Uma Abordagem de Anotação Semântica Automática Direcionada a Sistemas de Perguntas e Respostas

Laura L. Dias[1], Luciano V.B. Espiridião[1], Anderson A. Ferreira[1]

[1]Departamento de Ciência da Computação (DECOM)
Universidade Federal de Ouro Preto (UFOP) – Ouro Preto – MG – Brazil

{laura.lima1,luciano.espiridiao}@aluno.ufop.edu.br
anderson.ferreira@ufop.edu.br

Abstract. *The fast increasing of content repositories has led to the need for better indexing and searching mechanisms, including question answering systems. Users still face difficulties in navigating on the vast amount of information on the Web. However, studies on automatic semantic annotation have allowed us identifying contents in repositories and have supported different systems. This work proposes a method of processing questions, through BERT, to carry out the semantic annotation task, adding DBpedia resources as context to questions. Experimental results show advances up to 13% when compared with the baseline.*

Resumo. *O crescimento acelerado dos repositórios de conteúdo tem ocasionado à necessidade de melhores mecanismos de indexação e busca, incluindo sistemas de perguntas e respostas. Os usuários ainda enfrentam dificuldades para navegar no grande volume de informações na Web. No entanto, estudos sobre anotação semântica automática permitem a identificação de conteúdos nos repositórios e auxiliam diversos sistemas. Este trabalho propõe um método de processamento de perguntas, por meio da BERT, para a realização da tarefa de anotação semântica, agregando recursos da DBpedia como contexto às perguntas. Os resultados experimentais mostram avanços de até 13% quando comparados ao baseline.*

1. Introdução

Atualmente, o avanço tecnológico facilita a criação de conteúdos digitais, provocando o crescimento significativo dos repositórios digitais. Isso leva à necessidade de melhorar os mecanismos de classificação e busca desses conteúdos, bem como automatizar a recuperação dessas informações [Chandurkar and Bansal 2017, Mohasseb et al. 2018]. Em decorrência desse grande volume de dados, a maioria dos usuários ainda não consegue extrair o máximo potencial em suas buscas [Gupta et al. 2015]. Isso gera a necessidade de soluções que auxiliem o usuário nessa tarefa. Uma alternativa amplamente utilizada na literatura, que realiza a seleção criteriosa de informação, são os sistemas de perguntas e respostas (QA - *Question Answering*) [Chandurkar and Bansal 2017]. Esses sistemas devem facilitar o processo de recuperação da informação e permitir que um usuário faça uma pergunta na linguagem cotidiana e receba uma resposta rápida e sucinta, com contexto[1] suficiente para validar essa resposta [Hirschman and Gaizauskas 2001].

[1]Conjunto de palavras, frases, ou o texto que precede ou se segue a determinada palavra, frase ou texto e que contribuem para o seu significado.

Em [Dimitrakis et al. 2020], é apresentado um *pipeline* geral de um sistema de QA. Nele, o usuário fornece uma pergunta em linguagem natural e esta percorre 5 etapas, sendo: *Análise da pergunta*, que se concentra principalmente em classificar o tipo de pergunta, o tipo de resposta esperada e o foco da pergunta; *Correspondência de dados*, concentra-se em identificar trechos do texto, como palavras específicas da pergunta de entrada, referentes às entidades, classes ou propriedades em um grafo de conhecimento [Paulheim 2017]; *Pontuação e inferência conjunta*, essa etapa seleciona um dos recursos candidatos, a fim de construir uma única consulta SPARQL[2] [Seaborne and Prud'hommeaux 2005]; *Construção da consulta* é a tradução da pergunta para uma consulta SPARQL a fim de recuperar a resposta; *Apresentação da resposta*, em que é necessário um processamento para torná-la inteligível.

Na etapa de análise da pergunta, a precisão na classificação e no processamento das perguntas pode afetar a qualidade da informação recuperada e a resposta extraída pelos módulos restantes [Shah et al. 2018]. Os sistemas de QA utilizam métodos de extração de informação para identificar um conjunto de prováveis candidatos a resposta [Ko et al. 2010, Shah et al. 2018]. Com o auxílio de anotações semânticas, as perguntas são relacionadas a recursos que fazem parte de uma rede de relações entre significados, como uma ontologia ou um tesauro. Sendo assim, a anotação semântica da pergunta facilita não somente a classificação e o processamento da pergunta e da resposta, mas também o processo de busca nos repositórios. Por consequência, muitos pesquisadores têm trabalhado em melhorias nesse processo, considerando diversos tipos de mídias [Dasiopoulou et al. 2011, Qazi and Goudar 2016].

Em [Dimitrakis et al. 2020], é relatado que a etapa de análise da pergunta deve realizar uma análise linguística, capturando a estrutura sintática e semântica da pergunta. Para capturar a semântica da pergunta, Dimitrakis et al. [2020] descrevem abordagens como o Reconhecimento de Entidade Nomeada (NER - *Named Entity Recognition*) e a Vinculação de Entidades (EL - *Entity Linking*). Esses processos já estão implementados em ferramentas amplamente utilizadas como Stanford CoreNLP[3] ou DBpedia Spotlight[4]. No entanto, essas abordagens consistem em identificar a ocorrência de uma entidade nomeada no texto, no caso do NER, e em atribuir uma identidade única a entidade, no caso da EL. Porém, não é função dessas abordagens considerar todo o texto ao redor do nome da entidade para realizar esses processamentos, geralmente o foco é no nome da entidade [Amaral 2013].

Por isso, este trabalho propõe um método de processamento de perguntas através de uma abordagem de anotação semântica automática baseada em BERT e direcionada a sistemas de QA. O intuito é considerar na identificação semântica todo o texto da pergunta e enriquecê-la com informações que não estão explícitas, mas que podem ser agregadas através de bases de conhecimento. Neste trabalho, foram utilizados recursos da DBpedia [Lehmann et al. 2015]. Com isso, além de auxiliar no contexto semântico das perguntas, auxilia também na localização das possíveis respostas. Para avaliar a abordagem proposta, foi utilizado o conjunto de dados WikiQA [Yang et al. 2015], constituído por perguntas factoides de domínio aberto

[2]http://www.w3.org/TR/2005/WD-rdf-sparql-query-20050721
[3]https://stanfordnlp.github.io/CoreNLP/
[4]https://www.dbpedia-spotlight.org/

e, como *baseline*, foi adotada a Distância de Jaro-Winkler, uma métrica usual na tarefa de EL [Wang et al. 2017].

O restante deste trabalho está organizado como segue: Na seção 2 são discutidos os trabalhos relacionados; a Seção 3 descreve a abordagem proposta para a anotação semântica automática direcionada a sistemas de QA; a Seção 4 descreve como a proposta foi avaliada; e, finalmente, a Seção 5 conclui o trabalho e apresenta possíveis futuros direcionamentos.

2. Trabalhos Relacionados

Com o expressivo volume de dados ligados que vem sendo publicado na *Web*, em sistemas de recuperação de informação, seu uso já é efetivo. Dados ligados podem ser usados para agregar valor ao conjunto de dados original [de Ramos Araújo and de Souza 2011], associando-o a outros conjuntos de dados conexos. Além disso, é possível utilizá-los de diferentes formas e para diferentes propósitos. Com relação às bases externas de conhecimento, elas têm sido utilizadas para apoiar diversas tarefas, tais como, anotação semântica [Dias et al. 2020], alinhamento de entidades [Jain et al. 2010], identificação de contexto [Kawase et al. 2014], recomendação baseada em conteúdo [Dias et al. 2017], dentre outras.

Em [Ma et al. 2021] e [Karpukhin et al. 2020], os autores se preocupam com a construção de sistemas de QA de domínio aberto, baseados em representações vetoriais densas. O objetivo é retornar para cada pergunta uma lista dos documentos mais relevantes (ou seja, com maior probabilidade de conter a resposta). Em ambos os trabalhos, os documentos são divisões de 100 palavras não sobrepostas dos artigos obtidos de uma versão da Wikipedia em inglês. Os trabalhos usam um codificador de perguntas e um codificador de documentos, ambos baseados em BERT. Perguntas e documentos são codificadas como vetores de representação densos. A pontuação de relevância de um documento para uma pergunta é calculada pelo produto escalar. Assim, o problema de recuperar os documentos mais relevantes pode ser visto como o problema de encontrar o vizinho mais próximo no espaço vetorial. Este trabalho igualmente também tem como objetivo buscar candidatos a contexto semântico, por meio de representações vetoriais densas utilizando a rede BERT. No entanto, apesar de considerar as páginas da Wikipedia, considera também os recursos representantes na DBpedia, abrindo a possibilidade de que outras informações, como propriedades específicas, também sejam anexadas às perguntas no processo.

Em [Garg et al. 2020], os autores usam a rede BERT e, com o auxílio de um conjunto de dados, realizam o ajuste fino, para adaptar a rede à tarefa de seleção de frases de respostas (AS2 - *Answer Sentence Selection*). Em seus experimentos, Garg et al. [2020] utilizaram duas coleções, que possuem trechos possíveis de conterem as respostas corretas para as perguntas. Diferentemente, a proposta deste trabalho tenta identificar recursos em uma base de conhecimento, que possam conter a resposta, não necessitando especificar antes os trechos com as possíveis respostas. Além disso, este trabalho também proporciona automaticamente textos relacionados semanticamente à pergunta, por meio desses recursos.

Além disso, trabalhos como [Song et al. 2017] e [Kapanipathi et al. 2021]

têm o objetivo de representar perguntas em linguagem natural (NLQs- *Natural Language Questions*) nos sistemas de QA como padrão de consulta SPARQL, que é uma linguagem padronizada para realizar consultas a grafos RDF. O objetivo desses trabalhos é gerar consultas SPARQL a partir de perguntas em linguagem natural. Porém, para que esse processo se torne possível, são necessárias etapas de análise da pergunta, que muitas vezes envolvem vários aspectos linguísticos, como sintático, léxico e semântico. Liddy [1998] comenta que esses aspectos linguísticos podem ser usados em sua totalidade ou conforme a necessidade. Porém, quanto mais desses aspectos forem considerados, melhor tende a ser o resultado gerado pelo sistema.

Em [Dias et al. 2020], é construída uma análise entre diversas abordagens comuns na literatura que tratam o aspecto linguístico semântico de textos, como NER, EL e representação vetorial esparsa. Porém deixam em aberto o uso de técnicas de anotação semântica baseada em representações vetoriais densas. Neste trabalho, há o intuito de apresentar uma nova técnica utilizando uma abordagem recente na literatura, baseado e representações vetoriais densas, que também pode ser adaptada a outros cenários, como o investigado em [Dias et al. 2020].

3. Abordagem proposta

Como um meio para se chegar à tarefa de selecionar trechos de um texto para compor a resposta de uma pergunta, este trabalho propõe enriquecer a pergunta, por meio da anotação do seu contexto semântico. Como fonte de recursos, usa-se, neste trabalho, o conteúdo estruturado da DBpedia. A anotação do contexto semântico de uma pergunta é feita em três etapas: (1) obtenção de uma *lista finita de candidatos* ao contexto da pergunta e da resposta; (2) obtenção da *representação* da pergunta e dos candidatos; e (3) *escolha do contexto semântico* da lista de candidatos. A mesma estratégia utilizada para representar a pergunta é usada para representar os candidatos.

Inicialmente, pode-se definir uma pergunta factual q como uma sequência de termos $q =< t_1, t_2, t_3, ..., t_n >$, sendo t_j um termo que compõe a pergunta. Os candidatos a contexto semântico, neste trabalho, são recursos da DBpedia, associados a um conjunto de triplas RDF (s, p, o)[5]. Nessas triplas, s e p são URIs, i.e., recursos na *Web*, e o pode ser uma URI ou um literal. s, p e o representam sujeito, predicado e objeto, respectivamente. Assim, a cada recurso s^i tem-se associado um conjunto de pares $\{(p_1^i, o_1^i), \cdots, (p_m^i, o_m^i)\}$. A seguir são descritas as etapas para a definição do contexto semântico, foco principal da abordagem proposta.

Etapa 1 - Seleção de candidatos. Para obter a lista de candidatos a contexto semântico, é necessário desenvolver uma função que recebe como entrada uma pergunta q e uma base de conhecimento, neste trabalho a DBpedia, e fornece como saída um conjunto de recursos candidatos. Para isso, primeiro obtém-se uma lista de no máximo 10 URIs da Wikipedia, através de uma consulta a API da Wikipedia[6]. Em seguida, obtém-se o recurso da DBpedia correspondente a cada uma das URIs da Wikipedia contidas na lista. Por exemplo, para a URI "`https://en.wikipedia.org/wiki/Barack_Obama`" obtém-se o recurso

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

[6]https://en.wikipedia.org/w/api.php

"`https://dbpedia.org/resource/Barack_Obama`". Após isso, tem-se uma nova lista composta por recursos da DBpedia $\{s^1, s^2, \cdots, s^k\}$. Em seguida, para cada recurso s^i da lista de recursos, é submetida uma consulta SPARQL (c_{s^i}) à DBpedia para obter o *abstract* correspondente. A consulta SPARQL submetida tem o formato a seguir:

```
SELECT ?o
WHERE {< s > <http://dbpedia.org/ontology/abstract> ?o.
    FILTER langMatches(lang(?o),'en')}
```

O resultado da consulta SPARQL c_{s^i} é o *abstract* em inglês o^i do recurso s^i. Em seguida o *abstract* é dividido em suas frases componentes, que são processadas e representadas na próxima etapa. Essa divisão em frases tem o objetivo de considerar o máximo possível de contexto. Nesta etapa, optou-se por uma abordagem com código aberto e sem processamentos como o NER e EL, trazendo somente 10 candidatos, por meio da API da Wikipedia. Porém, necessita-se ressaltar que essa etapa pode ser aprimorada por meio de ferramentas que realizam esses processamentos como, por exemplo, Google Cloud Natural Language API[7] e IBM Watson[8]. Além disso, é possível aumentar a cobertura de URI's candidatos da Wikipedia retornadas nesta etapa.

Etapa 2 - Obtenção da Representação. Após a geração da lista de candidatos, é realizada a etapa de obtenção da representação da pergunta e dos possíveis candidatos a contextos. Nesta etapa, o modelo BERT [Devlin et al. 2018] é utilizado como serviço de codificação de sentenças, mapeando uma sentença de comprimento variável em um vetor de comprimento fixo. Nesse processo, são submetidas, como sequências, uma pergunta q e também um *abstract* o^i ou uma frase desse *abstract*, $(f \in o^i)$. Na Seção 4, são avaliadas ambas as possibilidades, tanto o *abstract* quanto uma frase do *abstract*. O objetivo é obter dois vetores densos, um representando a pergunta e outro representando uma das duas opções, o^i ou $(f \in o^i)$. Independente de qual opção, como resultado retornado pela rede BERT, tem-se uma matriz $[max\_seq\_len, 768]$ para cada sequência, possibilitando o uso de uma representação vetorial densa da sequência ou de cada termo individualmente. Nesta etapa, foi utilizado o modelo pré-treinado Google AI BERT-base, com uma arquitetura de rede neural de 110M de parâmetros, de 12 camadas (*layer*), 768 ocultas (*hidden*), 12 cabeças (*heads*). Onde *max_seq_len* é o comprimento máximo a ser considerado da sequência de entrada e 768 são as dimensões dos *embeddings* da BERT. Não foi realizado um *fine-tuning* da rede porque como parte do procedimento de pré-treinamento da BERT, usou-se textos da Wikipedia em inglês (2.500 milhões de palavras) [Devlin et al. 2018]. O que foi considerado domínio de conhecimento suficiente para associar contexto às perguntas factoides de domínio aberto.

Como a parte da sequência que ultrapassa o comprimento definido em *max_seq_len* é desconsiderada e os *abstracts* dos recursos são consideravelmente grandes, optou-se por submeter também as frases do *abstract* separadamente. Essa alternativa torna necessária que todas as frases do *abstract* que corresponde ao recurso passe por essa etapa de representação. Além disso, o comprimento definido em

[7]https://cloud.google.com

[8]https://www.ibm.com/watson

max_seq_len passa a ser fator importante a se investigar dado que interfere diretamente no tamanho da sequência de entrada. Como resultado desta etapa, é utilizada a representação vetorial densa de cada sequência retornada pelo modelo BERT, $\overrightarrow{q}$ e $\overrightarrow{o^i}$ ou $\overrightarrow{f}$.

Etapa 3 - Escolha do contexto semântico. Esta etapa tem o objetivo de definir a distância entre cada pergunta q e cada recurso s^i, $d(q, s^i)$, e, assim, escolher o s^i com a menor distância para ser associado, como contexto, à pergunta q. Para calcular essa distância, indiretamente, por meio do *abstract*, seguem-se os passos a seguir:

1. Para cada par $(\overrightarrow{q}, \overrightarrow{y})$, sendo $\overrightarrow{y}$ a representação do *abstract* $\overrightarrow{o^i}$ ou da frase $\overrightarrow{f}$, calcula-se a distância entre os dois vetores. Neste trabalho, foi usada a distância euclidiana d_e.
   $$d_e\left(\overrightarrow{q}, \overrightarrow{y}\right) = \left[(q_1 - y_1)^2 + ... + (q_n - y_n)^2\right]^{\frac{1}{2}}$$
2. Para distâncias $d_e(\overrightarrow{q}, \overrightarrow{y})$, onde $\overrightarrow{y}$ representa um *abstract*, ou seja, é um $\overrightarrow{o^i}$, faça:
   (a) $d(q, s^i) = d_e(\overrightarrow{q}, \overrightarrow{o^i})$.
   (b) Atribua o s^i com a menor distância à pergunta q como contexto de q.
3. Para distâncias $d_e(\overrightarrow{q}, \overrightarrow{y})$, onde $\overrightarrow{y}$ representa uma frase, ou seja, é um $\overrightarrow{f}$, optou-se por avaliar três situações para a escolha do recurso s^i:
   (a) **Melhor:** seleciona-se a frase $f \in o^i$ com a menor distância euclidiana $d_e(\overrightarrow{q}, \overrightarrow{f})$ para representar s^i, ou seja $d(q, s^i) = min_{f^i_j \in o^i} d_e(\overrightarrow{q}, \overrightarrow{f^i_j})$.
   (b) **Média:** realiza-se a média das distâncias euclidianas entre a representação da consulta q e das frases do *abstract* o^i como distância $d(q, s^i)$, ou seja, $d(q, s^i) = (d_e(\overrightarrow{q}, \overrightarrow{f^i_1}) + d_e(\overrightarrow{q}, \overrightarrow{f^i_2}) + \cdots + d_e(\overrightarrow{q}, \overrightarrow{f^i_n}))/n$, onde n é a quantidade de frases de o^i.
   (c) **Média Top5:** usando as distâncias euclidianas d_e's entre cada par $(\overrightarrow{q}, \overrightarrow{f^i_j})$, onde f^i_j é uma frase de o^i, ordene em ordem crescente as f^i_j's. Seja $< f^{i\prime}_1, f^{i\prime}_2, \cdots, f^{i\prime}_n >$ essas frases ordenadas.
   Faça $d(q, s^i) = (d_e(\overrightarrow{q}, \overrightarrow{f^{i\prime}_1}) + d_e(\overrightarrow{q}, \overrightarrow{f^{i\prime}_2}) + d_e(\overrightarrow{q}, \overrightarrow{f^{i\prime}_3}) + d_e(\overrightarrow{q}, \overrightarrow{f^{i\prime}_4}) + d_e(\overrightarrow{q}, \overrightarrow{f^{i\prime}_5}))/5$.

Após calcular as distâncias $d(q, s^i)$'s entre a consulta q e cada recurso s^i, associa-se a q, como contexto semântico, o s^i com a menor distância $d(q, s^i)$.

4. Avaliação Experimental

Para avaliar a abordagem proposta, foi utilizado o conjunto de dados WikiQA [Yang et al. 2015], um conjunto de pares de perguntas e respostas publicamente disponível, coletado e anotado para pesquisas sobre respostas a perguntas de domínio aberto. O conjunto de dados WikiQA usa os logs de consulta do Bing como fonte de perguntas e foi constituído por registros do período de 1º de maio de 2010 a 31 de julho de 2011. Para a construção, foram selecionadas as consultas que fossem semelhantes a uma pergunta, usando uma heurística simples, como consultas que começaram com uma palavra WH (por exemplo, *"what"* ou *"how"*) e consultas que terminava com um ponto de interrogação.

Focando nas perguntas factoides foram selecionadas apenas as consultas emitidas por pelo menos 5 usuários únicos e com cliques na Wikipedia. Cada pergunta está vinculada a uma página da Wikipedia que potencialmente tem a resposta. Como a seção resumo de uma página da Wikipedia fornece as informações básicas e geralmente mais importantes sobre o tópico, foram utilizadas frases dessa seção como respostas candidatas. Através de *crowdsourcing*, foram incluídas 3.047 perguntas e 29.258 sentenças no conjunto de dados, sendo 1.473 sentenças marcadas como sentenças de resposta às perguntas correspondentes.

Essas 1.473 perguntas do conjunto de dados WikiQA foram submetidas ao processo de seleção de recursos candidatos, representado pela Etapa 1, porém, essa quantidade foi reduzida para 655 perguntas. Isso ocorreu, pois, o método utilizado para elencar possíveis candidatos na etapa experimental estava restrito a somente 10 candidatos e o recurso esperado na base de avaliação não constou entre esses. Como o foco principal do trabalho é avaliar a abordagem semântica, as questões que não tinham a página correta em sua lista candidata foram descartadas sem causar prejuízo ao experimento. Pois, apesar da redução nos dados, manteve-se a diversidade dos tipos de perguntas existentes no conjunto de dados.

Definiu-se como métrica de avaliação a métrica TopN, que fornece o quão bem posicionado está o item relevante. O TopN será máximo, ou seja, igual a 1, se o item relevante está exatamente na primeira posição. Por outro lado, um TopN igual a 0 significa que o item relevante não foi retornado. Admitindo-se que o conjunto de dados WikiQA está correto, a melhor situação é aquela em que a busca obtém TopN máximo.

Para calcular o TopN, os itens retornados foram ordenados considerando o valor da distância. Quanto menor for esse número, mais correlatos estão os recursos e as perguntas, justificando uma melhor colocação no *ranking*. Como a base foi ajustada após o processo de seleção de recursos candidatos a pior situação é um TopN igual a 0.1, pois todas as questões passaram a conter o recurso correto em sua lista de recursos candidatos. Usou-se a fórmula:

$$\text{TopN} = 1 - \left(\frac{k_j - 1}{i}\right)$$

sendo k_j a posição do recurso j, i a quantidade de recursos recuperados no processo de seleção de recursos candidatos representado pela Etapa 1 e, nessa abordagem, o valor de $i \leq 10$, ou seja, um recurso irá contribuir com aproximadamente 0.1 no resultado final se ele ocorrer na décima colocação do resultado ordenado. Como o conjunto de dados usado nos experimentos possuía cada pergunta vinculada a apenas uma página da Wikipedia, métricas como precisão e cobertura podem ser obtidas a partir do resultado de TopN.

A seleção de parâmetros da Etapa 1 ocorreu na criação de uma consulta (*query*) da pergunta, submetida à API da Wikipedia sendo definidos os seguintes parâmetros:

action=query, **generator=search**, **gsrsearch=pergunta**, **format=json**, **gsrprop=snippet**, **prop=info** e **inprop=url**. Essa *query* retorna uma lista de URIs que possuem trechos da pergunta. Recursos da DBpedia

correspondentes dessas URIs retornadas são utilizados como sujeito na geração das consultas SPARQL para retornar o *abstract*.

Para o modelo BERT pré-treinado Google AI[9] foram selecionados os seguintes hiperparâmetros **BERT-Base=**$[Cased, Uncased]$, **12-layer**, **768-hidden**, **110M parameters**, **num_worker=2**, **max_seq_len=** $[NONE, 10, 15, 20, 25, 40, 100]$, onde *NONE* é um tamanho de sequência regulado dinamicamente de acordo com a entrada. Independente do tamanho da sequência original, o retorno é uma matriz $[max\_seq\_len, 768]$ para cada sequência.

Baseline

Como *baseline* para comparar os resultados, foi utilizada a distância Jaro-Winkler. Essa distância é uma métrica usada comumente na tarefa de EL [Wang et al. 2017]. Neste trabalho, utilizou-se a implementação *Jellyfish*[10], uma biblioteca de funções para correspondência aproximada de *strings*. Em [Dimitrakis et al. 2020], é relatado que na etapa de análise da pergunta, um modo de capturar a semântica da pergunta é através do NER e EL. A EL visa vincular todas as menções a entidades após o NER [Wang et al. 2017]. Por esse motivo a distância Jaro-Winkler foi elencada como *baseline*, no intuito de comparar o método proposto neste trabalho com uma abordagem consolidada na literatura.

Experimentos e Resultados

Nos experimentos realizados, foram alterados os parâmetros de *max_seq_len* em $[NONE, 10, 15, 20, 25, 40, 100]$ e foram utilizadas as opções $Cased$ e $Uncased$ do modelo pré-treinado BERT. A Tabela 1 lista a média dos resultados obtidos pela abordagem proposta apenas em uma parte do conjunto de dados, 98 perguntas do tipo *"who"* (PESSOA), aplicadas em todas as variações de parâmetros, utilizando a métrica apresentada na Seção 4. Foram feitos testes utilizando o texto completo do *abstract*, mostrado na Tabela 1 (*abstract*), e realizando algum tipo de tratamento nos textos dos *abstract*. Os tipos de tratamento foram: remoção de *stopwords*, Tabela 1 (sem *stopwords*), e a seleção de setenças substantivas, Tabela 1 (substantivo).

Tabela 1. Média TopN utilizando os textos dos abstracts, com remoção de stopwords e com seleção de setenças substantivas

max_seq_len	NONE	10	15	20	25	40	100
Uncased *abstract*	0,721	0,739	0,765	0,731	0,744	0,697	0,746
Cased *abstract*	0,593	0,692	0,702	0,721	0,731	0,708	0,707
Uncased sem *stopwords*	0,591	0,750	0,753	0,733	0,715	0,664	0,618
Cased sem *stopwords*	0,497	0,709	0,722	0,716	0,723	0,682	0,612
Uncased substantivo	0,675	0,737	0,729	0,708	0,683	0,695	0,706
Cased substantivo	0,478	0,723	0,727	0,726	0,691	0,656	0,578

Observando a Tabela 1, nota-se que, com o hiperparâmetro *max_seq_len* como *NONE*, 15 e 20, os resultados das médias do TopN se destacaram. Com base nisso, foram feitos novos testes utilizando o método de divisão do *abstract* em frases, como

[9]https://ai.google/

[10]https://jellyfish.readthedocs.io/en/latest/comparison.html

descrito na Seção 3. Na Tabela 2, são apresentados os resultados dos testes realizados com o método de divisão do *abstract* em frases e o modelo *Uncased* pré-treinado BERT. Nesse método, é possível obter as várias frases contidas no texto do *abstract*, o que possibilita calcular os melhores recursos de diversas formas, inclusive selecionando os n melhores resultados. Este trabalho avaliou as três diferentes opções de cálculo de distância, descritas na Seção 3.

Tabela 2. Média TopN utilizando os textos das frases dos abstracts e o modelo Uncased

max_seq_len	Melhor	Média	Média Top5
BASELINE	0,655	0,577	0,683
NONE	0,770	0,690	0,794
15	0,773	0,699	0,787
20	0,749	0,670	0,769

Na Tabela 3, são mostrados os resultados dos testes realizados com o método de dividir o *abstract* em frases e o modelo pré-treinado BERT *Cased*.

Tabela 3. Média de TopN utilizando os textos das frases dos abstracts e o modelo Cased

max_seq_len	Melhor	Média	Média Top5
BASELINE	0,655	0,577	0,683
NONE	0,725	0,633	0,763
15	0,763	0,674	0,789
20	0,747	0,643	0,767

Após observar os melhores resultados variando o hiperparâmetro *max_seq_len*, o método de dividir o *abstract* em frases melhor representa a semântica contida no *abstract* de cada recurso, por considerar cada parte individualmente e não descartar grandes trechos durante as variações do *max_seq_len*. Observou-se também uma discreta vantagem nos resultados quando utilizou-se o modelo pré-treinado BERT Uncased. Após a observação dos testes experimentais realizados a partir de perguntas do tipo *"who"* (PESSOA), elegendo então os melhores hiperparâmetros, o melhor modelo pré-treinado para BERT e o método que melhor representa a semântica de cada conceito possível, foram realizados testes que abrangem todos os tipos de pergunta presentes na base de teste. Os resultados obtidos estão dispostos na Tabela 4.

Tabela 4. Média de TopN utilizando os textos das frases dos abstracts e modelo Uncased

max_seq_len	Melhor	Média	Média Top5
BASELINE	0,661	0,619	0,682
NONE	0,764	0,733	0,796
15	0,795	0,758	0,817
20	0,775	0,750	0,803

O melhor valor para o parâmetro *max_seq_len* foi 15. A estratégia de dividir o *abstract* em frases é a melhor opção, uma vez que ela considera todas as informações possíveis de se obter do *abstract* de um recurso e evita que exista um viés em relação a base de avaliação. Os modelos *Uncased* e *Cased* não sofrem grandes variações em relação aos resultados de TopN obtidos para o cenário experimental. Porém, o modelo *Uncased* tem uma discreta vantagem e a estratégia de cálculo da distância usando as frases que considera a média dos top 5 é a que mais se destaca e apresenta resultados superiores com o parâmetro *max_seq_len* 15 e 20.

5. Conclusão e Trabalhos Futuros

Neste trabalho, foi descrita uma abordagem de anotação semântica direcionada a auxiliar sistemas de perguntas e respostas, composta por três etapas: seleção de candidatos, construção de representação e escolha de contexto semântico. O objetivo foi agregar contexto a uma pergunta em um sistema de QA, por meio do recurso de uma base de conhecimento, a DBpedia. Os testes experimentais foram feitos utilizando o conjunto de dados da WikiQA. Os melhores resultados ficaram em torno de 0.80 para a métrica TopN, superando o *baseline* em todos os testes. A melhor estratégia para calcular a distância na escolha do contexto semântico foi a média dos top 5. Acredita-se que a estratégia de dividir o *abstract* em frases elimina possíveis parcialidades dos testes em relação ao conjunto de dados, aumentando as chances de desempenho semelhante em outros conjuntos de dados.

Além disso, acredita-se que a abordagem possa ser utilizada de forma ampla, abrangendo diversos aspectos de um sistema de QA, como, encontrar a resposta da pergunta, auxiliar ou melhorar o resultado de sistemas de QA já existente, ou mesmo ser utilizada como uma etapa para um novo sistema de QA. Apesar dos testes experimentais serem limitados ao cenário de sistemas de QA, acredita-se que a abordagem possa ser adaptada a outros cenários, como anotação semântica ou mesmo reconhecimento de entidades.

Como contribuições do trabalho, está a abordagem de anotação semântica e a avaliação experimental realizada, cuja base anotada está disponível para pesquisas futuras[11]. Como lacunas, este trabalho apresenta a necessidade de um melhor método para buscar os possíveis recursos candidatos na DBpedia, a exploração/variação dos top n mais similares, a necessidade de comparação com mais abordagens de anotação semântica e uma avaliação qualitativa.

Como trabalhos futuros, pretende-se realizar experimentos em outros conjuntos de dados, analisar o impacto do título do recurso no processo, aplicar a abordagem em um sistema de QA e em cenários diferentes, como abordagem de anotação semântica e de reconhecimento de entidades.

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, da Universidade Federal de Ouro Preto (UFOP), do Conselho Nacional de Desenvolvimento Científico e Tecnológico – (CNPq) e da Fundação de Amparo à Pesquisa

[11] https://github.com/LauraLD/GAID-benchmark1

do Estado de Minas Gerais (FAPEMIG). Os autores gostariam de agradecer por apoiarem este trabalho.

Referências

Amaral, D. O. F. d. (2013). O reconhecimento de entidades nomeadas por meio de conditional random fields para a língua portuguesa. Master's thesis, Pontifícia Universidade Católica do Rio Grande do Sul.

Chandurkar, A. and Bansal, A. (2017). A composite natural language processing and information retrieval approach to question answering using a structured knowledge base. *International Journal of Semantic Computing*, 11(03):345–371.

Dasiopoulou, S., Giannakidou, E., Litos, G., Malasioti, P., and Kompatsiaris, Y. (2011). A survey of semantic image and video annotation tools. In *Knowledge-driven multimedia information extraction and ontology evolution*, pages 196–239. Springer.

de Ramos Araújo, L. and de Souza, J. F. (2011). Aumentando a transparência do governo por meio da transformação de dados governamentais abertos em dados ligados. *Revista Eletrônica de Sistemas de Informação*, 10(1).

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dias, L., Barbosa, J., Barrére, E., and De Souza, J. (2017). An approach to identify similarity among educational resources using external knowledge bases. *Brazilian Journal of Computers in Education*, 25(2):18–37.

Dias, L. L., Barrére, E., and de Souza, J. F. (2020). The impact of semantic annotation techniques on content-based video lecture recommendation. *Journal of Information Science*, page 1–13.

Dimitrakis, E., Sgontzos, K., and Tzitzikas, Y. (2020). A survey on question answering systems over linked data and documents. *Journal of Intelligent Information Systems*, 55(2):233–259.

Garg, S., Vu, T., and Moschitti, A. (2020). Tanda: Transfer and adapt pre-trained transformer models for answer sentence selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7780–7788.

Gupta, Y., Saini, A., and Saxena, A. (2015). A new fuzzy logic based ranking function for efficient information retrieval system. *Expert Systems with Applications*, 42(3):1223–1234.

Hirschman, L. and Gaizauskas, R. (2001). Natural language question answering: the view from here. *natural language engineering*, 7(4):275–300.

Jain, P., Hitzler, P., Sheth, A. P., Verma, K., and Yeh, P. Z. (2010). Ontology alignment for linked open data. In *International Semantic Web Conference*, pages 402–417. Springer.

Kapanipathi, P., Abdelaziz, I., Ravishankar, S., Roukos, S., Gray, A., Astudillo, R., Chang, M., Cornelio, C., Dana, S., Fokoue, A., et al. (2021). Leveraging abstract

meaning representation for knowledge base question answering. *Findings of the Association for Computational Linguistics: ACL*.

Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W.-t. (2020). Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Kawase, R., Siehndel, P., Pereira Nunes, B., Herder, E., and Nejdl, W. (2014). Exploiting the wisdom of the crowds for characterizing and connecting heterogeneous resources. In *Proceedings of the 25th ACM conference on Hypertext and social media*, pages 56–65. ACM.

Ko, J., Si, L., and Nyberg, E. (2010). Combining evidence with a probabilistic framework for answer ranking and answer merging in question answering. *Information processing & management*, 46(5):541–554.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., et al. (2015). Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Liddy, E. D. (1998). Enhanced text retrieval using natural language processing. *Bulletin of the Association for Information Science and Technology*, 24(4):14–16.

Ma, X., Sun, K., Pradeep, R., and Lin, J. (2021). A replication study of dense passage retriever. *arXiv preprint arXiv:2104.05740*.

Mohasseb, A., Bader-El-Den, M., and Cocea, M. (2018). Question categorization and classification using grammar based approach. *Information Processing & Management*, 54(6):1228–1243.

Paulheim, H. (2017). Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web*, 8(3):489–508.

Qazi, A. and Goudar, R. (2016). Emerging trends in reducing semantic gap towards multimedia access: A comprehensive survey. *Indian Journal of Science and Technology*, 9(30).

Seaborne, A. and Prud'hommeaux, E. (21 July 2005). Sparql query language for rdf. *W3C Working Draft*.

Shah, A. A., Ravana, S. D., Hamid, S., and Ismail, M. A. (2018). Accuracy evaluation of methods and techniques in web-based question answering systems: a survey. *Knowledge and Information Systems*, 58(3):611–650.

Song, S., Huang, W., and Sun, Y. (2017). Semantic query graph based sparql generation from natural language questions. *Cluster Computing*, 22(1):847–858.

Wang, Y., Qin, J., and Wang, W. (2017). Efficient approximate entity matching using jaro-winkler distance. In *International Conference on Web Information Systems Engineering*, pages 231–239. Springer.

Yang, Y., Yih, W.-t., and Meek, C. (2015). Wikiqa: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2013–2018.