

OpenAPI와 웹서비스의 통합 모델을 통한 효율적인 서비스 검색 시스템 개발¹⁾

천동석* 차승준*, 김경옥**, 이규철*[†]
*충남대학교 컴퓨터공학과
**한국전자통신연구원
{ikarus1004, junii, kcleo}@cnu.ac.kr
kokim@etri.re.kr

Development of Efficient Service Retrieval System for the Integrated Model of OpenAPI and Web Service

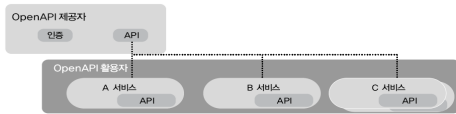
Dong-Suk Chun*, Seung-Jun Cha*, Kyong-Ok Kim**, Kyu-Chul Lee*[†]
*Department of Computer Engineering, Chungnam National University
**Electronics and Telecommunications Research Institute

요 약

오늘날 웹에서는 OpenAPI, 웹서비스등 보급이 증가함에 따라 많은 양의 서비스로부터 적합한 서비스들을 발견해 내는 문제는 중요한 문제로 부각되고 있다. 본 논문에서는 OpenAPI와 웹서비스의 통합 모델을 제안하고 이 통합 모델에 알맞은 검색 알고리즘을 적용하여 효율적인 서비스 검색을 사용자에게 제공한다. 사용자는 키워드 질의를 통해 순위화 된 서비스 검색 결과를 제공받을 수 있다.

1. 서론

OpenAPI는 자사의 API를 외부에 공개한 것으로 일반적으로 웹서비스(Web Services)형태로 공개한 것을 말하는 것으로 “응용 프로그램에서 사용할 수 있도록 운영 체제나 프로그래밍 언어가 제공하는 기능을 제어할 수 있도록 만든 인터페이스”로 정의 된다.



(그림 1) OpenAPI의 개념

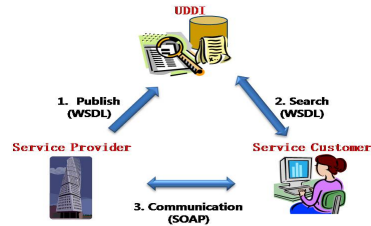
OpenAPI는 SOAP(Simple Object Access Protocol), 자바스크립트 등 웹 기술에 의해 웹 사이트들이 서로 상호작용을 할 수 있도록 하는 기술로 이루어져 있다. OpenAPI의 기본적인 개념은 (그림1)과 같다. OpenAPI 제공자는 OpenAPI 활용자들에게 각각 인증키를 분배하고 활용자들은 그 인증키를 소유함으로써 제공자의 OpenAPI를 사용할 수 있게 되는 것이다.

1) 본 연구는 건설교통부 첨단도시기술개발사업 - 지능형 국토정보기술혁신 사업과제의 연구비지원(07국토정보C05)에 의해 수행되었습니다.

본 연구는 지식경제부 및 정보통신연구진흥원의대학 IT 연구센터 육성·지원사업 (IITA-2008-C1090-0801-0031)의 연구결과로 수행되었음

[†] 교신저자

웹서비스는 네트워크상에서 서로 다른 종류의 컴퓨터들 간에 상호작용을 하기 위한 소프트웨어 시스템이다[1]. 웹서비스는 WSDL(Web Service Description Language)로 기술된다. WSDL에는 웹서비스의 위치, 제공하는 연산과 입력 및 출력 파라미터와 같은 웹서비스에 대한 정보가 들어있다.



(그림 2) 웹서비스 모델

다음 (그림 2)은 웹서비스의 모델을 그림으로 표현한 것이다. 1번은 서비스를 외부 저장소(repository)에 발행하는 서비스 제공자를 나타낸다. 한편 서비스가 레지스트리에 등록되면 서비스는 클라이언트에 의해서 사용될 수 있다. 2번의 의미하는 바는 서비스를 저장소로부터 찾고(look up) 서비스에 관련된 정보를 되돌려주는 것(retrieve)을 나타내고 있다. 프로시저 요청 포맷에 대한 정보와 서비스 제공자의 위치는 일반적으로 다른 세부 사항에 의해서 제공되어 진다. 3번은 서비스에 바인딩(binding)하는 클라이언트와 서비스가 제공하는 기능들을 사용하기 위해 클라이언트가 서비스에 접근하는 과정을 나타낸다.

오늘날 웹에는 OpenAPI, 웹서비스등 많은 서비스들을 제공하고 있다. 이로 인해 사용자들은 많은 서비스들로부터

터 다양한 혜택을 받을 수도 있지만 제대로 된 검색 방법을 제공하지 않아 많은 서비스로부터 자신이 원하는 서비스를 검색하는데 어려움을 겪을 수 있다.

이러한 문제점을 해결하기 위하여 OpenAPI와 웹서비스의 통합 모델을 만들고 통합 모델에 대하여 보다 효율적인 서비스 검색을 가능하게 하고, 나아가 기존의 평문(plain text) 대상의 정보 검색 기법을 서비스에 적용하고 이를 이용할 수 있는 방안을 연구함으로써 서비스 검색(service retrieval)이라는 새로운 분야에 대한 연구를 선도 수행할 수 있는 이점을 가진다.

2. 관련연구

2.1 기존의 OpenAPI 검색 기법

Programmableweb[2]은 오픈API 및 매쉬업 정보를 제공하는 웹사이트이다. 하지만 Programmableweb에서는 각각의 OpenAPI 정보를 저장하여 검색 서비스를 제공하고 있지만 단지 서비스의 이름과 설명정보 혹은 정의된 카테고리 이름에서의 검색만 제공한다. 이것은 단지 단순한 키워드를 통한 질의를 순위화된 정보를 확인할 수도 없을 뿐만 아니라, 검색 범위가 한정되었기 때문에 OpenAPI 매쉬업을 위한 서비스 검색을 제공할 수 없다.

2.2 기존의 웹서비스 검색 기법

웹서비스 검색은 크게 레지스트리 기반 검색, 시맨틱 기술 정보를 이용한 검색, 유사도 기반 검색으로 분류할 수 있다.

레지스트리 기반 검색은 웹서비스 제공자와 요청자가 웹서비스에 대한 WSDL 문서와 설명 정보를 레지스트리에 등록하고, 검색하는 방법이다. 그러나 이 UDDI는 웹서비스의 등록에 대한 단일화 된 방법을 제공한다는 이점은 있지만, 검색에 있어 여러 제약점을 가진다.

시맨틱 기술 정보를 활용한 검색 방법은 서비스의 검색과 이용을 위해 웹서비스 표준 기술에 시맨틱 설명 정보를 추가하고 이의 추론을 통한 검색과 이용의 가능성을 목적으로 하고 있다. 대표적인 연구로는 OWL-S[3], WSDL-S[4], METEOR-S[5]와 같은 연구들이 존재한다. 이들 연구들은 레지스트리 기반 검색 보다 지적이며 정확한 발견을 보장하지만 다음과 같은 문제점들을 가진다. 시맨틱 정보의 추론과 처리에 많은 부하가 걸리고, 서비스 개발 시 서비스의 시맨틱 정보를 자동 생성하기가 어렵다.

레지스트리 기반 검색의 문제점과 레지스트리 기반 검색의 해결을 위한 시맨틱 기술 정보를 이용한 검색의 방법에서의 이런 문제점으로 인하여, 최근에는 유사도 기반 검색과 같이 기존의 정보 검색 기법을 도입하여 웹서비스 검색 방법을 풀고자 하는 시도가 이루어졌다. 이러한 연구들은 웹서비스 또는 서비스 내의 연산자들에 대한 유사도를 계산함으로써 가장 유사도가 높은 서비스들을 추려 검색 결과로 제공한다. 대표적인 연구로는 Woogole[6], Peng[7], Lee[8]와 같은 연구들이 존재한다.

Lee[8]은 서비스 검색에 있어 기존 기법들과 달리 UDDI 레지스트리에 저장된 서비스 설명 정보와 WSDL 서비스 정의 모두를 활용하였으며 n-gram을 이용한 키워드 질의를 통해 부분문자열의 효율적인 검색을 보장 하였고 가중치 부여와 유사도 계산 기법을 새로 고안하여 사용자는 키워드 질의를 통해 순위화 된 서비스 검색 결과를 제공 받을 수 있다.

이 연구에서는 용어 기반 역인덱스 검색의 기본 구조로 사용하고 있으므로, 가중치도 용어에 대해서 부여된다. 또한 키워드 질의를 통한 순위화는 사용자에게 알맞은 서비스를 제공한다.

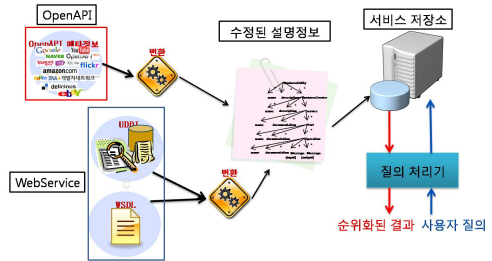
3. OpenAPI와 웹서비스의 통합 검색 시스템 개발

3.1 시스템 구현 환경

다음은 본 시스템의 구현 환경이다.

- 운영체제 : Linux Fedora 8
- 구현언어 : Java, JSP
- 데이터베이스 : Mysql

3.2 검색 시스템 구조



(그림 3) 검색 시스템의 구조

(그림 3)는 본 시스템의 전체 구조를 나타낸다. 본 시스템에서는 UDDI와 WSDL에서 필요한 정보만을 추출하여 설명정보를 만든다. 또한 OpenAPI의 정보를 직접 입력받아서 설명정보로 변환한다. 생성된 설명정보는 서비스 저장소에 저장된다. 사용자의 질의를 받으면 Query Evaluator는 서비스 저장소에 저장된 데이터를 기반으로 유사도 알고리즘을 통해 사용자에게 순위화 된 결과 값을 반환한다.

3.3 OpenAPI와 웹서비스의 통합 모델의 설명정보 정의

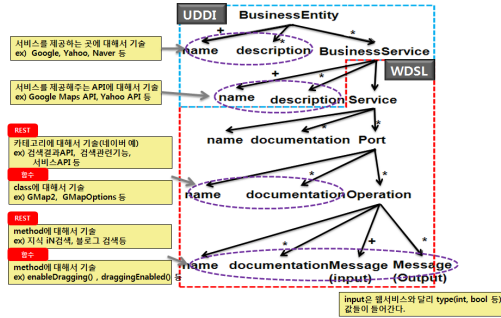
Lee[8]에서는 웹서비스 검색을 위해 UDDI와 WSDL에서 필요한 정보만 추려 독자적인 구조의 서비스 설명 정보를 구성하지만 웹서비스와 유사한 구조를 가지는 OpenAPI를 적용을 고려하지 않았다. 따라서 본 논문에서는 앞선 연구의 웹서비스 검색 구조를 활용하여 OpenAPI의 저장을 가능하게 하는 저장 구조와 이를 통해 검색을 가능하게 하는 검색 엔진을 개발하였다. 이를 위해서는 기존의 웹서비스의 설명정보를 수정하여 OpenAPI를 적용해야한다. 다음 (그림 4)은 OpenAPI와 웹서비스의 통합 모델의 설명정보이다.

3.4 순위화 기반 알고리즘

본 논문의 설명 정보는 Lee[8]가 제안하는 설명 정보와 동일하므로 Lee[8]가 제안하는 순위화 기반 알고리즘을 이용하였다. Lee[8]은 기존의 정보 검색에서 가장 일반화된 가중치

부여 방법 기법인 TF*IDF를 개선한 TF*IEF를 제안하였고, 서비스 유사도의 측정을 위해 VSM을 이용하였다. 또한 OpenAPI에서 메소드 값으로 설정된 문자열은 여러 단어들 이 합쳐져 합성 명사화되어 이용된다. 이 경우 역 인덱스 와 같은 단어-기반 인덱싱에서는 해당 문자열을 단어 단위로 분할을 시키던지 아니면, 원 문자열을 그대로 단어 로써 인식하고 일반적인 부분문자열 검색을 행하는 방법 을 수행해야 한다. 그래서 부분 문자열 매칭을 지원하기 위한 인덱스 기법인 n-그램을 이용한다.

4. 서비스 검색



(그림 4) 본 시스템의 설명정보

서비스 검색은 특정 범위 검색과 확장 범위 검색으로 나누어진다. 특정 범위 검색의 검색 데이터는 (그림 4)과 같이 입력한다. 검색 종류에서는 검색의 대상을 선택하는 것으로 웹서비스와 OpenAPI를 체크박스로 선택할 수 있다. 검색 범위는 Business, Service, Operation으로 나누어진다. 사용자가 검색어를 입력하면 해당 검색어가 포함된 검색 범위 내에서의 검색이 이루어진다.

특정 범위 검색은, 사용자들이 Business, Service, Operation을 선택하면 선택된 범위만을 검색하는 것이다. 즉, Business 검색일 경우 Business의 유사도만 계산하여 해당 검색어를 가진 Service만 검색하고, Service를 선택 할 경우 Service의 유사도만 계산하여 해당 검색어를 가진 Service만 검색하고, Operation을 선택할 경우 Operation의 유사도만 계산하여 해당 검색어를 가진 Service만 검색하게 된다.

유사 서비스 검색 - 특정 범위 검색

검색 종류: Web Service OpenAPI

검색 범위: Business Service Operation

검색 단어: 네이버

검색 결과

Business	위치	유사도	종류
31	www.naver.com	0.807	OpenAPI
36	http://maps.naver.com/api/geocode.php	0.807	OpenAPI

(그림 5) 유사 서비스 검색 - 특정 범위 검색

사용자가 조합할 서비스를 찾기 위해 해당 문서의 구조를 보려면 결과 값 중 원하는 것을 선택한 후 “문서 구조 보기”를 누르면 (그림 6)과 같은 문서의 구조가 출력되는 것을 확인할 수 있다.

문서 구조

WSDL로 저장

Structure Overview :

- Service Name: 네이버 지도 API
 - Port Name: NMap
 - Port Name: NSize
 - Port Name: NPoint
 - Port Name: NEvent
 - NEvent.addListener
 - NEvent.removeListener
 - Port Name: NMark
 - Port Name: NIcon
 - Port Name: NInfoWindow
 - Port Name: NPolyline
 - Port Name: NXmlhttp
 - Port Name: NZoomControl
 - Port Name: NSaveBtn
 - Port Name: NIndexMap

서비스 타입: 이진 서비스 다른 서비스

추가 검색어:

(그림 6) 문서 구조 보기

검색된 결과에서 선택한 문서의 구조가 트리형식으로 출력되고 사용자는 operation의 정보를 얻을 수 있다. operation의 정보는 라디오 버튼 형식으로 선택할 수 있고, 선택한 operation의 정보를 WSDL 문서의 형태로 변환할 수 있다. 조합 가능한 서비스 검색은 선택한 operation의 정보와 조합 가능한 서비스를 검색하는 것으로, 이전 서비스일 경우에는 선택한 서비스의 입력인자가 찾고자 하는 서비스의 출력인자와 일치하고 추가 검색어를 가지고 있는 서비스를 검색하며, 다음 서비스일 경우에는 선택한 서비스의 출력인자가 찾고자 하는 서비스의 입력인자와 일치하고 추가 검색어를 가지는 서비스를 검색한다. 이렇게 검색을 수행한 결과는 (그림 7)와 같이 출력된다. 마찬가지로 출력된 결과에 대해 WSDL 문서 형식으로 저장이 가능하며, 검색된 결과와 또다시 조합될 수 있는 서비스 검색도 가능하다.

조합 가능한 서비스 검색

서비스 타입: 이진 서비스 다른 서비스

추가 검색어: weather

검색 결과

Business	Service	Operation	URL	Similarity	type
34	1	1	http://d3ab.dnu.ac.kr	1.729	OpenAPI
34	1	2	http://d3ab.dnu.ac.kr	1.729	OpenAPI

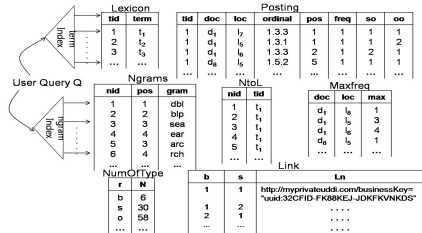
(그림 7) 조합 가능한 서비스 검색

확장 범위 검색은, 사용자들이 특정 범위 검색처럼 Business, Service, Operation을 선택할 수 있지만 선택된 범위만을 검색하는 것이 아닌 상/하위 개념 간 서비스 유사도의 전파를 고려하여 검색한다. 즉, Business 검색일 경우 하위 유사도 전파를 통해 해당 검색어를 가진 Service를 검색하고, Service를 선택할 경우, 상/하위 유사도 전파를 통해 해당 검색어를 가진 Business와 Operation을 모두 검색한다. Operation을 선택할 경우 상 위 유사도 전파를 통해 해당 검색어를 가진 Service까지 검색하게 된다. 이렇게 검색된 결과는 특정 범위 검색과 마찬가지로 “문서 구조 보기”를 통해 조합 서비스 검색에 사용된다.



(그림 8) 유사 서비스 검색 - 확장 범위 검색

5. 본 시스템의 저장구조



(그림 9) 서비스 설명정보 저장을 위한 저장 구조

(그림 9)는 본 논문에서의 서비스 설명정보 저장을 위한 DB 테이블의 구조를 보인다. 본 논문에서 서비스 저장소는 기본적으로 DB 테이블을 이용한 역 인덱스의 구현이며 SQL 질의를 통해 서비스 유사도를 계산해낸다. 역 인덱스는 Lexicon과 Posting 테이블로 구현이 되며, 공백 문자로 구별되는 각 용어들은 Lexicon 테이블에 저장되고, 각기 유일한 용어 ID, tid를 부여받는다. Posting 테이블은 해당 용어의 빈도와 물리적, 개념적 위치 정보를 저장한다. Posting 테이블에서 doc 컬럼은 문서의 ID를 나타내며, loc, ordinal은 엘리먼트의 경로 id, ordinal은 듀이 넘버링 기법의 일종인 DLN으로 표시되는 각 노드들의 트리 상에서의 순서 정보를 나타낸다. pos와 freq는 각각 한 용어가 해당 엘리먼트의 텍스트에서 출현하는 오프셋과 빈도를 기록한다. co 컬럼은 트리 상에서 각 개념들 간의 순서 정보를 DLN으로 표시한 값을 가진다. 역 인덱스로의 동작은 다음과 같은 두 테이블 간의 동등 조인을 통해 가능하다.

6. 결론

우선 대상이 되는 서비스에 대해 서비스 검색의 개념을 정립하고 시스템을 설계하였다. 서비스 검색에서 검색의 대상이 되는 문서는 평문(plain text)이 아닌 서비스 설명 정보(service description)이다. Lee[8]에서는 웹서비스 검색을 위해 UDDI와 WSDL에서 필요한 정보만 추려 독자적인 구조의 서비스 설명 정보를 구성하지만 웹서비스와 유사한 구조를 가지는 OpenAPI를 수용을 고려하지 않았다. 제안된 시스템에 OpenAPI를 적용시키기 위해 OpenAPI에 대해 살펴보았다. OpenAPI는 전송 방식에 따라 REST형과 함수형 OpenAPI로 구분할 수 있다. 각각의 방식에 따라 설명정보에 데이터를 저장시켜 검색을 가능하게 하였다.

앞에서 정의한 알고리즘을 바탕으로 시스템의 효율성 증진을 위한 저장구조를 개발하였다. 해당 시스템은 관계형 DBMS를 이용하여 구현되었으며, 웹서비스 레지스트리인 UDDI와의 연계를 수행한다. 서비스 설명정보의 저장소는 데이터베이스 테이블들과 이들에 걸쳐지는 인덱스들로 구성된다. DB 구축을 위해 먼저 UDDI의 programmer's API를 호출하여 businessEntity 정보를 가져온다. 또한, UDDI에서 링크로 연결되는 WSDL 문서 또한 가져와, 이 문서들을 파싱하여 DB에 저장하고 SQL을 통해 값을 검색한다. 또한 사용자 인터페이스에 대해 설명하였다.

향후 연구로는 지금의 조합 가능한 검색 기법은 직접 연결된 서비스를 대상으로 하는 검색 기법이다. 향후 이를 발전시켜 간접적으로 연결된 서비스를 대상으로 하는 검색 기법의 개발이 요구된다.

웹서비스와 OpenAPI 조합 검색을 통해 검색된 내용을 자동으로 통합시켜 제공해줄 도구의 개발이 필요하다. 이러한 자동화된 도구는 발전하고 있는 메쉬업(mesh-up)에 큰 도움을 줄 것이다.

참고문헌

- [1] David Booth, Hugo Haas, Francis McCabe, Eric Newcomer, Michael Champion, Chris Ferris, David Orchard, "Web services architecture", Feb 2004. <http://www.w3.org/TR/ws-arch/>
- [2] <http://www.programmableweb.com/>
- [3] Martin, D., Burnstein, M., et al., OWL-S: Semantic Markup for Web Services, <http://www.w3.org/Submission/OWL-S>, 2004
- [4] Akkiraj, R., Farrel, J., et al., Web Service Wemantics-WSDL-S, A joint UGA-IBM Technical Note, version 1, 2005
- [5] Patil, A. A., Oundhakar, S.A., et al., Meteor-S: Web Service Annotation Framework. ACM Press, NY, USA, 2004
- [6] Curbera, F., et al., Unraveling the Web services web: An introduction to SOAP, WSDL, and UDDI, IEEE Internet Computing 6(2) pp. 86-94, 2002
- [7] Peng, D., Huang S., Wang, X., Zhou, A., Concept-based retrieval of alternate Web services, Proceedings of the 10th Conference on DASFAA(Database Systems for Advanced Applications) pp. 359-371, 2005
- [8] Kyong-Ha Lee, Kyu-Chul Lee, To Maximize Web Service Retrieval, Convergence Information Technology ICCIT, pp 2318-2325, 2007