AER-MinT - Apoio ao processo de Extração de Relações baseado em Mineração de dados Textuais

Jones O. Avelino[1,2], **Kelli F. Cordeiro**[1,2], **Maria C. Cavalcanti**[1]

[1]Instituto Militar de Engenharia (IME)
Praça General Tibúrcio 80, Praia Vermelha – 22.290-270 – Rio de Janeiro

[2]Centro de Análise de Sistemas Navais
Ed. 23 do AMRJ – R. da Ponte, s/n, Centro – 20.091-000 – Rio de Janeiro

{jones.avelino,kelli}@marinha.mil.br, yoko@ime.eb.br

Abstract. *The growth of unstructured data on the Web provides some services. One of them is the acquisition of knowledge that the information extraction process is able to offer. To this end, dataset enrichment approaches began to use unstructured data, adopting machine learning algorithms in order to increase its effectiveness. However, there is a lack of support instruments and there is a low supply of datasets. Thus, this article proposes AER-MinT, an approach capable of applying a training model from a corpus of texts, using BERT and a Convolutional Neural Network, in order to support the extraction of relations in text sentences. As a result, exploration through an RDF graph is possible.*

Resumo. *O crescimento de dados não estruturados na Web propicia alguns serviços. Um deles é a obtenção de conhecimento que o processo de extração de informações é capaz de oferecer. Para tal, abordagens de enriquecimento de datasets começaram a utilizar dados não estruturados, adotando algoritmos de machine learning a fim de aumentar a sua efetividade. Entretanto, há carência de instrumentos de apoio e há baixa oferta de datasets. Assim, este artigo propõe AER-MinT, uma abordagem capaz de aplicar um modelo de treinamento a partir de um corpus de textos, utilizando o BERT e uma Rede Neural Convolucional, com objetivo de apoiar a extração de relações em sentenças de textos. Como resultado, é possível a exploração através de um grafo RDF.*

1. Introdução

Nos últimos anos, com o aumento de dados não estruturados na Web houve uma demanda de consumo e serviços sobre esses dados. Um desses serviços é a obtenção de conhecimento a partir da extração de dados sobre fontes não estruturadas. Nesse contexto, surgiram abordagens propondo respostas mais efetivas, utilizando algoritmos de Aprendizado de Máquina (AM) para processar dados em cenários de atividades repetitivas. Todavia, se inseria a difícil tarefa de extrair conhecimento sobre esses dados, agora, não estruturados.

A Web Semântica (WS) apresentou o Linked Open Data (LOD) a partir de um modelo capaz de publicar e interligar dados na Web, alguns associados às ontologias. Porém, algumas interligações não foram efetivas na extração de conhecimento dada sua precariedade semântica. Dessa forma, surgiram abordagens focadas na ampliação de *datasets* a fim de enriquecer a semântica das relações utilizando dados textuais [Sherif et al. 2015].

Neste trabalho propomos AER-MinT, uma abordagem que tem o objetivo de apoiar o processo de Extração de Relações a partir da submissão de um Corpus de Textos a um modelo pré-treinado, classificado com base em uma Rede Neural Convolucional (CNN), a fim de produzir uma lista de sugestões com a extração de relações. Para tal, implementamos o protótipo AER-MinTTool que realiza o treinamento da base de dados utilizando o modelo BERT. De acordo com os resultados obtidos são geradas as sugestões que podem ser exploradas via grafo RDF. Como contribuições: (i) uma abordagem capaz de apoiar a extração de relação; (ii) uma ferramenta que implementa as atividades da abordagem; e (iii) um estudo de caso real que demonstra a viabilidade e utilidade.

2. Ampliação de *Datasets* através de Interligações

No contexto da WS, as ontologias representam a conceituação compartilhada de um determinado domínio, utilizando vocabulários controlados a fim de explicitar a sua semântica [Guarino 1995]. Considerando as necessidades de representação, a linguagem Resource Description Framework (RDF)[1] utiliza grafos direcionados em que nós e arestas são rotulados e representados por triplas (*sujeito, predicado e objeto*). Cada elemento da tripla permite associar e reutilizar vocabulários controlados e/ou ontologias. Busca-se não somente expressar o conhecimento, mas também extrair informações através das relações dos *datasets*, aqui denominadas como interligações [Avelino et al. 2020].

As interligações são baseadas na especificação do *Link Discovery* e fundamentais para ampliação de *datasets*, sendo descritas por rótulos que expressam as equivalências entre *datasets*, como exemplo: *sameAs* e *seeAlso*. Apesar de haver interligações, nem sempre há clareza semântica, dificultando a extração do conhecimento [Sherif et al. 2015]. Alguns autores propuseram abordagens que utilizam o Processamento de Linguagem Natural (PLN), do inglês *Natural Language Processing*, através de técnicas de Extração de Relações (RE do inglês *Relation Extraction*), a fim de extrair dados de fontes não estruturadas para enriquecer *datasets* [Silveira and Cavalcanti 2020].

3. Extração de informação para enriquecimento de *datasets* a partir de textos

As atividades de treinamento em Inteligência Artificial podem envolver modelos de Redes Neurais baseados em *Deep Learning* (DL) e dispõem de um método de aprendizagem que utiliza uma série de camadas sucessivas para descobrir recursos ocultos [Géron 2019]. As camadas representam neurônios matemáticos que consistem em: *input*, *output* e *hidden*. Em PLN, uma das formas de representar as relações entre os termos é utilizar as CNNs. As CNNs foram desenvolvidas para imagens, mas também são propícias para estruturas de texto, i.e. sequências de palavras e árvores de dependências. Como os modelos de DL utilizam vetores de números é necessário converter o texto em números para utilizar o modelo [Miwa and Bansal 2016]. A conversão é também conhecida como vetorização de texto, no qual pode ser utilizada a estratégia de *word embedding* que consiste em um vetor de números que representa cada palavra do vocabulário, fornecendo uma representação densa em que palavras semelhantes são codificadas de mesmo modo [Géron 2019].

Uma evolução na estratégia de vetorização é o **B**idirectional **E**ncoder **R**epresentations from **T**ransformers (BERT) que utiliza o treinamento bidirecional do *Transformer* e consiste em um modelo para pré-treinamento de representação profunda

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

de texto não rotulado em todas as camadas, dividindo-se em duas arquiteturas: *BASE* de 12 camadas e *LARGE* de 24 camadas [Devlin et al. 2019]. O *Transformer* é um modelo baseado na arquitetura *enconder-decoder* com um mecanismo de autoatenção que permite aprender relações contextuais entre palavras dentro do texto, relacionando diferentes posições, ao invés de analisar a sequência de modo unidirecional [Vaswani et al. 2017].

4. Trabalhos relacionados

Há trabalhos como em [Teixeira et al. 2018] e [Silveira and Cavalcanti 2020], que apresentam abordagens que adotam mineração de dados textuais utilizando PLN para enriquecer *datasets*. Em [Teixeira et al. 2018], a abordagem utiliza a matriz TF-IDF para relacionar os termos e incorporá-los aos *datasets* já existentes como recursos para extração de novas triplas. Já o trabalho de [Silveira and Cavalcanti 2020], propõe a **P**redicate **LA**bel**IN**g (PLAIN)[2] a partir da avaliação dos predicados com baixa relevância semântica. Para tal, utiliza a implementação OpenNRE para extrair as relações das sentenças de textos com base no corpus pré-treinado da Wikipedia. Apesar de ambos os trabalhos utilizarem técnicas de RE, não oferecem flexibilidade de treinamento do corpus alinhado ao domínio do negócio. Assim, este trabalho diferente dos demais, propõe uma ferramenta semiautomatizada que utiliza o modelo BERT e uma CNN para classificação do corpus, permitindo a exploração dos dados através de um grafo RDF.

5. AER-MinT

A abordagem supervisionada AER-MinT (**A**poio à **E**xtração de **R**elações baseado em **Min**eração de dados **T**extuais) possui um modelo de processo composto por 9 atividades, ilustrado na Figura 1. O objetivo é submeter um conjunto de sentenças a um corpus pré-treinado, assim como o par de termos, denominados de *head* e *tail*, dentre os quais se deseja extrair a relação, através da avaliação de sentenças. Já **AER-MinTTool**[3] é um protótipo desenvolvido em Python baseado em AER-MinT para realizar experimentos e avaliar a sua viabilidade. As relações extraídas são armazenadas no *Triplestore* GraphDB.

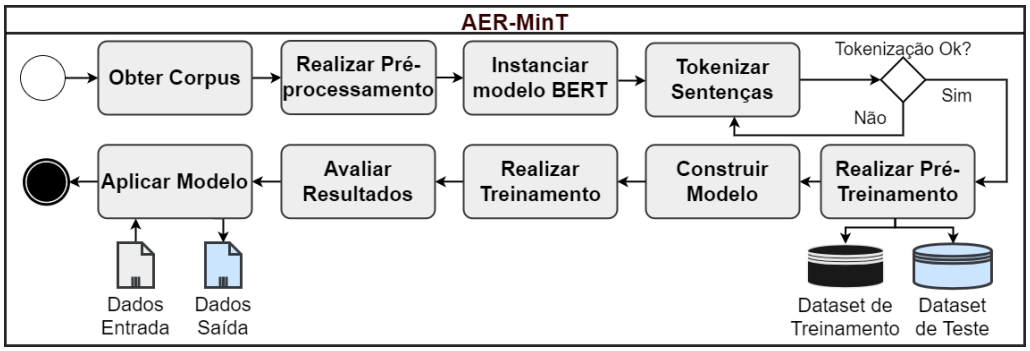

Figura 1. Processo da abordagem AER-MinT. Imagem do autor

Basicamente, as quatro atividades iniciais de AER-MinT objetivam treinar o modelo. A atividade **Obter Corpus** seleciona as sentenças s do corpus de texto não rotulado, C, denotado por $C = \{s_1, s_2, \ldots, s_n\}$. A atividade **Realizar Pré-processamento** recupera as sentenças s e carrega o par de termos, *head* e *tail*, definidos pelo par $\langle x_i, y_i \rangle$. O par

[2]https://github.com/rafans222/plain

[3]https://github.com/jonesavelino/AER-MinTTool

é utilizado como referência a ser buscado nas sentenças na aplicação do modelo. Além disso, as sentenças são padronizadas para minúsculo e removidas as *stopwords*. A atividade **Instanciar modelo BERT** define a arquitetura BERT_{LARGE}, a matriz de *embeddings* de 1024 e 16 *subspaces*. A atividade **Tokenizar Sentenças** recupera as sentenças s, divide cada termo em *tokens* tk e transforma-os em identificadores id, como exemplo a Tabela 1. Como essa atividade é realizada pelo BERT, a abordagem limita-se em em verificar se a sentença foi de fato tokenizada. Caso contrário, a sentença é submetida novamente. Por fim, o BERT recupera os id e os atribui à matriz de *embeddings* para sua representação vetorial, representado por $DS = \{s_n tk_m id_1, s_n tk_m id_2, \ldots, s_n tk_m id_i\}$. A atividade **Realizar Pré-Treinamento** recupera os id, define os parâmetros de treinamento e divide o DS em treinamento e testes, correspondendo a 75% e 25% dos dados.

Tabela 1. Exemplo da atividade Tokenizar Sentenças

Sentença (S_n)	*Tokenização* ($S_n tk_m$)	Identificadores ($S_n tk_m id_i$)
Vegetables contain significant quantities of nitrate and nitrite.	['vegetables','contain','significant','quantities','of','nitrate','and','ni','\#\#tri','\#\#te]	[11546, 5383, 3278, 12450, 1997, 29607, 1998, 9152, 18886, 2618, 1012]

A atividade **Construir Modelo** adota uma CNN de uma dimensão para classificação. A primeira camada da matriz possui o mesmo tamanho da sentença. Ao multiplicarmos os valores das matrizes são produzidos nas camadas de convolução/filtros os resultados dos trechos das sentenças (*bigramas, trigramas e fourgramas*) para capturar as diferentes escalas que as palavras podem se relacionar. A camada de *pooling* extrai o maior valor das camadas de convolução e o concatena ao vetor final. Esse vetor final é repassado à camada densa (*merged*) de acordo com as classes, utilizando a função *softmax* que retorna a probabilidade para cada uma das classes. Por fim, é aplicado um *dropout* de 20% para zerar uma parte dos neurônios da rede neural para evitar o *overfitting*. A atividade **Realizar Treinamento** define os hiperparâmetros e cinco épocas para o treinamento. A atividade **Avaliar Resultados** utiliza a função de cálculo de erro (*loss*) e a métrica (*accuracy*) para avaliar o retorno do treinamento, como ilustrado na Figura 2.

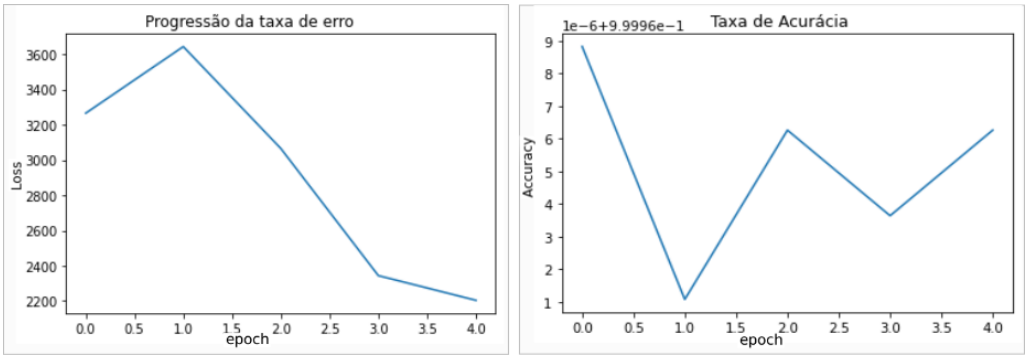

Figura 2. Resultados da execução do Treinamento de AER-MinT.

Por fim, a atividade **Aplicar Modelo** é dividida em duas etapas. A primeira aplica as sentenças de textos ao modelo, realiza a predição e gera o resultado de saída C', como visto no Algoritmo 1. Já a segunda, recupera os resultados C' e realiza a triplificação para o padrão RDF. As sentenças s são definidas como sujeito e as demais relações são construídas. Assumimos os termos, x_i e y_i, e os verbos v_i como objetos das triplas.

Algoritmo 1: Extrair Relações baseado na abordagem AER-MinT

Entrada: Corpus; $C = \{s_1, s_2, \ldots, s_n\}$; Termos: [$x_i$ *(head)* e y_i *(tail)*]
Saída: Extração de Relações: [$C' = \{(s_i, x_i, v_i, y_i), i = 1, \ldots, m; m < n\}$]
1 **para cada** $(s_i, x_i, y_i) \in$ ***C*** **faça**
2 **C'** $\leftarrow$ $obter\_predicao(s_i)$;
3 **fim**
Retorna: C';

6. Estudo de caso

AER-MinTTool foi aplicado no estudo de caso baseado na amostra do trabalho de [Silveira and Cavalcanti 2021], que buscou relações semânticas entre compostos químicos nas águas extraídas em poços de captação subterrâneas. Neste trabalho, o objetivo é recuperar os dados da amostra, submeter ao modelo pré-treinado de acordo com o NLM-Chem corpus[4], baseado em textos bioquímicos de publicações da PubMed, e avaliar a similaridade semântica dos compostos. Na Seção 5, as quatro atividades iniciais de AER-MinT foram apresentadas. Agora, vamos executar a atividade **Aplicar Modelo**, como visto no Algoritmo 1. A partir das sentenças, s_1, s_2, s_3, obtém-se como saída as respostas das RE, $C' = \{(s_i, x_i, v_i, y_i), i = 1, \ldots, m; m < n\}$, ilustrado na Tabela 2.

Tabela 2. Exemplo com as sentenças (s_1, s_2, s_3)

Entrada	Predição	Saída
$C = \{s_1, s_2, \ldots, s_n\}$	$obter\_predicao(s_i)$	$C' = \{(s_i, x_i, v_i, y_i), i = 1, \ldots, m; m < n\}$
(S_1) For a century, the Griess reaction ... used ... bacterial infection ... anion in human urine.	*Encontrado!*	For a century... urine.;**nitrate;nitrite;used** For a century... urine.;**nitrate;nitrite;produced**
(S_2) Nitrite and nitrate are among ... widely used in agricultural and industrial products.	*Encontrado!*	Nitrite and nitrate... industrial products.;**nitrate;nitrite;used**
(S_3) I then discuss ... patient management.	*Não Encontrado!*	

Na Figura 3, são ilustrados os resultados de C' em um grafo RDF, permitindo explorar as relações entre os recursos s_1 e s_2, favorecendo a navegação e inferências.

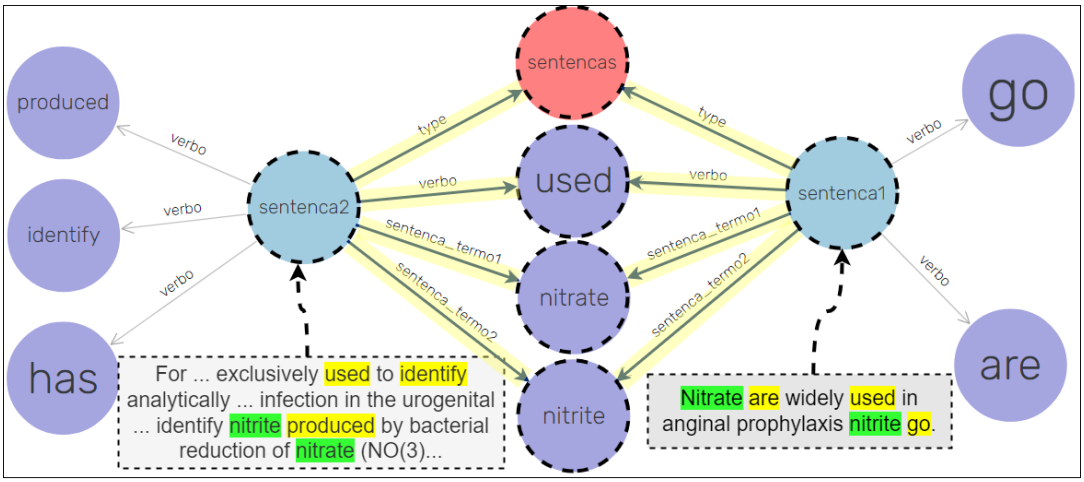

Figura 3. Destaque das relações entre os recursos: (s_1) e (s_2).

[4]https://ftp.ncbi.nlm.nih.gov/pub/lu/NLMChem/

7. Considerações Finais

Este artigo apresentou AER-MinT, uma abordagem para apoiar o processo de RE com base na avaliação de sentenças submetidas ao modelo pré-treinado no domínio de aplicação. Além disso, a abordagem foi capaz de obter um corpus de textos com base no domínio do estudo de caso e realizar as rotinas necessárias para treinar seus dados. Por fim, AER-MinT foi capaz de explicitar o resultado das extrações em um grafo RDF, favorecendo análises dos recursos armazenados e possibilitando a inferência de cenários ainda não explorados sobre as relações. Para isso, foi implementada a ferramenta AER-MinTTool que fornece mecanismos semiautomatizados baseados em técnicas de PLN, utilizando o modelo BERT e uma CNN para classificação, com a flexibilidade de treinamento de corpus de textos. AER-MinTTool foi submetida ao estudo de caso sobre um cenário real e os resultados obtidos evidenciaram tanto a utilidade quanto a sua viabilidade. Trabalhos futuros incluem: (i) a implementação de um *Web Crawler* para buscar novas entradas de dados a partir de sites alinhados ao negócio. (ii) AER-MinT pode ser estendida para agregar *Knowledge Graph*.

Referências

[Avelino et al. 2020] Avelino, J., Cordeiro, K., and Cavalcanti, M. C. (2020). An RDF Based Approach for Integrating Data at Different Levels of Abstraction. WebMedia '20, page 81–88.

[Devlin et al. 2019] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the NAACL*, pages 4171–4186. Association for Computational Linguistics.

[Géron 2019] Géron, A. (2019). *Mãos à Obra: Aprendizado de Máquina com Scikit-Learn TensorFlow*.

[Guarino 1995] Guarino, N. (1995). *The Ontological Level*, pages 443–456. Holder-Pivhler-Tempsky.

[Miwa and Bansal 2016] Miwa, M. and Bansal, M. (2016). 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*, pages 1105–1116.

[Sherif et al. 2015] Sherif, M. A., Ngomo, A.-C. N., et al. (2015). Automating rdf dataset transformation and enrichment. In *The Semantic Web. Latest Advances and New Domains*, pages 371–387.

[Silveira and Cavalcanti 2020] Silveira, R. and Cavalcanti, M. (2020). Método para rotular ligações semânticas na web de dados. In *Anais do XXXV Simpósio Brasileiro de Bancos de Dados*, pages 49–60.

[Silveira and Cavalcanti 2021] Silveira, R. and Cavalcanti, M. (2021). *Método para Rotular Ligações Semânticas na Web de Dados*. Mestrado em Sistemas e Computação, IME.

[Teixeira et al. 2018] Teixeira, K. T., Campos, M. L. M., et al. (2018). Extração de dados de fontes textuais: Uma abordagem para enriquecimento de dados abertos interligados. In *SEMISH*. SBC.

[Vaswani et al. 2017] Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). Attention is All you Need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.