Anotação de entidades em cadeias de consulta

Flávio Nuno Maia de Sousa Filho

Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brasil
fn_maia@outlook.com

Resumo. O SMAPH é uma família de algoritmos de ligação de entidades que pode ser aplicado em uma cadeia de consulta. Para isso, ele se utiliza de técnicas de aprendizado de máquina que são aplicadas sobre resultados de consultas feitas em buscadores atuais. O primeiro passo de todas as versões do algoritmo envolve a realização e o processamento desses resultados de busca. O objetivo deste trabalho é implementar uma versão simplificada do SMAPH, que não utiliza aprendizado de máquina, e comparar a sua eficiência com o anotador de entidades WAT. Assim, será possível simplificar a implementação dessa técnica, permitindo que ela seja usada de maneira mais abrangentes em buscas em geral.

Palavras-Chave: Ligação de entidades, Busca na web, Processamento de HTML.

1 Introdução

1.1 Motivação

Em processamento de linguagem natural, a ligação de entidades é o processo de atribuir significado semântico às entidades mencionadas em um texto. Esse significado semântico permite que informação seja extraída mais facilmente de um texto escrito em linguagem natural, e a maneira mais comum de fazer essa ligação envolve associar os termos do texto a uma base de conhecimento, normalmente a Wikipédia [1]. Um dos problemas dessa técnica é que os algoritmos padrões da área são otimizados para fazerem o processamento de textos longos, que são mais estruturados e contextualizados do que uma cadeia de consulta. Além disso, buscas na internet frequentemente contém termos escritos incorretamente ou fora de ordem.

Devido a essas diferenças, os algoritmos padrões não produzem bons resultados quando aplicados a cadeias de consulta. SMAPH-1 e seus sucessores, SMAPH-S e SMAPH-3, são algoritmos criados para fazer a ligação de entidades nesses caso [2]. Eles funcionam utilizando buscadores da web tradicionais para obter possíveis entidades associadas à cadeia, onde estas são definidas pelo seu nome e por um link da Wikipédia correspondentes, e usam esses resultados junto com técnicas de aprendizado de máquina para determinar corretamente a quais entidades a consulta se refere.

Neste trabalho, o foco será em implementar uma versão simplificada do algoritmo SMAPH. Isso será feito obtendo os resultados diretamente de páginas da web de buscadores, processando o arquivo HTML resultante para obter os dados necessários. Então, será feita a seleção utilizando técnicas heurísticas baseadas nas frequências dos resultados encontrados.

1.2 Geração de Candidatos

A geração de candidatos do SMAPH é feita criando três conjuntos distintos de entidades [2]:

- O primeiro conjunto é formado realizando a consulta no buscador selecionado e, entre os cinco primeiros resultados, são selecionados todos aqueles que são links da Wikipédia.
- O segundo conjunto é formado realizando a consulta com a adição da palavra "wikipedia" e, entre os dez primeiros resultados, são selecionados todos aqueles que são links da Wikipédia.
- O terceiro conjunto é formado obtendo os trechos de texto dos 15 primeiros resultados da consulta original, passando cada um desses trechos por um ligador de entidades, e escolhendo entre as entidades encontradas somente aquelas que estavam marcadas em negrito no trecho.

O conjunto final de candidatos é a união desses três conjuntos. Este primeiro passo é comum às três variações do SMAPH. Neste trabalho, o mecanismo de busca escolhido será o DuckDuckGo [3], e o ligador de entidades será o WAT Entity Annotator [4]. Apenas o último conjunto será utilizado, pois os conjuntos anteriores não contribuíram para resultados melhores com o algoritmo simplificado. Até o momento, o WAT só suporta textos em inglês, portanto as consultas também deve ser em inglês.

2 Trabalhos Relacionados

O SMAPH, apesar de ser focado na ligação de entidades de cadeias curtas de consultas, depende de técnicas de ligação de entidades usuais para gerar seus candidatos [2]. Usado na fase de geração de candidatos, o ligador de entidades WAT é uma evolução do TagME, que foi desenvolvido com o intuito de de anotar textos relativamente curtos [5]. Por isso, ele é ideal para fazer a anotação de trechos de texto retornados por buscadores.

O funcionamento do WAT é baseado em três passos: identificação, desambiguação, e poda [6]. O primeiro passo, a identificação, consiste em detectar quais termos no texto são possíveis entidades. Para isso, é feito um pré-processamento dos artigos da Wikipédia, que indexa as páginas usando os termos que são links para elas. Note que um termo pode indexar mais de uma página diferente. Então, são encontrados quais termos no texto corresponde a um termo indexador.

O segundo passo é a desambiguação. Neste estágio, para aqueles termos que possuem mais de uma possível anotação é feita a escolha de quais são apropriadas para o contexto atual. Para cada link de um termo, é calculado um valor de acordo com a relação dele com todos os outros possíveis significados de todos os outros termos, e o que tiver o maior valor resultante é escolhido para aquele termo.

O último passo é a poda. Neste passo, são selecionados entre os termos escolhidos, e são eliminados aqueles que não são relevantes. Isso é feito escolhendo entre as anotações escolhidas na desambiguação quais ocorrem mais comumente dentro da própria Wikipédia, e são mantidas apenas essas e as que têm um alto valor de relação com elas. Assim existe maior garantia de que todas as entidades encontradas no texto são relacionadas ao contexto do texto original.

3 Implementação

A implementação consistirá de uma versão simplificada do algoritmo SMAPH. A geração de candidatos será baseada apenas no terceiro conjunto de candidatos dele. A partir daí, será selecionado para cada termo qual das anotações melhor corresponde ao contexto da busca. Porém, ela não será feita com base em aprendizado de máquina, mas sim aplicando uma heurística simplificada.

Primeiro, é feita a busca do termo no buscador escolhido. A página resultante em HTML é então analisada em busca de duas informações: os trechos de texto dos primeiros resultados de pesquisa e pela sugestão de correção que o buscador apresenta caso encontre erros de digitação. Como esses erros são comuns, é importante que eles sejam corrigidos antes da anotação, e o próprio buscador já realiza esse processo. É então feito um mapeamento entre os termos corrigidos e os originais.

Os trechos de texto são então processados individualmente pelo WAT, a partir de sua API. Dentre as anotações retornadas, são mantidas apenas aquelas que anotam um dos termos presente na busca corrigida, e as restantes são descartadas. Anotações que incluem múltiplos termos são incluídas para cada um dos termos de busca que está presente entre esses termos.

Em seguida, para cada termo presente é selecionada a melhor anotação que corresponde àquele termo no contexto da busca. Isso é feito selecionando a anotação que apareceu mais vezes entre todas as encontradas e, em caso de empate, é selecionada a anotação que foi encontrada primeiro.

Por fim, é feita a união dos termos que correspondem à mesma entidade. Para isso, partindo do primeiro termo da busca é verificado se o próximo termo foi anotado com o mesmo link, e no caso positivo, ambos serão tratados como o mesmo termo. Isso acontece até ser encontrado um termo anotado com um link diferente. O processo então recomeça partindo desse novo termo, até todos os termos terem sido checados.

4 Experimentação

Para testar a eficiência das anotações, será feita a comparação entre os resultados do WAT aplicado na cadeia de consulta e os do algoritmo simplificado. Apesar de ser mais simples que o SMAPH, o algoritmo ainda deve ser capaz de realizar a ligação de entidades com maior precisão do que o WAT, já que este não é otimizado para esse caso de uso.

A técnica de comparação usada consiste em contar o número de termos anotados corretamente em comparação com uma anotação de teste feita previamente [7], incluindo termos não anotados, e calcular a porcentagem de acertos. Isto é feito tanto no resultado do WAT quanto do algoritmo próprio, assim podendo ver numericamente a diferença entre eles.

Tabela 1. Tabela mostrando as diferentes anotações.

Consulta: Armstrong mon lading	
Base	
Armstrong	Neil_Armstrong
mon lading	Moon_landing
WAT	
Armstrong	Armstrong_Whitworth
Algoritmo	
Armstrong	Neil_Armstrong
mon	Moon

Tabela 2. Porcentagem de acerto dos dois algoritmos.

Consulta	WAT	Algoritmo
Armstrong mon lading	0%	33%
Barak Obama mandate lenght	100%	100%
WAT api token	33%	66%
seven wonders of the world	40%	80%
best cheap cat food	100%	100%
c book default	33%	66%
optimizer programming in animal feeding	0%	25%
large company payroll service providers	0%	0%
photos of starry night	50%	25%
metronome setting of allegro	100%	100%

A tabela 1 mostra um exemplo dos resultados do WAT e do algoritmo, além da anotação verdadeira. A tabela 2 mostra as porcentagens de acertos de ambos para algumas cadeias.

Baseado nesses resultados, pode-se ver que em média o algoritmo possui uma taxa de acerto de 59,5%, enquanto o WAT obteve apenas 45,6%. Portanto, a melhora obtida foi de 30,5%. Porém, como a quantidade de cadeias de consulta testadas foi relativamente pequena, ainda não é possível concluir que a melhoria em relação ao WAT seja geral, mas ainda assim o resultado preliminar é promissor.

5 Trabalhos Futuros

Para os próximos trabalhos, será feito um refinamento da técnica de seleção de entidades, tratando alguns erros que foram encontrados, como *stopwords* classificadas incorretamente, entidades compostas fora de ordem, e variações de palavras além de erros de digitação.

No futuro também se planeja fazer uma comparação estatística mais abrangente da precisão do algoritmo em relação aos anotadores tradicionais, fazendo uso de bancos de dados contendo diversas anotações. Assim será possível se certificar de que a melhoria aparente dos resultados se aplica em casos mais gerais, não apenas em algumas cadeias específicas.

6 Conclusões

A ideia de realizar anotação de entidades em cadeias de consultas é promissora, e o algoritmo SMAPH mostra que é possível fazer isso de maneira eficiente. Este trabalho foi uma tentativa de desenvolver uma técnica mais simples que atinge o mesmo objetivo, sem precisar aplicar aprendizado de máquina.

Seguindo a ideia do SMAPH de usar o poder dos buscadores da Web aliados a um anotador de entidades tradicional, neste caso o WAT, foi possível obter uma melhoria na anotação das cadeias de consulta em relação a esses. Isto foi alcançado utilizando um algoritmo simples baseado na frequência das anotações encontradas nos trechos de texto retornados pelo buscador da Web escolhido.

References

1. Xianpei Han, Le Sun, and Jun Zhao. 2011. Collective entity linking in web text: a graph-based method. In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval (SIGIR '11). Association for Computing Machinery, New York, NY, USA, 765–774. DOI:https://doi.org/10.1145/2009916.2010019.
2. Marco Cornolti, Paolo Ferragina, Massimiliano Ciaramita, Stefan Rüd, and Hinrich Schütze. 2018. SMAPH: A Piggyback Approach for Entity-Linking in Web Queries. ACM Trans. Inf. Syst. 37, 1, Article 13 (January 2019), 42 pages. DOI:https://doi.org/10.1145/3284102.
3. DuckDuckGo, https://duckduckgo.com, último acesso 14/11/2020.
4. WAT API Documentation, https://sobigdata.d4science.org/web/tagme/wat-api, último acesso 14/11/2020.
5. Francesco Piccinno and Paolo Ferragina. 2014. From TagME to WAT: a new entity annotator. In Proceedings of the first international workshop on Entity recognition & disambiguation (ERD '14). Association for Computing Machinery, New York, NY, USA, 55–62. DOI:https://doi.org/10.1145/2633211.2634350.
6. Paolo Ferragina and Ugo Scaiella. 2010. TAGME: on-the-fly annotation of short text fragments (by wikipedia entities). In Proceedings of the 19th ACM international conference on Information and knowledge management (CIKM '10). Association for Computing Machinery, New York, NY, USA, 1625–1628. DOI:https://doi.org/10.1145/1871437.1871689.
7. BAT-framework, https://github.com/marcocor/bat-framework, último acesso 14/11/2020.