도메인 지식을 이용한 랩퍼에서 규칙 생성 정확도 향상

정창후* 서정현* 류범종* 맹성현**
한국과학기술정보연구원*, 한국정보통신대학원대학교**
{chjeong, jerry, ybj}@kisti.re.kr,
myaeng@icu.ac.kr

Improving Rule Generation Precision for Wrappers using Domain Knowledge

Chang-Hoo Jeong*, Jerry Hyeon Seo*, Beom-Jong You*, Sung-Hyon Myaeng**
Korea Institute of Science and Technology Information*,
Information and Communications University**

요 약

기존의 도메인 지식 기반의 랩퍼 학습 방법은 도메인에 대한 정보를 바탕으로 해당 정보 소스에 대한 랩퍼를 생성한다. 응용 분야에 맞게 정의된 도메인 지식을 이용함으로써 정보 소스에서 제공하는 다양한 텍스트의 의미와 형태를 이해할 수 있다. 그러나 정보 소스에서 제공되는 모든 텍스트에 의미 인식의 근거가 되는 레이블이 붙어서 제공되는 것이 아니기 때문에, 도메인 지식만을 이용해서 랩퍼를 학습하는 방법은 한계에 부딪힐 수 밖에 없다. 이러한 문제를 해결하기 위해서 본 논문은 인터넷에 존재하는 다양한 웹 정보 소스에서 효율적이고 정확하게 랩퍼를 생성할 수 있도록 하는 도메인 지식 기반의 확률적 랩퍼 생성 시스템을 제안한다. 효율적이고 정확한 랩퍼 생성 시스템을 구축하기 위해서 도메인 지식뿐 아니라 상세 정보로 연결되어 있는 하이퍼링크와 엔티티 인식을 위한 확률 모델을 이용하였다. 이렇게 여러 가지 방법을 적용함으로써 사용자의 개입없이 다양한 정보 소스에 대해서 보다 추출 성능이 좋은 랩퍼를 생성할 수 있다.

1. 서 론

웹에서의 데이터 추출의 문제를 다루는 보편적인 접근법은 다양한 데이터 소스에 접근하는 이질성을 캡슐화하는 랩퍼를 작성하는 것이다. 랩퍼(Wrapper)는 특정한 정보 소스에 대해서 관심있는 데이터의 위치와 구조 포맷 등을 나타내는 추출 규칙이라고 정의할 수 있다. 기존의 도메인 지식 기반의 랩퍼 학습 방법은 도메인에 대한 정보를 바탕으로 해당 정보 소스에 대한 랩퍼를 생성한다. 도메인 지식(Domain Knowledge)은 해당 도메인에서 유용하게 사용될 수 있는 정보, 즉 중요한 속성 정보를 나타내는 엔티티와 엔티티의 구성 형태(포맷)를 기술하는 역할을 한다. 응용 분야에 맞게 정의된 도메인 지식을 이용함으로써 시스템은 정보 소스에서 제공하는 다양한 텍스트의 의미를 이해할 수 있고, 구조 또한 자동으로 감지할 수 있다는 장점이 있다. 그러나 정보 소스에서 제공되는 모든 텍스트에 인식의 근거가 되는 레이블이 붙어서 제공되는 것이 아니기 때문에 도메인 지식만을 이용해서 랩퍼를 학습하는 방법은 한계에 부딪힐 수 밖에 없다.

본 연구에서는 인터넷에 존재하는 준구조화된 웹 정보 소스에서 효율적이고 정확하게 정보를 추출하도록 하는 도메인 지식 기반의 확률적 랩퍼 생성 시스템에 관한 연구에 주안점을 둔다. 따라서 레이블이 없이 나오는 텍스트들에 대해서 해당 텍스트의 엔티티를 자동으로 인식할 수 있는 확률적 방법에 대한 모델을 새롭게 제안하고자 한다. 이것은 도메인 지식 기반의 랩퍼 생성과 마찬가지로 인간의 개입을 최소로 요구하기 때문에 실세계의 응용에 보다 편리하게 적용시킬 수 있을 뿐만 아니라, 도메인 지식 기반의 랩퍼 생성 시스템이 수행하지 못하는 단서가 없는 텍스트에 대해서도 엔티티 인식을 효과적으로 수행하게 된다.

2. 관련 연구

자동 랩퍼 생성은 주로 기계 학습 기술을 사용하는데, 랩퍼 연구 커뮤니티에서는 매우 간단한 것부터 상대적으로 복잡한 것까지 랩퍼의 생성을 위한 다양한 학습 알고리즘을 개발해 왔다. 알고리즘을 이용한 랩퍼의 자동 생성이 전문가의 개입을 최소로 요구한다고 하지만, 이러한 방법은 보통 학습 단계를 거쳐야 하기 때문에 시간이 많이 소요될 수 있다.

자동 랩퍼 생성의 연구로는 추출 가능한 정보 소스의 클래스(타입)를 구분해 놓고서 어떤 클래스에 속하는 지를 학습하는 방법[1]과 도메인 지식 기반의 학습 방법[2,3]이 있다. 이러한 자동 랩퍼 생성은 학습 문서에 의해서 규칙이 생성되고 많은 경우에 이러한 학습은 통제하에 수행되어야만 한다. 사용자는 페이지 집합의 관심있는 데이터에 표식을 붙이고 시스템은 이러한 예제 문서에 기반하여 추출 규칙을 배운다. 이렇게 생성된 랩퍼에 있어서 추출 규칙의 정확성은 학습에 사용된 예제 문서의 수와 질에 의존한다.

3. 랩퍼 생성

랩퍼 생성의 추출 성능 향상을 위해서 고려해 볼 수 있는 요소로는 다음과 같이 크게 두 가지가 있다.

3.1 하이퍼링크 활용 방법

많은 웹 정보 소스가 사용자에게 정보를 제공할 때, 처음에는 간략 정보만을 제공하는 방식을 취하고 있다. 그리고 나서 해당 아이템의 상세 정보를 보기를 원했을 경우에만 하이퍼링크로 연결되어 있는 상세 정보를 보여주도록 한다. 이러한 방법은 사용자가 원하는 정보를 대략적으로 빨리 훑어볼 수 있게 해주는 장점이 있다. 또한 사용자가 처음에 접속하는 페이지에 많은 정보를 주기 위해서는 데이터베이스에서 한 번에 모든 정보를 가져와서 웹 페이지를 생성해야 하는데, 이럴 경우에 정보를 생성하도록 하는 웹 프로그램의 동작 시간이 상당히 길어질 수 있다. 이것은 사용자가 정보 소스에 접속할 때 초기 접속 시간을 길어지게 하기 때문에, 사용자에게

서비스에 대한 불편을 초래할 수 있다. 따라서 아이템에 대한 충분한 정보를 얻기 위해서는 이러한 웹의 특성을 고려하여 하이퍼링크에 연결되어 있는 정보를 잘 활용하여야 한다.

하이퍼링크를 이용하기 위한 방법은 다음과 같다.

- 랩퍼 생성시
  - 첫 페이지에서 제공되는 정보들의 패턴을 분석하여 각 아이템의 바운더리를 감지한다.
  - 감지된 바운더리 안에 있는 모든 하이퍼링크를 쫓아가서 유용한 정보가 있는 지를 확인한다. 도메인 지식을 이용해서 인식된 엔티티의 개수가 가장 많은 문서가 유용한 문서이다.
  - 하이퍼링크의 정보가 유용하다고 판단되면 링크의 위치와 발견된 엔티티 관련 정보를 통합하여 랩퍼에 기록한다.
- 정보 추출시
  - 랩퍼를 읽어 들여 하이퍼링크에서 정보를 추출해야 하는 지를 결정한다.
  - 처음 페이지에서 정보를 추출하고, 하이퍼링크의 정보 추출 표시가 있으면 하이퍼링크에 연결된 페이에서도 정보를 추출한다.
  - 첫 페이지(Front page)에서 정보를 추출한 것과 하이퍼링크에 연결된 페이지(Back end page)에서 정보를 추출한 것을 하나의 아이템 단위로 합쳐서 통합된 추출 정보를 생성한다.

이와 같이 하이퍼링크에 포함되어 있는 정보를 분석해서 이용함으로써 정보 소스에서 얻을 수 있는 유용한 엔티티의 개수를 증가시킬 수 있다.

3.2 확률 정보 활용 방법

정보 소스에서 랩퍼를 생성할 때, 레이블을 가지고 있는 텍스트는 도메인 지식에 의해서 자동으로 인식되게 된다. 그러나 레이블을 가지고 있지 않는 텍스트는 도메인 지식을 이용한다고 하더라도 해당 텍스트에 대한 의미를 이해할 수 있는 단서가 없기 때문에, 텍스트에 대한 엔티티를 인식할 수가 없게 된다. 본 연구에서는 이렇게 인식되지 않는 텍스트의 의미를 이해하기 위해서 확률적인 방법을 새롭게 모델링하고자 한다.

확률 모델에서 사용하는 기호 t와 e는 각각 이미 인식된 토큰과 할당된 엔티티를 나타내고, t'와 e'은 각각 아직 인식되지 않은 토큰과 할당되지 않은 엔티티를 나타낸다. 또한 t와 t'의 대문자 T와 T'은 같은 역할을 수행하는 여러 개의 토큰이 모여서 구성된 토큰 집합을 나타낸다.

확률적 방법의 첫 번째 모델은 베이지언 모델을 이용하는 것이다. 베이지언 모델은 조건부 확률을 이용하는 방법으로서, 레이블이 없어서 인식되지 않는 토큰이 있을 때 토큰을 어떠한 엔티티로 식별하는 게 옳은 것인가를 결정하기 위해서 기존에 어떤 엔티티에서 어떤 토큰들이 나왔나를 역으로 관찰하는 방법이다. 단, 이때 웹페이지에서 출력되는 페이지에 여러 개의 아이템이 존재하기 때문에 하나의 토큰만을 고려하는 것이 아니라 각 아이템에 대해서 같은 위치에 나오는 모든 토큰들을 합쳐서(토큰 집합을 구성해서) 고려하도록 한다. 하나의 토큰이 어떤 엔티티로 식별되는 확률을 계산하는 것보다는 같은 성격을 가지고 있는 여러 개의 토큰이 어떤 엔티티로 식별되는 확률을 계산하는 것이 좀더 변별력있는 확률을 구할 수 있기 때문이다. 이러한 개념을 이용하면 정보 소스의 아이템에 대해서 레이블이 없어서 식별되지 않는 토큰들을 확률 값을 이용하여 새로운 엔티티로 할당할 수 있게 된다.

$$P(e'_i \mid t'_j) = \frac{P(t'_j \mid e'_i) * P(e'_i)}{P(t'_j)} \cong P(e'_i) * P(t'_j \mid e'_i) \quad \text{--- ①}$$

$$P(e'_i \mid T'_j) \cong P(e'_i) * P(T'_j \mid e'_i) \cong P(e'_i) * \frac{1}{\nu} \sum_{k=1}^{\nu} P(t'_{jk} \mid e'_i) \quad \text{--- ②}$$

첫 번째 모델에서 토큰이 엔티티에 속할 확률 값은 ①과 같이 계산한다. 그러나 정보 소스에 여러 개의 아이템이 존재하기 때문에, 토큰이 엔티티에 속할 확률 값보다는 토큰 집합이 엔티티에 속할 확률 값을 계산하는 것이 보다 신뢰성있는 정보를 얻을 수 있다. 따라서 ②와 같이 계산하도록 한다.

확률적 방법의 두 번째 모델은 컨텍스트 정보를 이용하는 것이다. 하나의 아이템 안에 같이 존재하는 주변 정보들을 이용하는 방법으로서, 레이블이 없어서 인식되지 않은 토큰이 있을 때 토큰을 어떠한 엔티티로 식별하는 게 옳은 것인가를 결정하기 위해서 토큰과 같은 아이템에 속해 있는 레이블이 있는 텍스트 정보를 이용하는 방법이다. 도메인 지식에 의해서 이미 인식된 텍스트 정보를 이용하면 인식되지 않은 토큰의 레이블을 추정해 볼 수 있기 때문이다. 이것은 기존에 추출되었던 아이템들이 관련 데이터를 가지고 있기 때문에 적용이 가능하다. 즉, 여러 정보 소스에 대해서 랩퍼를 생성하고 정보를 추출하도록 하고 있기 때문에 다른 정보 소스에서 추출된 정보를 이용하여 현재 사이트에서 문제가 되고 있는 것들을 해결할 수 있다.

$$\begin{aligned} &P(e'_i = t'_j \mid \{e_1 = t_1 \,\&\, e_2 = t_2 \,\&\, \ldots \,\&\, e_n = t_n\}) \\ &= \{P(e'_i = t'_j \mid e_1 = t_1) * P(e_1 = t_1)\} \\ &+ \{P(e'_i = t'_j \mid e_2 = t_2) * P(e_2 = t_2)\} \\ &+ \ldots \\ &+ \{P(e'_i = t'_j \mid e_n = t_n) * P(e_n = t_n)\} \\ &= \sum_{h=1}^{n} P(e'_i = t'_j \mid e_h = t_h) * P(e_h = t_h) \end{aligned} \quad \text{--- ③}$$

$$P(e'_i = T'_j \mid \{e_1 = T_1 \,\&\, e_2 = T_2 \,\&\, \ldots \,\&\, e_n = T_n\}) \cong \frac{1}{\nu} \sum_{k=1}^{\nu} \sum_{h=1}^{n} P(e'_i = t'_{jk} \mid e_h = t_{hk}) * P(e_h = t_{hk}) \quad \text{--- ④}$$

두 번째 모델에서 토큰이 엔티티에 속할 확률 값은 ③과 같이 계산한다. 그러나 첫 번째 모델에서와 같은 이유로 실제적으로는 ④와 같이 계산하도록 한다.

확률적 방법의 세 번째 모델은 첫 번째 모델과 두 번째 모델을 결합한 것이다. 첫 번째 모델과 두 번째 모델이 나름대로의 타당성있는 가치를 지니기는 하지만, 각각 상대적인 가중치를 두어 두 가지 방법을 혼합함으로써, 좀 더 신뢰성있는 그리고 여러 가지 정보가 혼합된 견고한 모델을 구성할 수 있다.

첫 번째 모델의 확률 값을 P_1, 두 번째 모델의 확률 값을 P_2, 첫 번째 모델이 두 번째 모델에 비해서 상대적으로 중요한 정도(비율)를 α라고 하면, 세 번째 모델의 확률 값은 $(\alpha * P_1) + ((1 - \alpha) * P_2)$가 된다. 첫 번째 모델과 두 번째 모델의 상대적인 중요도 값을 나타내는 α 값은 도메인별 실험을 통하여 적당한 값으로 추정하도록 한다.

4. 실험

4.1 실험 방법

본 논문에서 제안한 몇 가지 방법들의 유용성을 검증하기 위해서 실험을 단계적으로 수행하였다. 즉, 처음에는 도메인 지식만을 적용하여 랩퍼를 생성하도록 하였고, 다음에는 하이퍼링크에 대한 처리를 추가하여 랩퍼를 생성하도록 하였다. 마지막으로 본 논문에서 가장 중요하게 생각하는 인식되지 않은 토큰들에 대한 엔티티 인식 알고리즘을 적용하여 랩퍼를 생성하여 그 결과를 비교하였다.

각 사이트의 추출 성능의 정확도는 다음과 같이 계산된다.

정확도(P) = (추출된 엔티티의 개수 / 추출해야 될 엔티티의 개수) * 100

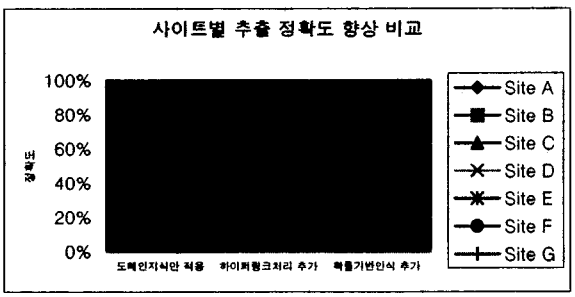
여기서 추출된 엔티티의 개수는 랩퍼를 학습하면서 인식된 엔티티의 개수라고 볼 수 있고, 추출해야 될 엔티티의 개수는 해당 도메인에서 정의한 엔티티의 개수라고 볼 수 있다.

전체 사이트에 대한 평균 정확도는 다음과 같이 계산된다.

평균 정확도(Global Precision) = (각 사이트의 정확도 / 평가를 수행한 사이트의 수)

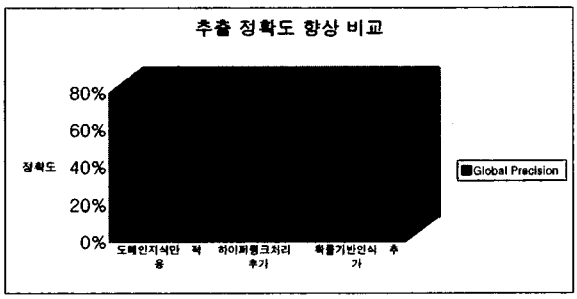
4.2 결과 및 분석

사이트별 추출 성능 향상에 대한 결과는 [그림 1]과 같다.



[그림 1] 사이트별 추출 정확도 향상 비교

전체적인 추출 성능 향상에 대한 결과는 [그림 2]과 같다.



[그림 2] 전체적인 추출 정확도 향상 비교

첫 번째 실험에서는 도메인 지식만을 적용하여 랩퍼를 생성하도록 하였다. 실험 결과 해당 정보 소스에서 추출할 수 있는 엔티티들에 대해서 적절하게 랩퍼를 생성하는 것을 관찰할 수 있었다. 그러나 이러한 방법은 웹 사이트가 가지고 있는 하이퍼링크의 유용성을 제대로 활용하지 못한 결과를 초래하였다. 따라서 추출할 수 있는 엔티티의 수에 많은 제약이 있다고 볼 수 있다.

두 번째 실험에서는 하이퍼링크에 대한 처리를 수행하여 랩퍼를 생성하도록 하였다. 실험 결과 일부 정보 소스에서 추출할 수 있는 엔티티의 수가 배가 넘게 증가하는 것을 관찰할 수 있었다. 이것은 웹 사이트의 구조적 특성을 감안하여 하이퍼링크에 대한 처리를 수행했기 때문이라고 보여진다. 웹을 필두로 한 인터넷의 발전에 가장 크게 기여한 요소가 하이퍼링크라고 말하는 경우가 많은데, 실제적으로 웹 정보 소스를 기반으로 한 랩퍼 생성 시스템에서도 이러한 하이퍼링크의 특성을 이용하는 것이 효과적임을 살펴볼 수 있었다.

세 번째 실험에서는 인식되지 않은 토큰들에 대해서 엔티티 인식 알고리즘을 적용하여 랩퍼를 생성하도록 하였다. 실험 결과 일부 정보 소스에서 추출할 수 있는 엔티티의 수가 증가하는 것을 관찰할 수 있었다. 이것은 레이블이 없는 토큰들에 대해서 확률적 방법을 적용해서 엔티티 인식을 수행한 방법이 적절했다는 것을 보여준다.

여기서 새롭게 인식된 엔티티의 성격을 살펴 볼 필요가 있다. 타이틀과 같은 정보는 어느 도메인에서 사용되든지 간에 항상 존재해야만 하는 핵심 엔티티라고 볼 수 있는데, 이러한 정보들이 추출되지 않으면 정보 추출은 자칫 무의미한 작업이 될 수도 있다. 그러나 많은 정보 소스에서 타이틀과 같은 중요한 정보에 레이블을 주지 않는 경우가 상당수 발견되고 있다. 이것은 타이틀과 같이 중요한 정보에 대해서는 텍스트의 폰트를 키우거나 색깔을 화려하게 부각시켜서 가장 중심적인 내용이라는 것을 알려주려고 하기 때문이다. 그리고 타이틀과 같이 아이템을 구별하는 정보로 사용되는 엔티티는 사용자의 직관에 의해 쉽게 인지될 수 있기 때문에, 굳이 레이블을 붙이지 않는 이유도 있다고 보여진다. 이와 같은 상황에서 확률 기반의 엔티티 인식 방법을 사용하면, 이전 단계까지는 인식이 되지 않았던 정보 소스의 아이템에 있어서 핵심적인 역할을 수행하는 타이틀을 효과적으로 인식하는 것을 살펴볼 수 있다.

5. 결론 및 향후 연구

결과적으로 도메인 지식을 이용하여 랩퍼를 생성하는 시스템은 그 나름대로 많은 장점을 가지고 있음에도 불구하고 레이블이 없는 텍스트 인식에 있어서는 치명적인 약점을 가지고 있기 때문에, 확률적인 방법을 적용한 랩퍼 생성 시스템은 그 중요성이 아주 크다고 볼 수 있다. 더군다나, 확률적인 방법을 적용해서 새롭게 인식하고 있는 텍스트의 대부분이 해당 아이템의 이름이 될 수 있는 타이틀 역할의 엔티티가 많다는 점은 본 연구에서 제시한 방법론이 아주 유용하고 효과적이었다는 것을 입증하고 있다.

향후 연구로는 정보 소스별로 추출된 개별적 결과의 통합에 관련된 작업과 규칙 생성의 정확도 향상 관점에서 다루어질 수 있는 자동 도메인 지식의 확장에 관련된 작업이 이루어져야 할 것이다.

6. 참고 문헌

[1] N. Kushmerick, D. Weld, and R. Doorenbos, "Wrapper Induction for information extraction", International Joint Conference on Artificial Intelligence (IJCAI), Nagoya, Japan, 1997.

[2] H. Seo, J. Yang, and J. Choi, "Knowledge-based Wrapper Generation by Using XML", IJCAI-2001 Workshop on Adaptive Text Extraction and Mining (ATEM 2001), pp. 1-8, Seattle, USA, 2001.

[3] 서희경, 양재영, 최중민, "준구조화 정보소스에 대한 지식기반 Wrapper 학습 에이전트", 정보과학회 논문지: 소프트웨어 및 응용, 29권 1-2호, pp. 42-52, 2002.