. 20 nov. de 2015. Disponível em: <https://foodon.org/>.

140 INSTITUTE, E. E. B. *Chemical Entities of Biological Interest (ChEBI)*. 2015. 20 nov. de 2015. Disponível em: <https://www.ebi.ac.uk/chebi/>.

141 COMPUTING, N. C. for B. *NCBO BioPortal - Biomedical Ontology*. 2005. 10 jul. de 2005. Disponível em: <https://bioportal.bioontology.org/ontologies>.

142 LI, J.; SUN, Y.; JOHNSON, R. J.; SCIAKY, D.; WEI, C.-H.; LEAMAN, R.; DAVIS, A. P.; MATTINGLY, C. J.; WIEGERS, T. C.; LU, Z. *Corpus de tarefas BioCreative V CDR: um recurso para extração de relações químicas com doenças*. 2020. 20 nov. de 2020. Disponível em: <https://paperswithcode.com/dataset/bc5cdr>.

143 TRAJANOSKA, M.; STOJANOV, R.; TRAJANOV, D. *Enhancing Knowledge Graph Construction Using Large Language Models*. 2023.

144 CABOT, P.-L. H.; NAVIGLI, R. Rebel: Relation extraction by end-to-end language generation. In: *Findings of the Association for Computational Linguistics: EMNLP 2021*. [S.l.: s.n.], 2021. p. 2370–2381.

145 PAROVIĆ, M.; LI, Z.; DU, J. Generating domain-specific knowledge graphs from large language models. In: CHE, W.; NABENDE, J.; SHUTOVA, E.; PILEHVAR, M. T. (Ed.). *Findings of the Association for Computational Linguistics: ACL 2025*. Vienna, Austria:

Association for Computational Linguistics, 2025. p. 11558–11574. ISBN 979-8-89176-256-5. Disponível em: <https://aclanthology.org/2025.findings-acl.602/>.

146 SILVEIRA, R.; CAVALCANTI, M. Método para rotular ligações semânticas na web de dados. In: *Anais do XXXV Simpósio Brasileiro de Bancos de Dados*. Porto Alegre, RS, Brasil: SBC, 2020. p. 49–60. ISSN 2763-8979. Disponível em: <https://sol.sbc.org.br/index.php/sbbd/article/view/13624>.

147 ZHU, X.; LI, H.; SU, T. Autonomous complex knowledge mining and graph representation through natural language processing and transfer learning. *Automation in Construction*, v. 155, p. 105074, 2023. ISSN 0926-5805. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0926580523003345>.

148 HSU, C.-C.; SANDFORD, B. A. The delphi technique: making sense of consensus. *Practical assessment, research, and evaluation*, University of Massachusetts Amherst Libraries, v. 12, n. 1, 2019.

149 TRAPPEY, A. J. C.; LIANG, C.-P.; LIN, H.-J. Using machine learning language models to generate innovation knowledge graphs for patent mining. *Applied Sciences*, v. 12, n. 19, 2022. ISSN 2076-3417. Disponível em: <https://www.mdpi.com/2076-3417/12/19/9818>.

150 LAN, Z.; CHEN, M.; GOODMAN, S.; GIMPEL, K.; SHARMA, P.; SORICUT, R. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019. Disponível em: <https://arxiv.org/abs/1909.11942>.

151 GROOTENDORST, M. Keybert: Minimal keyword extraction with bert. *arXiv preprint arXiv:2010.11965*, 2020. Disponível em: <https://arxiv.org/abs/2010.11965>.

152 REIMERS, N.; GUREVYCH, I. Sentence-bert: Sentence embeddings using siamese bert-networks. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. [s.n.], 2019. p. 3982–3992. Disponível em: <https://arxiv.org/abs/1908.10084>.

153 CHANTRAPORNCHAI, C.; TUNSAKUL, A. Information extraction tasks based on bert and spacy on tourism domain. v. 15, p. 108–122, Jan. 2021. Disponível em: <https://ph01.tci-thaijo.org/index.php/ecticit/article/view/228621>.

154 KIM, S.-Y.; GÖRZ, M.; GEISLER, S. Konda: An llm-based tool for semantic annotation and knowledge graph creation using ontologies for research data. In: *CEUR Workshop Proceedings*. [S.l.: s.n.], 2025. v. 4065.

155 LUCASSEN, G.; ROBEER, M.; DALPIAZ, F.; WERF, J. M. E. M. van der; BRINKKEMPER, S. Extracting conceptual models from user stories with visual narrator. *Requirements Engineering*, v. 22, n. 3, p. 339–358, Sep 2017. ISSN 1432-010X. Disponível em: <https://doi.org/10.1007/s00766-017-0270-1>.

156 SHWETA; SANYAL, R.; GHOSHAL, B. Automated class diagram elicitation using intermediate use case template. *IET Software*, v. 15, n. 1, p. 25–42, 2021. Disponível em: <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/sfw2.12010>.

157 RAMZAN, M.; SADIQI, G. S.; BASHIR, M. S.; RAZA, S.; BATOOL, A. A rule-based approach for automatic generation of class diagram from functional requirements using natural language processing and machine learning. *Journal of Computing & Biomedical Informatics*, v. 7, n. 02, 2024. Disponível em: <https://jcbi.org/index.php/Main/article/view/546>.

158 PROKOP, D.; STENCHLÁK, Š.; ŠKODA, P.; KLÍMEK, J.; NEČASKÝ, M. Enhancing domain modeling with pre-trained large language models: An automated assistant for domain modelers. In: MAASS, W.; HAN, H.; YASAR, H.; MULTARI, N. (Ed.). *Conceptual Modeling*. Cham: Springer Nature Switzerland, 2025. p. 235–253. ISBN 978-3-031-75872-0.

159 FILL, H.-G.; FETTKE, P.; KÖPKE, J. Conceptual modeling and large language models: impressions from first experiments with chatgpt. *Enterprise Modelling and Information Systems Architectures (EMISAJ)*, v. 18, p. 1–15, 2023.

160 CHAABEN, M. B.; BURGUENO, L.; SAHRAOUI, H. Towards using few-shot prompt learning for automating model completion. In: *Proceedings of the 45th International Conference on Software Engineering: New Ideas and Emerging Results*. IEEE Press, 2023. (ICSE-NIER '23), p. 7–12. ISBN 9798350300390. Disponível em: <https://doi.org/10.1109/ICSE-NIER58687.2023.00008>.

161 X, S.; MITTAL, S.; CHAUHAN, S. Advancing class diagram extraction from requirement text: A transformer-based approach. In: PAWAR, J. D.; DEVI, S. L. (Ed.). *Proceedings of the 20th International Conference on Natural Language Processing (ICON)*. Goa University, Goa, India: NLP Association of India (NLPAI), 2023. p. 433–441. Disponível em: <https://aclanthology.org/2023.icon-1.36/>.

162 BABAALLA, Z.; JAKIMI, A.; OUALLA, M. Llm-driven mda pipeline for generating uml class diagrams and code. *IEEE Access*, v. 13, p. 171266–171286, 2025.

163 AUGENSTEIN, I.; DAS, M.; RIEDEL, S.; VIKRAMAN, L.; MCCALLUM, A. SemEval 2017 task 10: ScienceIE - extracting keyphrases and relations from scientific publications. In: *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Vancouver, Canada: Association for Computational Linguistics, 2017. p. 546–555. Disponível em: <https://aclanthology.org/S17-2091>.

164 FALBO, R. de A.; MENEZES, C. S. de; ROCHA, A. R. A systematic approach for building ontologies. In: SPRINGER. *IBERAMIA*. [S.l.], 1998. v. 1484, p. 349–360.

165 BRASIL. Doutrina militar terrestre. *Estado Maior do Exército*, Brasília, DF, 2019. Disponível em: <https://www.gov.br/defesa/pt-br/arquivos/doutrina_militar/lista_de_publicacoes/md31a_sa_02a_conopsa_sismca_1a_eda_2015.pdf>.

166 AGUIAR, C. Z.; SOUZA, V. E. S. *SABiOx: the extended systematic approach for building ontologies*. [S.l.]: sn, 2024.

167 AVELINO, J.; CORDEIRO, K.; CAVALCANTI, M. C. Aer-mint - apoio ao processo de extração de relações baseado em mineração de dados textuais. In: *Anais do XXXVII Simpósio Brasileiro de Bancos de Dados*. Porto Alegre, RS, Brasil: SBC, 2022. p. 409–414. ISSN 2763-8979. Disponível em: <https://sol.sbc.org.br/index.php/sbbd/article/view/21828>.

168 KRUCHTEN, P. Architectural blueprints: the 4+ 1 view model of software architecture (1995). *IEEE Software*, v. 12, n. 6, 2005.

169 GANE, C. *Análise estruturada de sistemas*. [S.l.]: LTC, 1983.

170 YOURDON, E.; CONSTANTINE, L. L. *Structured Design: Fundamentals of a Discipline of Computer Program and Systems Design*. 1st. ed. USA: Prentice-Hall, Inc., 1979. ISBN 0138544719.

171 PRESSMAN, R. S. et al. A practitioner's approach. *Software Engineering*, v. 2, p. 41–42, 2010.

172 BRASIL. Garantia da Lei e da Ordem. *Ministério da Defesa*, Brasília, DF, 2014. Disponível em: <https://www.gov.br/defesa/pt-br/arquivos/2014/mes02/md33-m-10-garantia-da-lei-e-da-ordem-2a-ed-2014-31-jan.pdf>.

173 BRASIL. Manual de campanha: Operações (eb70-mc-10.223). *Estado Maior do Exército*, Brasília, DF, 2017. Disponível em: <https://bdex.eb.mil.br/jspui/handle/1/848>.

174 BRASIL. Manual de campanha: Operações (eb70-mc-10.246). *Estado Maior do Exército*, Brasília, DF, 2020. Disponível em: <https://bdex.eb.mil.br/jspui/handle/123456789/7073>.

175 FALBO, R. de A. Sabio: Systematic approach for building ontologies. In: *ONTO.COM/ODISE@FOIS*. [s.n.], 2014. Disponível em: <https://api.semanticscholar.org/CorpusID:17310637>.

176 GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J.; PATTERNS, D. Elements of reusable object-oriented software. *Design Patterns*, 1995.

177 LASKAR, M. T. R.; JAHAN, I.; DOLATABADI, E.; PENG, C.; HOQUE, E.; HUANG, J. *Improving Automatic Evaluation of Large Language Models (LLMs) in Biomedical Relation Extraction via LLMs-as-the-Judge*. 2025. Disponível em: <https://arxiv.org/abs/2506.00777>.

178 PIRES, R.; ABONIZIO, H.; ALMEIDA, T. S.; NOGUEIRA, R. Sabiá: Portuguese large language models. In: _____. *Lecture Notes in Computer Science*. Springer Nature Switzerland, 2023. p. 226–240. ISBN 9783031453922. Disponível em: <http://dx.doi.org/10.1007/978-3-031-45392-2_15>.

179 GUIZZARDI, G.; BENEVIDES, A. B.; FONSECA, C. M.; PORELLO, D.; ALMEIDA, J. P. A.; SALES, T. P. Ufo: Unified foundational ontology. *Applied ontology*, IOS Press, v. 17, n. 1, p. 167–210, 2022.

APÊNDICE A – DETALHES DAS ESPECIALIZAÇÕES DE C2RM

Tabela 9 – Quadro detalhado das especializações de C2RM

Tipo	**Construtor**	**Descrição**
Entidade	**Entity**	Descreve a entidade identificada na etapa de anotação. Essa entidade poderá assumir papéis relevantes de acordo com a sua aplicação no domínio. Além disso, a entidade anotada pode estar relacionada à outra entidade de acordo com a sua função. A entidade é algo essencial para o negócio, devendo ser identificada e rotulada de modo a expressar o seu significado. Exemplo: "Tarefa – trabalho ou conjunto de ações cujo propósito é contribuir para alcançar o objetivo geral da operação." Nesse caso, as **ENTIDADES**: "Tarefa", "Ações" e "Operação" devem ser anotadas.
Relação	**Type_of**	Tem como objetivo relacionar entidades identificadas no texto com a relação de generalização/especialização. Exemplo: "ELEMENTOS DO PODER DE COMBATE - Os elementos do poder de combate terrestre representam ... São eles: Liderança, Informações e as Funções de Combate...". Nesse caso, a relação entre as **ENTIDADES** "Liderança", "Informações" "Funções de Combate" e "poder de combate terrestre" serão anotadas através da **Type_of**.

Relação	**Responsible_for**	Tem como objetivo relacionar duas entidades que denotem uma relação de um comando ou responsabilidade para execução de algo. Exemplo: "Estado Final Desejado (EFD) – conjunto de condições futuras, relacionadas ao inimigo, ao terreno e às considerações civis, que o "comandante" visualiza que devem existir para que operação chegue ao fim." Nesse caso, entre as **ENTIDADES** "Estado Final Desejado" e "comandante" existe uma relação implícita entre essas duas entidades que expressa a subordinação ou um grau de responsabilidade. Por isso, que a relação entre elas deve ser anotada através de **Responsible_for**.
Relação	**Occurs_in**	Tem como objetivo anotar as entidades que possuem relação de ocorrência. Exemplo: "O emprego da F Ter pode ocorrer em dois tipos de situações: Nas situações de Guerra... e Nas situações de Não Guerra." Nesse caso as **ENTIDADES** "F Ter", "Guerra" e "Não Guerra" são anotadas através da especialização **Occurs_in**.
Relação	**Instance_of**	Tem como objetivo anotar as entidades que exerçam um papel de instância. Exemplo: "As operações militares têm como traço comum o ambiente interagências... A Operação Ágata, que integra o Plano Estratégico de Fronteiras (PEF) do Governo Federal, criado para prevenir e reprimir a ação de criminosos na divisa do Brasil com dez países sul-americanos...". Nesse caso, a anotação da relação entre as **ENTIDADES** "Operação Ágata" e "operações militares" é realizada através de **Instance_of**.

Relação	**Capacity_of**	Exemplo: Descreve uma característica intrínseca dos documentos doutrinários que são descritos de acordo com as capacidades necessárias que algo ou alguém deve possuir. Exemplo: "A obtenção dessas competências e capacidades é fundamental para que se possa atuar em todo o espectro dos conflitos..." Também devem ser considerados outros fatores para o emprego da F Ter, tais como: a) a letalidade seletiva; b) a proteção da tropa; c) a superioridade das informações; ...". Nesse caso, as **ENTIDADES** "F Ter", 'letalidade seletiva", "proteção da tropa", etc. serão anotadas através da especialidade **Capacity_of**.
Relação	**Composed_of**	Tem como objetivo anotar composições explicitadas nos textos. Exemplo: "A F Ter é constituída pelas organizações militares (OM) operativas, permanentes ou não, fundamentadas em um Quadro de Organização (QO)...". Nesse caso, a relação entre as **ENTIDADES**: "F Ter", "organizações militares" serão anotadas através da especialidade **Composed_of**.
Relação	**Applied_to**	Tem como objetivo anotar textos presentes nas doutrinas que indicam o "emprego", no sentido de aplicação, de "algo em alguma coisa". Exemplo: "A Defesa Móvel emprega uma combinação de ações ofensivas, defensivas e retardadoras..." Nesse caso, a anotação da relação entre as **ENTIDADES** "Defesa Móvel" e "ações ofensivas", "ações defensivas" e "ações retardadoras" é através da especialização **Applied_to**
Relação	**Defined_by**	Tem como objetivo relacionar as entidades à sua definição principal. **Exemplo**: "Tarefa e trabalho ou conjunto de ações cujo propósito é contribuir para alcançar o objetivo geral da operação.". Nesse caso, deve ser anotada a relação entre as **ENTIDADES** "Tarefa" e "trabalho ou conjunto de ações cujo propósito é contribuir para alcançar o objetivo geral da operação." através de **Defined_by**.

APÊNDICE B – APOIO AO EXPERIMENTO Ex_5

B.1 Minimundo

O minimundo a seguir é baseado nos conceitos de Operações Conjuntas (Op Cj) à luz da Doutrina de Operações Conjuntas - MD30-M-01[1], ressalvadas algumas simplificações para facilitar o entendimento. Nesse sentido, construa um modelo conceitual de dados, listando de modo separado as entidades identificadas e os relacionamentos entre as entidades. Observe que não é necessário identificar os atributos. Além disso, considere as Questões de Competência (QC) listadas após o minimundo que deverão ser respondidas através dos construtos (entidades e relacionamentos) utilizados na sua modelagem.

A Operação Conjunta (Op Cj) é um tipo de operação militar que se diferencia pela heterogeneidade dos processos de emprego e peculiaridades das Forças Componentes (F Cte), empregando as Forças Armadas (FA), representada por Marinha (MB), Exército (EB) e Aeronáutica (FAB), de modo conjunto, alinhada aos objetivos estratégicos estabelecidos nos diversos níveis (político, estratégico, operacional e tático), assim como os seus documentos resultantes, compreendendo maior integração das estruturas de comando e controle, de inteligência e de logística. Dado o caráter "conjunto", uma das capacidades da Op Cj é a interoperabilidade das FA, principalmente no apoio logístico, meios e de seus sistemas, os quais deverão ser coordenados no Comando Operacional através do Comando Logístico (C Log). De modo amplo, por exemplo, o Comandante Operacional conduz a Op Cj para alcançar os objetivos estratégicos e operacionais, em harmonia com os esforços políticos, diplomáticos e econômicos.

No nível político, o Comandante Supremo (CS), representado pelo Presidente da República (PR), estabelece os objetivos políticos e formula as diretrizes para ações estratégicas, consolidada na Diretriz Presidencial de Emprego de Defesa (DPED). Enquanto que no nível estratégico, o Ministério da Defesa (MD) estabelece diretrizes e planos, consolidado na Diretriz Ministerial de Emprego de Defesa (DMED). Os planos estratégicos servem de base para os Comandos Operacionais produzirem os Planos Operacionais, os quais norteiam os respectivos Planos Táticos dos Comandos das Forças Componentes. Nesse nível, são identificadas as Áreas de Responsabilidade (AR), também conhecida como Área de Interesse (AI), dos Comandos Operacionais a serem ativados, incluindo os meios adjudicados. Por exemplo, na Área de Operações (A Op), o COMDABRA ou COMAE é o responsável por empregar meios, como defesa aeroespacial ativa e passiva, na defesa antiaérea. No nível operacional, o Comandante Operacional elabora o planejamento

[1] https://www.gov.br/defesa/pt-br/arquivos/legislacao/emcfa/publicacoes/doutrina/md30-m-01-vol-1-2a-edicao-2020-dou-178-de-15-set.pdf

militar da campanha alinhado aos documentos dos outros níveis. Além disso, ele estabelece os objetivos operacionais e das missões a serem atribuídas às Forças Componentes (F Cte). Por fim, no nível tático, as F Cte são responsáveis pelos planos táticos e ordens de operações

- QC_1: Quais os níveis envolvidos e os responsáveis pelas Operações Conjuntas?
- QC_2: Quais os objetivos envolvidos em uma Operação Conjunta e seus respectivos níveis?
- QC_3: Quais as diretrizes e os planos envolvidos em uma Operação Conjunta?
- QC_4: Quais as áreas que compreendem as Op Cj, destacando os níveis e responsáveis?
- QC_5: Quais os meios empregados na defesa antiaérea?

B.2 Extrato do IDEA-C2-KG disponibilizado aos participantes

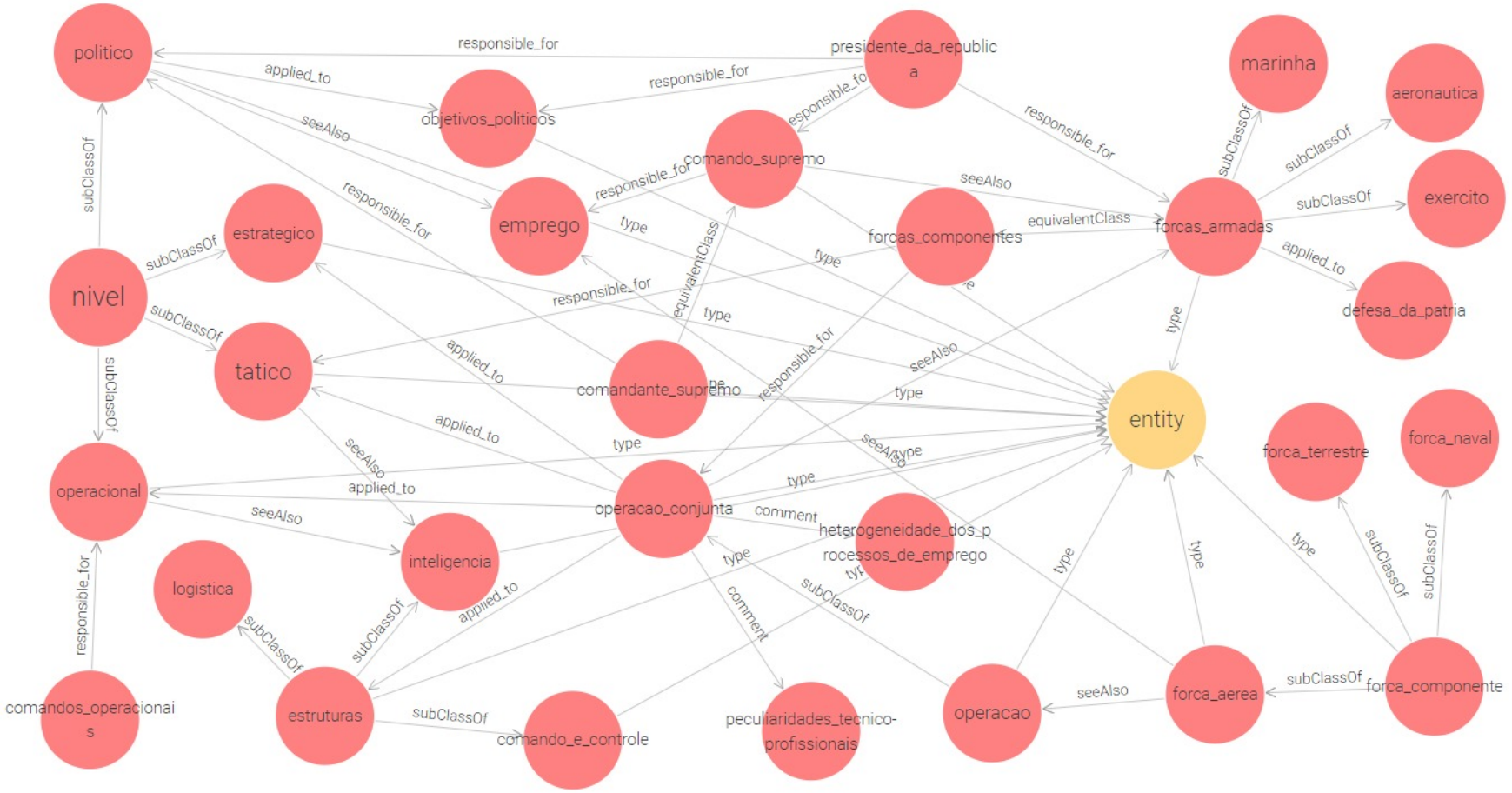


Figura 36 – IDEA-C2-KG disponibilizado como suporte no experimento Ex_5. Imagem do autor.

B.3 Modelo de domínio *Gold Standard* (DM^D)

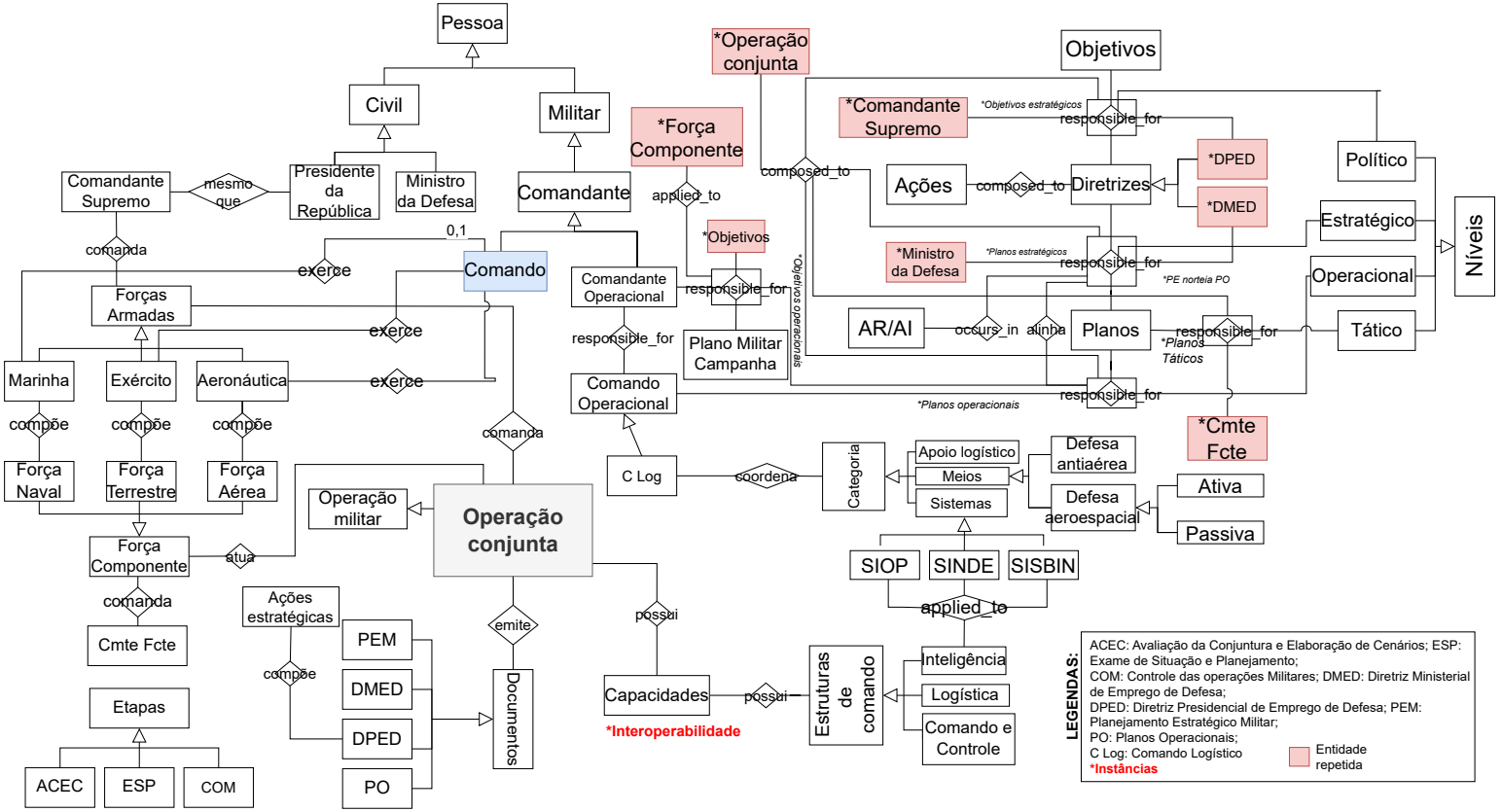


Figura 37 – DM^D *Gold Standard* do experimento Ex_5. Imagem do autor.

B.4 Modelo de domínio DM^{24}

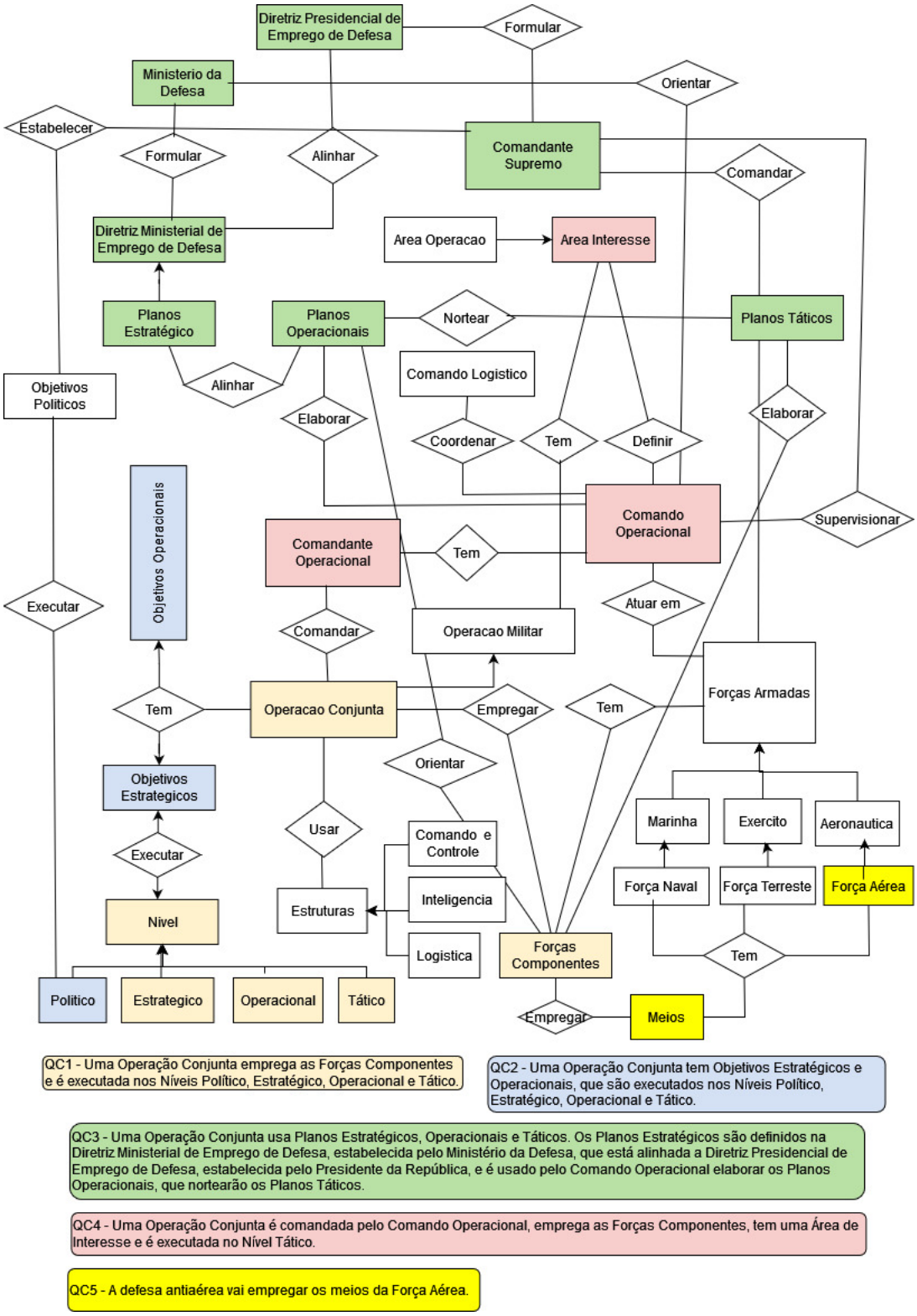


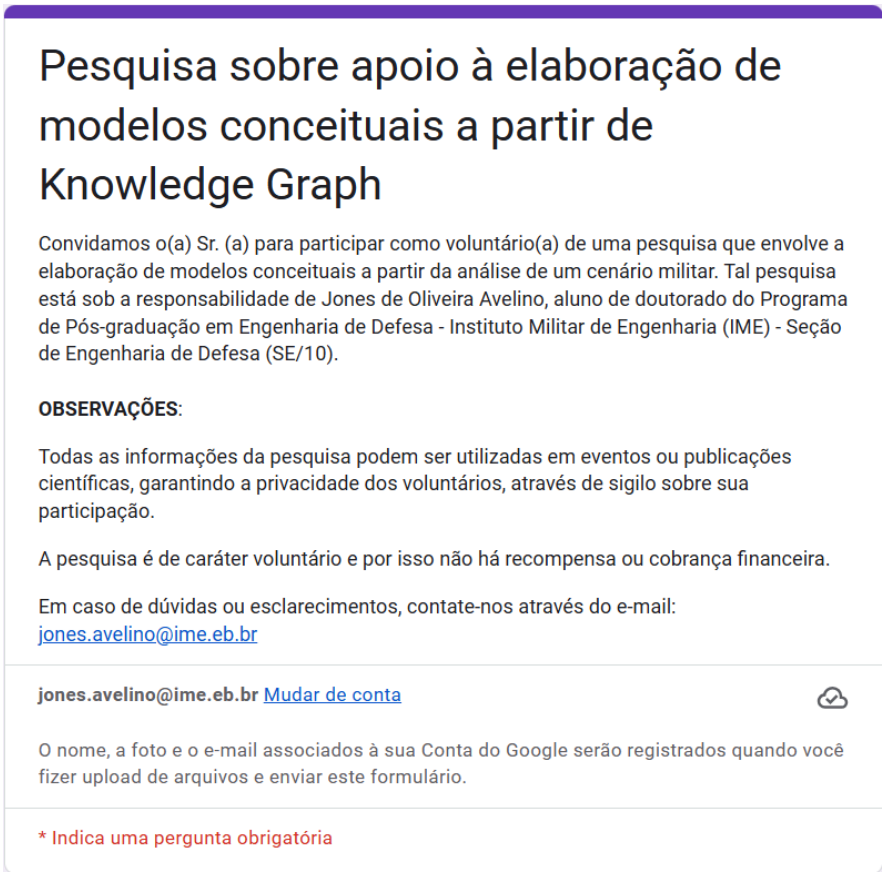
Figura 38 – DM^{24} do experimento Ex_5. Imagem do autor.

B.5 Convocação dos participantes

Figura 39 – E-mail de convocação dos participantes. Imagem do autor.

B.6 Formulário de participação na pesquisa

Na Figura 40, é ilustrado o formulário que aplicado no experimento EX_5, o qual é dividido em quatro seções principais: (i) informações sobre o experimento; (ii) informações sobre o participante; (iii) detalhamento sobre o cenário de aplicação; e (iv) questionário de avaliação. Na seção "informações sobre o experimento", são repassadas aos participantes as informações básicas sobre o experimento. Já na seção "informações sobre o participante", são solicitadas informações sobre o participantes, principalmente sobre seu perfil profissional e experiência em modelagem conceitual. Na seção "detalhamento sobre o cenário de aplicação", são informados aos participantes a descrição do cenário de aplicação através do minimundo que explora o universo das operações militares conjuntas. Por fim, na seção "questionário de avaliação", são solicitados aos participantes a avaliação do experimento, principalmente o suporte do IDEA-C2-KG na construção do modelo de domínio.



Pesquisa sobre apoio à elaboração de modelos conceituais a partir de Knowledge Graph

Convidamos o(a) Sr. (a) para participar como voluntário(a) de uma pesquisa que envolve a elaboração de modelos conceituais a partir da análise de um cenário militar. Tal pesquisa está sob a responsabilidade de Jones de Oliveira Avelino, aluno de doutorado do Programa de Pós-graduação em Engenharia de Defesa - Instituto Militar de Engenharia (IME) - Seção de Engenharia de Defesa (SE/10).

OBSERVAÇÕES:

Todas as informações da pesquisa podem ser utilizadas em eventos ou publicações científicas, garantindo a privacidade dos voluntários, através de sigilo sobre sua participação.

A pesquisa é de caráter voluntário e por isso não há recompensa ou cobrança financeira.

Em caso de dúvidas ou esclarecimentos, contate-nos através do e-mail: jones.avelino@ime.eb.br

jones.avelino@ime.eb.br Mudar de conta

O nome, a foto e o e-mail associados à sua Conta do Google serão registrados quando você fizer upload de arquivos e enviar este formulário.

* Indica uma pergunta obrigatória

Figura 40 – Formulário disponibilizado aos participantes do experimento. Imagem do autor.

No que diz respeito ao questionário de avaliação aos participantes, as questões indagavam sobre a concordância com uma afirmativa, as quais foram fornecidas as seguintes

opções de resposta em uma escala de 1 (um) a 5 (cinco), sendo: (1) Discordo fortemente; (2) Discordo parcialmente; (3) Não concordo nem discordo; (4) Concordo parcialmente; e (5) Concordo totalmente.

Os participantes foram instruídos a selecionar a opção a opção "Concordo totalmente" quando considerassem que concordavam com a afirmação na proporção de 100%. Já a opção "Concordo parcialmente", quando considerassem uma proporção superior a 50% e inferior a 100%. Assim como, a selecionar a opção "Não concordo nem discordo" quando considerassem a proporção de 50%. Já a opção "Discordo parcialmente", quando a proporção for inferior a 50% e superior a 0%. E, finalmente, a selecionar a opção opção "Discordo parcialmente", quando a proporção for 0%. As questões de cada seção são listadas a seguir.

- Seção 1: Informações sobre o experimento
  - E-mail
- Seção 2: Informações sobre o participante
  - Nome completo
  - Grau de escolaridade (Opções de resposta: Ensino Médio; Graduação; Pós-graduação Lato Sensu; Mestrado; e Doutorado).
  - Atividade profissional (Opções de resposta: Setor público - Militar / Civil; Setor privado - Militar / Civil)
  - Tempo de experiência profissional (Opções de resposta: menor que 5 anos; entre 5 e 10 anos; e maior que 10 anos)
  - Tempo de experiência profissional em modelagem de dados (Opções de resposta: menor que 2 anos; entre 2 e 5 anos; entre 6 e 10 anos; e mais que 10 anos)
  - Consentimento de participação (Opção de resposta: Concordo)
- Seção 3: Detalhamento sobre o cenário de aplicação
  - Realizar o download do cenário de aplicação
- Seção 4: Questionário de avaliação
  - O minimundo apresentado expressa o cenário de maneira clara e concisa? (Opções de resposta: Escala de 1 a 5)
  - O tempo de 60 minutos disponível para elaborar o modelo conceitual é suficiente para realizar o exercício? (Opções de resposta: Escala de 1 a 5)
  - Durante a modelagem ocorreram situações de dúvida quanto à representação de uma entidade ou relacionamento? (Opções de resposta: Escala de 1 a 5)

- Você utilizou o grafo como apoio? (OBS: Caso não tenha utilizado o grafo, não é necessário responder as próximas perguntas.) (Opções de resposta: Sim / Não)
- As entidades do grafo retratam o domínio do minimundo? (Opções de resposta: Escala de 1 a 5)
- As relações do grafo retratam o domínio do minimundo? (Opções de resposta: Escala de 1 a 5)
- Compreendi claramente os nós e arestas do grafo em relação ao contexto do minimundo. (Opções de resposta: Escala de 1 a 5)
- No geral, o grafo contribuiu com a elaboração do modelo conceitual do minimundo. (Opções de resposta: Escala de 1 a 5)
- Recomendo o uso de grafo de conhecimento em apoio à elaboração de modelos conceituais. (Opções de resposta: Escala de 1 a 5)
- No geral, comente sobre dificuldades encontradas durante sua modelagem. (Opções de resposta: Texto livre)
- Agradecemos a sua participação! Caso queira fazer comentários ou sugestões, fique à vontade. (Opções de resposta: Texto livre)

APÊNDICE C – REPOSITÓRIO DA PESQUISA

Durante a pesquisa, todo o material produzido foi armazenado em um repositório no Github[1]. Nele, são disponibilizados o código-fonte, os corpora (texto bruto, pré-anotado e curado), os pré-requisitos arquiteturais do protótipo IDEA-C2-Tool e os experimentos. Nesse último, cabe destacar que a cada experimento, o código-fonte foi clonado por meio do Google Colab, preservando cada linha de código executada, bem como os dados utilizados e os resultados produzidos.

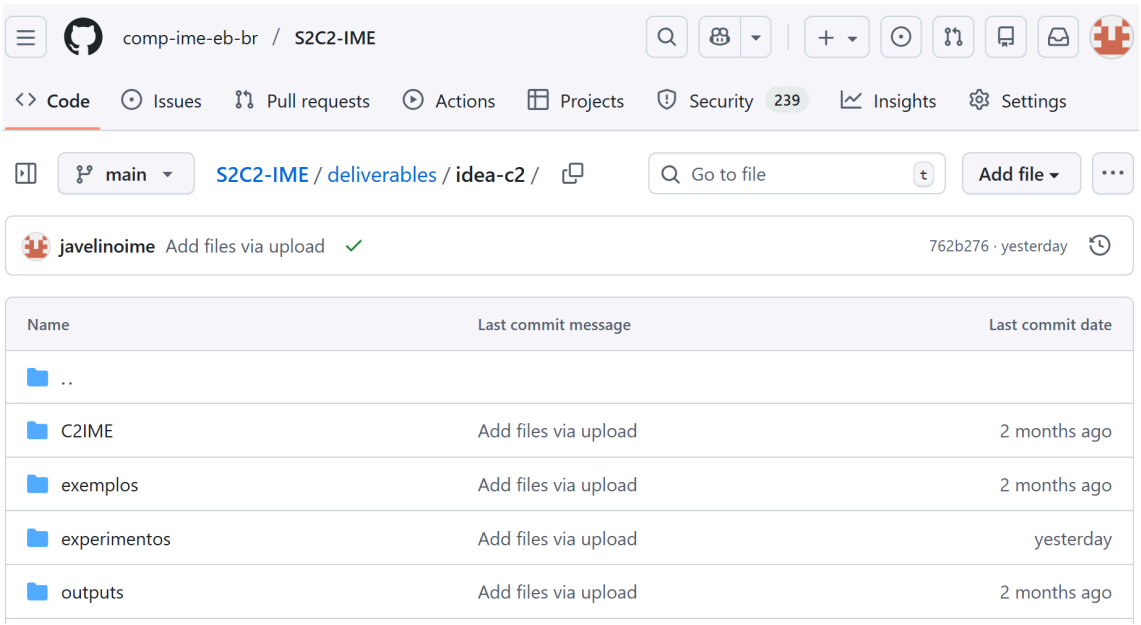


Figura 41 – Repositório do IDEA-C2. Imagem do autor.

Na Figura 41, é ilustrada a tela principal do repositório, em destaque as pastas: (i) Principal (*main*): onde está localizado o código-fonte, em formato de caderno (*nootebook*) do IDEA-C2-Tool; (ii) C2IME: pasta que reúne os códigos-fontes de apoio e reutilizáveis no protótipo; (iii) Exemplos: pasta com os códigos-fontes dos exemplos utilizados no texto da tese; (iv) Experimentos: pasta com os seis experimentos da tese; (v) Outputs: pasta que reúne os arquivos de saída obtidos por meio do processamento do IDEA-C2-Tool; (vi) Texts: pasta com os arquivos dos corpora pré-anotados e curado. Além disso, na pasta principal há o arquivo README.md que reúne informações explicativas e passo a passo sobre o protótipo, corpus, pré-requisitos, etc.

[1] <https://github.com/comp-ime-eb-br/S2C2-IME/tree/main/deliverables/idea-c2>