KIPS Tr. Software and Data Eng.
Vol.3, No.10 pp.397~406 pISSN: 2287-5905

http://dx.doi.org/10.3745/KTSDE.2014.3.10.397

Design and Implementation of a Large-Scale Spatial Reasoner Using MapReduce Framework

Sang Ha Nam† · In Cheol Kim††

ABSTRACT

In order to answer the questions successfully on behalf of the human in DeepQA environments such as Jeopardy! of the American quiz show, the computer is required to have the capability of fast temporal and spatial reasoning on a large-scale commonsense knowledge base. In this paper, we present a scalable spatial reasoning algorithm for deriving efficiently new directional and topological relations using the MapReduce framework, one of well-known parallel distributed computing environments. The proposed reasoning algorithm assumes as input a large-scale spatial knowledge base including CSD-9 directional relations and RCC-8 topological relations. To infer new directional and topological relations from the given spatial knowledge base, it performs the cross-consistency checks as well as the path-consistency checks on the knowledge base. To maximize the parallelism of reasoning computations according to the principle of the MapReduce framework, we design the algorithm to partition effectively the large knowledge base into smaller ones and distribute them over multiple computing nodes at the map phase. And then, at the reduce phase, the algorithm infers the new knowledge from distributed spatial knowledge bases. Through experiments performed on the sample knowledge base with the MapReduce-based implementation of our algorithm, we proved the high performance of our large-scale spatial reasoner.

Keywords : Spatial Reasoner, Directional Relations, Topological Relations, Distributed Processing, MapReduce

맵리듀스 프레임워크를 이용한 대용량 공간 추론기의 설계 및 구현

남 상 하† · 김 인 철††

요 약

미국의 Jeopardy! 퀴즈쇼와 같은 DeepQA 환경에서 인간을 대신해 컴퓨터가 효과적으로 답하기 위해서는, 광범위한 지식베이스와 빠른 시공간 추론 능력이 요구된다. 본 논문에서는 대표적인 병렬 분산 컴퓨팅 환경인 맵리듀스 프레임워크를 이용해, 새로운 방향 및 위상 관계를 효율적으로 추론할 수 있는 대용량 공간 추론 알고리즘을 제시한다. 이 추론 알고리즘은 CSD-9 방향 관계들과 RCC-8 위상 관계들을 포함한 대용량 공간 지식베이스를 입력으로 가정하며, 이로부터 새로운 방향 관계와 위상 관계들을 추론해내기 위해 지식베이스에 대한 경로 일관성 검사와 교차 일관성 검사를 수행한다. 맵리듀스 프레임워크의 원리에 따라 추론 계산의 병렬성을 극대화하기 위해, 맵 단계에서는 대용량의 지식베이스를 다수의 노드들에 효과적으로 분할하여 분산시키고, 리듀스 단계에서는 분산된 지식베이스들로부터 새로운 공간 지식을 유도하도록 공간 추론 알고리즘을 설계하였다. 본 연구에서는 맵리듀스 프레임워크로 구현한 대용량 공간 추론기와 샘플 공간 지식베이스를 이용한 실험들을 수행하고, 이를 통해 본 논문에서 제안한 대용량 공간 추론기의 높은 성능을 확인할 수 있었다.

키워드 : 공간 추론, 방향 관계, 위상 관계, 분산 처리, 맵리듀스

1. 서 론

최근에 IBM의 Watson[1] 시스템이 Jeopardy! 퀴즈쇼[2]에서 인간 경쟁자들을 이기고 우승한 사건은 자연어 처리(natural language processing), 질의응답(question answering), 지식 표현 및 추론(knowledge representation and reasoning), 증거-기반 학습(evidence-based learning) 등 거의 인공지능

※ 본 연구는 미래창조과학부 및 한국산업기술평가관리원의 SW컴퓨팅산업원천기술개발사업(SW)의 일환으로 수행하였음(10044494, WiseKB: 빅데이터 이해 기반 자가학습형 지식베이스 및 추론 기술 개발).
※ 본 논문은 제40회 춘계학술발표대회에서 "맵리듀스 프레임워크를 이용한 대용량 공간 추론 방식"의 제목으로 발표된 논문을 확장한 것임.
† 준 회 원 : 경기대학교 컴퓨터과학과 석사과정
†† 종신회원 : 경기대학교 컴퓨터과학과 교수
Manuscript Received : June 2, 2014
Accepted : September 17, 2014
* Corresponding Author : In Cheol Kim(kic@kyonggi.ac.kr)

전 분야에 걸쳐 새로운 원동력을 제공하는 기회가 되었다. 퀴즈쇼에서 주어지는 질문들에 효과적으로 답하기 위해서는 인물, 지리, 사건, 역사 등을 포함하는 광범위한 지식베이스와 빠른 시공간 추론(temporal and spatial reasoning) 능력이 필요하다. 퀴즈쇼에 등장하는 공간 질의(spatial query)들은 주로 주요 장소들 사이의 방향(direction), 포함(containment), 그리고 경계(border) 관계 등을 포함하고 있다. 이러한 공간 관계들을 다루기 위한 대표적인 공간 지식 표현과 정성적 추론 이론으로는 CSD(Cone-Shaped Directional)-9[3]와 RCC (Region Connection Calculi)-8[4] 등이 있다[5, 6]. 그리고 이 이론들에 기초해 개발된 대표적인 공간 추론기(Spatial reasoner)들로는 SOWL[7], PelletSpatial[8], CHOROS[9, 10], QUSAR[11] 등이 존재한다. SOWL[7]은 시맨틱 웹 규칙 언어인 SWRL[12]로 구현한 시공간 추론기(spatio-temporal reasoner)이다. 한편, PelletSpatial[8]은 효율성이 높은 경로 일관성(path-consistency) 알고리즘을 채용한 RCC-8 공간 추론기이며, CHOROS[9, 10]는 CSD-9 추론도 가능하도록 PelletSpatial을 확장한 공간 추론기이다. QUSAR[11]은 PelletSpatial을 확장하여 CSD-9 방향 관계 집합과 RCC-8 위상 관계 집합 각각에 대한 추론도 가능하고 이 둘 간의 상호 교차 일관성 검사 기능도 추가된 공간 추론기이다.

최근 시맨틱 웹의 발전에 따라 웹에 존재하는 정보를 반-구조적 형태로 표현하는 것이 가능해지고 이를 이용해 대용량의 지식베이스를 생성할 수 있게 되었다. 이에 따라 공간 추론에 사용되는 지식베이스도 수십, 수백억 개의 지식들로 구성됨에 따라 단일 머신으로 공간 추론을 수행하기에는 성능적 한계가 존재한다. 이에 대한 해결책으로 최근 대용량 웹 스케일 지식베이스 추론에 대한 연구가 활발히 진행중이고, 대표적인 연구로는 WebPIE[13]가 있다. WebPIE는 맵리듀스(MapReduce)[14] 분산 환경에서 RDF 및 OWL 추론을 수행하는 대용량 분산 추론기이다. 하지만 현재 이 추론기는 시공간 지식에 대한 어떤 추론 기능도 제공하지 못하고 있다.

일반적으로 분산 시스템의 장점을 잘 활용하려면, 대규모 입력 데이터를 클러스터의 각 노드에 효과적으로 나누어서 할당해야 하고, 각 노드는 자신에게 할당된 데이터를 토대로 서로 독립적으로 계산을 수행할 수 있어야 한다. 그러나 만약, 데이터들 사이에 강한 연관성과 의존성이 존재한다면, 이와 같이 각 노드가 서로 독립적으로 작업을 수행하도록 데이터를 분할하기 어렵게 되고, 결국 노드 간의 빈번한 통신 유발로 인해 분산 시스템의 성능을 저하시키게 된다. 하지만, 본 연구에서 다루고자 하는 CSD-9 및 RCC-8 기반의 대용량 공간 지식베이스는 지식들 간의 연관성도 높고, 새로운 지식을 추론하기 위한 연산들 간의 상호 의존성도 크다. 따라서 분산 시스템을 이용한 대용량 공간 추론기를 개발하기 위해서는, 먼저 효과적인 지식 분할(knowledge partitioning)을 통해 병렬화의 이점을 잘 살릴 수 있도록 알고리즘을 설계해야 한다.

본 논문에서는 대표적인 병렬 분산 컴퓨팅 환경인 맵리듀스 프레임워크[14]를 이용하여 방향 및 위상 관계를 추론하는 효율적인 대용량의 공간 추론 알고리즘을 제시한다. 이 추론 알고리즘은 CSD-9 방향 관계들과 RCC-8 위상 관계들을 포함한 대용량 공간 지식베이스를 입력으로 가정하며, 이로부터 새로운 방향 관계와 위상 관계들을 추론해내기 위해 지식베이스에 대한 경로 일관성 검사와 교차 일관성 검사를 수행한다. 맵리듀스 프레임워크의 특성을 고려하여 병렬 분산처리의 효과를 높이기 위해, 맵 단계에서 지식 분할 문제를 해결하고, 이것을 토대로 리듀스 단계에서 효과적으로 새로운 공간 지식을 유도하도록 추론 알고리즘을 설계하였다. 본 논문에서 제안하는 대용량 공간 추론 알고리즘의 성능을 분석하기 위해, 맵리듀스 프레임워크로 구현한 대용량 공간 추론기와 공간 지식 생성기로 만든 샘플 공간 지식베이스를 이용한 성능 분석 실험을 수행하고 그 결과를 소개한다.

2. 관련 연구

2.1 SOWL

SOWL[7]은 시맨틱 웹 온톨로지 언어인 RDF/OWL을 기초로 시공간 지식을 4차원 서술자(4-D fluent)와 다자 관계(N-ary relation)로 표현하고, 추론 규칙들을 시맨틱 웹 규칙 언어인 SWRL로 구현한 시공간 추론기(spatio-temporal reasoner)이다. 이 추론기에서는 시간 지식 표현과 추론을 위해서는 Allen의 이론을, 공간 지식 표현과 추론을 위해서는 CSD-9과 RCC-8 이론을 각각 적용하였다. 하지만 SOWL은 SWRL 규칙 엔진(rule engine)을 이용한 구현 방식의 한계성과 시공간 추론 규칙들 간의 최적화가 충분히 이루어지지 않아 실용적으로 이용하기 어려운 성능을 보였다.

2.2 PelletSpatial

PelletSpatial[8]은 RDF/OWL 추론기 중 하나인 Pellet을 확장한 것으로서 공간 지식을 RCC-8 형태로 표현하고, 이를 바탕으로 새로운 공간 지식을 추론하는 정성적 공간 추론기이다. 이 추론기는 2가지 방식으로 RCC-8 공간 추론을 수행할 수 있다. 첫 번째는 RCC-8 공간 관계를 RDF/OWL 관계로 변경한 후 추론하는 방식이고, 두 번째는 효율성 높은 경로 일관성 알고리즘을 채용하여 RCC-8 조합표를 기반

으로 추론하는 방식이다. 실험 결과 두 번째 방식이 첫 번째 방식보다 성능 면에서 훨씬 나은 모습을 보였다. 게다가 이 추론기에는 하나 이상의 RCC-8 공간 관계를 포함하는 SPARQL 형태의 공간 질의에 대한 답을 찾아내는 요소도 포함되어 있다.

2.3 CHOROS

CHOROS[9, 10]는 PelletSpatial에 CSD-9 추론도 가능하도록 확장한 공간 추론기로서 공간 지식을 RCC-8과 CSD-9 형태로 표현하고, PelletSpatial에서 채용한 경로 일관성 알고리즘으로 공간 추론을 수행한다. 공간 추론에 사용되는 RCC-8 조합표와 CSD-9 조합표는 SOWL에서 정의한 조합표들을 사용하였고, 멀티스레딩 기법을 이용하여 RCC-8과 CSD-9 추론을 병렬로 수행하였다. 그리고 PelletSpatial과 마찬가지로 하나 이상의 공간 관계를 포함하는 SPARQL 형태의 공간 질의 처리 기능을 포함하고 있다. 하지만, 이 추론기는 두 공간 사이의 방향 관계를 다루는 CSD-9와 위상 관계를 다루는 RCC-8 지식들 간의 상호 교차 일관성을 검사하는 추론 요소는 포함하고 있지 못하다는 한계성을 가진다.

2.4 QUSAR

QUSAR[11]는 PelletSpatial에 CSD-9 추론과 교차 추론도 가능하도록 확장한 공간 추론기이다. 이 추론기는 RCC-8 위상 관계 집합과 CSD-9 방향 관계 집합 각각에 대한 추론을 할 뿐만 아니라, 이 둘 간의 상호 교차 일관성 검사 기능도 포함하고 있다. 상호 교차 일관성 검사 기능이란, 위상 관계 집합과 방향 관계 집합을 통합적으로 추론하는 것으로서 CHOROS 방법보다 더 많은 공간 지식을 생성하고 강도 높은 공간 지식베이스의 불일치성 발견 능력을 가지고 있다. 이 추론기도 역시 질의 처리 기능을 포함하고 있다.

2.5 WebPIE

WebPIE[13]는 맵리듀스 프레임워크를 이용한 대용량의 웹 스케일 지식베이스에 대한 RDF/OWL 추론기이다. 이 추론기는 RDF 추론과 OWL 추론을 각각 수행할 수 있다. RDF 추론은 총 14개의 RDFS 추론 규칙 분석을 통해 효율적인 추론 순서를 정의하였으며, 매우 빠른 추론 속도를 보였다. OWL 추론은 총 23개의 OWL 추론 규칙을 분석하여 추론 순서를 정했으나, OWL 추론의 특성상 반복적인 추론 작업 실행이 불가피해 느린 추론 속도를 보였다. [11]에서는 입력 데이터에 대한 인코딩과 디코딩 기법을 적용하여 추론 속도를 향상시켰다. 그러나 이 추론기는 시공간 지식에 대한 추론을 하지 못하는 한계성을 지니고 있다.

3. 정성적 공간 추론

3.1 공간 지식 표현

공간 추론기를 설계하기 위해서는 먼저 추론에 필요한 공간 지식을 어떻게 표현할 것인지, 즉 공간 지식 표현법(spatial knowledge representation)을 정할 필요가 있다.

본 연구에서는 추론에 이용되는 공간 지식베이스는 시맨틱 웹 표준 온톨로지 언어인 RDF/OWL에 따라 모두 (subject predicate object) 형태의 트리플 문장(triple statement)들로 표현되며, 지식베이스에 등장하는 각 장소는 GeoInstance 클래스에 속하는 한 원 소로 정의된다고 가정한다. 공간 지식베이스를 구성하는 각 문장(statement) 혹은 사실(fact)들은 Fig. 1과 같다. 두 공간 혹은 두 장소(GeoInstance) 사이의 방향, 경계, 위상 관계 등을 CSD-9와 RCC-8에서 정의한 공간 서술자(spatial property)들을 이용하여 표현하는 형태이다. 예컨대, “캐나다는 미국의 북쪽에 위치하고 있다.”라는 두 공간 간의 방향 관계는 (Canada northOf USA)와 같이 표현할 수 있다.

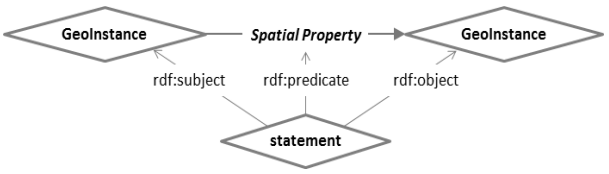

Fig. 1. Triple statement representation

CSD-9와 RCC-8 공간 추론은 모두 공간과 공간 사이의 관계를 표현하는 지식 형태를 가정한다. CSD(Cone-Shaped Directional)-9 이론에서는 2차원 공간 위의 임의의 두 지점(point) 간의 방향 관계는 Fig. 2와 같이 한 지점을 중심으로 판단할 때 다른 한 지점의 방향을 동(E), 서(W), 남(S), 북(N), 북동(NE), 북서(NW), 남동(SE), 남서(SW), 그리고 동향(Identical) 등 총 9개 방향 중 하나로 표현할 수 있다고 가정한다. 반면에, RCC(Region Connection Calculi)-8 이론에서는 2차원 공간 위의 임의의 두 지역(region) 간의 위상 관계를 Fig. 3과 같이 DC(disconnect), EC(externally connected), PO(partially overlapping), EQ(equal), TPP(tangential proper part), TPPi(tangential proper part inverse), NTPP(non-tangential proper part), NTPPi(non-tangential proper part inverse) 등 총 8개 관계 중 하나로 표현할 수 있다고 가정한다. 따라서 CSD-9 공간 지식은 두 공간의 방향 관계를

점(point)의 관점에서 기술하는 데 반해, RCC-8 공간 지식은 두 공간의 위상 관계를 영역(region)의 관점에서 기술한다고 볼 수 있다. 많은 실세계 공간 혹은 장소는 때로는 하나의 점으로, 때로는 하나의 영역으로 해석해야 할 필요가 있는 다면성을 가지기 때문에, CSD-9 공간 지식과 RCC-8 공간 지식은 실세계 공간들 간의 다양한 관계를 표현하고 추론하는 데 상호 보완적으로 이용될 수 있다.

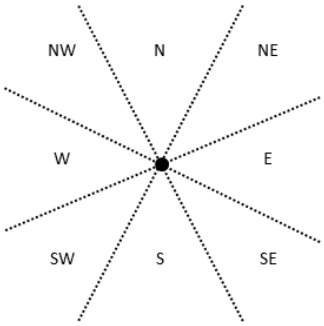

Fig. 2. Directional relations in CSD-9

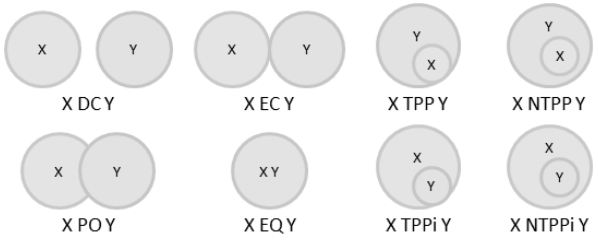

Fig. 3. Topological relations in RCC-8

3.2 공간 추론 규칙

9개의 방향 관계들로 표현되는 공간 지식베이스에 적용되는 CSD-9 공간 추론 규칙들(inference rules)은 Table 1과 같은 하나의 조합표(composition table)로 요약할 수 있다.

즉, Table 1은 가로 행의 사실과 세로 열의 사실이 동시에 참이라면, 해당 행과 열이 교차하는 난에 열거된 새로운 사실들을 조합해낼 수 있음을 암시한다. 예를 들어, 표에 음영으로 표시된 부분과 같이, 장소 A가 장소 B의 북쪽에 위치[N(A, B)]하고 B는 장소 C의 북동쪽에 위치[NE(B, C)]하고 있을 때, A는 C의 북쪽이나 북동쪽에 위치[[N, NE](A, C)]할 수 있다는 새로운 사실들을 추론할 수 있다. 그리고 N(A, B)와 NE(B, C)와 같이 두 공간 사이의 방향 관계가 명확히 하나로 정의된 경우, 이러한 사실들을 정의 관계들(defined relations)이라 부른다. 이에 반해, [N, NE](A, C)와 같이 두 공간 사이의 방향 관계를 하나로 명확히 정할 수 없을 때는 이들을 이접 관계들(disjunctive relations)이라고 부른다. 이와 유사한 방법으로, RCC 공간 추론 규칙들은 Table 2와 같은 조합표로 요약할 수 있다[4]. 예를 들어, 표에 음영으로 표시된 부분과 같이, A가 B의 경계에 접해있고[EC(A, B)] B가 C를 접점 없이 완전히 내포하고 있을 때 [NTPPi(B, C)], A는 C와 서로 떨어져 있다[DC(A, C)]는 새로운 사실을 추론해낼 수 있다. 이러한 의미에서 기존의 공간 지식베이스로부터 Table 1과 Table 2의 조합표에 따라 새로운 사실들을 유도하는 추론과정을 이행적 조합(transitive composition)이라고도 부른다.

Table 1. Composition table for inferring new CSD-9 relations

	N	NE	E	SE	S	SW	W	NW	O
N	N	N, NE	N, NE, E	N, NE, E, SE	*	N, SW, W, NW	N, W, NW	N, NW	N, O
NE	N, NE	NE	NE, E	NE, E, SE	NE, E, SE, S	*	N, NE, W, NW	N, NE, NW	NE, O
E	N, NE, E	NE, E	E	E, SE	E, SE, S	E, SE, S, SW	*	N, NE, E, NW	E, O
SE	N, NE, E, SE	NE, E, SE	E, SE	SE	SE, S	SE, S, SW	SE, S, SW, W	*	SE, O
S	*	NE, E, SE, S	E, SE, S	SE, S	S	S, SW	S, SW, W	S, SW, W, NW	S, O
SW	N, SW, W, NW	*	E, SE, S, SW	SE, S, SW	S, SW	SW	SW, W	SW, W, NW	SW, O
W	N, W, NW	N, NE, W, NW	*	SE, S, SW, W	S, SW, W	SW, W	W	W, NW	W, O
NW	N, NW	N, NE, NW	N, NE, E, NW	*	S, SW, W, NW	SW, W, NW	W, NW	NW	NW, O
O	N, O	NE, O	E, O	SE, O	S, O	SW, O	W, O	NW, O	*

Table 2. Composition table for inferring new RCC-8 relations

	DC	EC	PO	TPP	NTPP	TPPi	NTPPi	EQ
DC	*	DC, EC, PO, TPP, NTPP	DC, EC, PO, TPP, NTPP	DC, EC, PO, TPP, NTPP	DC, EC, PO, TPP, NTPP	DC	DC	DC
EC	DC, EC, PO, TPPi, NTPPi	DC, EC, PO, TPP, TPPi, EQ	DC, EC, PO, TPP, NTPP	EC, PO, TPP, NTPP	PO, TPP, NTPP	DC, EC	DC	EC
PO	DC, EC, PO, TPPi, NTPPi	DC, EC, PO, TPPi, NTPPi	*	PO, TPP, NTPP	PO, TPP, NTPP	DC, EC, PO, TPPi, NTPPi	DC, EC, PO, TPPi, NTPPi	PO
TPP	DC	DC, EC	DC, EC, PO, TPP, NTPP	TPP, NTPP	NTPP	DC, EC, PO, TPP, TPPi, EQ	DC, EC, PO, TPPi, NTPPi	TPP
NTPP	DC	DC	DC, EC, PO, TPP, NTPP	NTPP	NTPP	DC, EC, PO, TPP, NTPP	*	NTPP
TPPi	DC, EC, PO, TPPi, NTPPi	EC, PO, TPPi, NTPPi	PO, TPPi, NTPPi	PO, TPP, TPPi, EQ	PO, TPP, NTPP	TPPi, NTPPi	NTPPi	TPPi
NTPPi	DC, EC, PO, TPPi, NTPPi	PO, TPPi, NTPPi	PO, TPPi, NTPPi	PO, TPPi, NTPPi	PO, TPP, NTPP, TPPi, NTPPi, EQ	NTPPi	NTPPi	NTPPi
EQ	DC	EC	PO	TPP	NTPP	TPPi	NTPPi	EQ

Table 3. Conversion table for inferring new RCC-8 relations from CSD-9 ones and vice versa

CSD-9	RCC-8
O	EQ, PO, TPPi, NTPPi, TPP, NTPP
N, NE, E, SE, S, SW, W, NW	DC, EC, PO

본래 CSD-9와 RCC-8은 각자 서로 다른 관점에서 공간 지식 표현과 추론 방법을 다루는 독립적인 이론들이다. 그러나 앞서 설명한 바와 같이 실세계의 많은 공간과 장소들은 CSD-9와 같은 방향 관계와 RCC-8과 같은 위상 관계를 함께 표현하고 추론해야 하는 경우가 많다. 이러한 경우, 이들을 통합적으로 추론해야 하는 공간 추론 알고리즘은 CSD-9의 방향 관계를 나타내는 사실들로부터 RCC-8의 위상 관계 관점에서는 어떤 새로운 사실들을 유추해낼 수 있는지, 혹은 그 반대 방향으로는 어떤 추론이 가능한지를 알아야 한다. 본 논문에서는 다양한 사례의 공간 지식베이스 분석을 통해, Table 3과 같은 CSD-9와 RCC-8 관계들 사이의 변환 규칙들을 발견하였다. Table 3에서 CSD-9의 O(Identical)관계를 나타내는 하나의 사실은 RCC-8의 {EQ, PO, TPPi, NTPPi, TPP, NTPP} 관계들 중 하나를 만족할 수 있다는 사실을 암시하며, CSD-9의 {N, NE, E, SE, S, SW, W, NW} 등의 관계를 나타내는 사실은 RCC-8의 {DC, EC, PO} 관계들 중 하나를 만족할 수 있다는 사실을 뜻한다. 또한 그 반대 방향으로 RCC-8의 {EQ, PO, TPPi, NTPPi, TPP, NTPP} 등의 관계는 CSD-9의 O 관계를 암시하며, RCC-8의 {DC, EC, PO} 등의 관계는 CSD-9의 {N, NE, E, SE, S, SW, W, NW} 등의 관계를 만족할 수 있다. 따라서 각 경우에 해당하는 정의 관계(defined relation) 또는 이접 관계들(disjunctive relations)을 지식베이스에서 발견하면, 이들이 암시하는 새로운 관계들로 변환할 수 있다. 예컨대, 미국 캘리포니아 주가 LA 도시를 완전히 내포하고 있을 때(California NTPPi LA), 캘리포니아 주는 LA의 8개 방향 중 어느 방향에도 위치한다고 말할 수 없고, 따라서 캘리포니아 주는 LA와 동향(California O LA) 관계라고 해석할 수 있다. 만약 두 지역이 서로 부분적으로 겹쳐 있는 PO 관계를 만족하는 경우에는, 두 지역의 각 중심부가 서로 어떻게 위치하느냐에 따라서 동향(O) 혹은 나머지 8개 방향 중 하나의 관계를 갖는 것으로 해석할 수 있다.

4. 대용량 공간 추론기의 설계

본 논문에서는 앞서 3절에서 언급한 공간 추론 규칙들을 토대로 맵리듀스 프레임워크 기반의 대용량 공간 추론 알고리즘을 설계하였다. 전체 작업의 구성과 흐름은 Fig. 4와 같다. 공간 지식들은 서로 간의 연관성이 강하고, 특히 추론을 위해 조인 작업을 해야 할 경우가 많다. 따라서 WebPIE에서 밝힌 바와 같이, 입력 데이터를 각 노드에 알맞게 분배해야 알고리즘의 성능도 높이고 부하 분산(load balancing) 문제도 해결할 수 있다. 본 알고리즘의 맵(map) 단계들은 입력 데이터를 각 추론 작업에 알맞도록 각 노드에 분할하는 역할을 한다. 그리고 리듀스(reduce) 단계에서는 분할되어 들어온 데이터들에 대해 적절한 방법으로 새로운 공간 지식을 추론하는 역할을 한다.

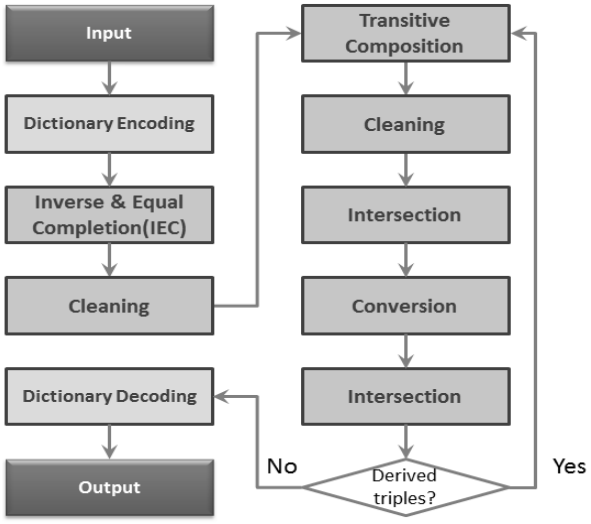

Fig. 4. Job flow of the MapReduce-based spatial reasoner

4.1 인코딩과 디코딩

Fig. 4에서 보는 바와 같이, 먼저 입력 데이터(Input)에 대한 인코딩 작업(Dictionary Encoding)이 진행된다. 이 작업은 문자열로 표현된 공간 지식을 임의의 숫자로 변환하여 공간 지식의 각 요소별로 하나의 숫자에 대응되는 사전을 만드는 작업이다. 이는 [11]에서 밝힌 바와 같이, 인코딩 작업 이후의 추론 단계뿐만 아니라 데이터 전송에 있어서도 효율성을 높일 수 있는 방법이다. 인코딩 작업의 예제는 Fig. 5와 같다.

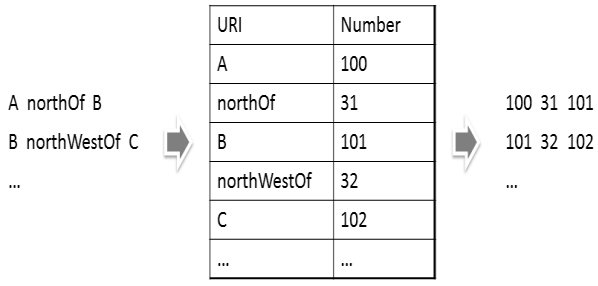

Fig. 5. Example of dictionary encoding

Fig. 5의 왼쪽은 공간 지식을 N-Triple 형태로 나타낸 것이다. 이처럼 공간 지식 내에는 장소(A, B, C)와 공간 서술자(northOf, northWestOf)가 존재한다. 이때 공간 서술자

는 총 17가지로 한정되기 때문에 미리 정의된 숫자로 할당되고, 장소는 임의의 숫자를 할당한다. 인코딩된 공간 지식은 그림의 오른쪽과 같이 트리플(triple) 형태로 표현된다. 입력 데이터에 대한 인코딩 작업을 성공적으로 마치게 되면 5가지의 추론 작업을 순차 또는 반복적으로 수행한다. 마지막으로, 유도된 공간 지식을 출력 데이터(Output)로 내보내기 전 임의의 숫자로 표현된 공간 지식을 문자열로 변환하기 위해 디코딩 작업(Dictionary Decoding)을 수행한다.

4.2 역 관계 및 동일 관계 추론

입력 데이터에 대한 인코딩 작업이 완료된 후, 역 관계 및 동일 관계 추론(Inverse & Equal completion) 작업을 가장 먼저 수행한다. 이 작업은 입력으로 들어온 모든 공간 지식들에 대해 역(inverse) 관계와 동일(equal) 관계들을 생성한다. 예를 들어 (A northOf B)라는 하나의 공간 지식에 대해 역 관계인 (B southOf A)와 동일 관계들인 (A O A), (B O B)를 각각 생성한다.

```
map(key, value) :
 // key : irrelevant
 // value : triple
 if (value.subj != value.obj) then
    inversePred = inverse_table.get(value.pred)
    write(triple(value.obj, inversePred, value.subj,
    CSD/RCCflag), null)
 else
    if (value.pred != EQ || value.pred != O) then
       exit          // not consistency

reduce(key, iterator values) :
 write(null, triple(key.subj, O, key.subj, CSDflag))
 write(null, triple(key.subj, EQ, key.subj, RCCflag))
 write(null, triple(key.obj, O, key.obj, CSDflag))
 write(null, triple(key.obj, EQ, key.obj, RCCflag))
 write(null, key)
```

Fig. 6. Inverse & Equal completion

역 관계 및 동일 관계 추론(IEC) 작업의 의사 코드(pseudo code)는 Fig. 6과 같다. 맵 단계에서는 공간 지식의 주어와 목적어가 같지 않을 때, 역 관계를 생성한다. 반대로, 주어와 목적어가 같을 때 둘 사이의 공간 서술자가 EQ 혹은 O가 아닌 것은 공간 지식베이스의 불일치성을 발견한 경우이다. 이때는 더 이상의 추론을 하지 않고 모든 작업을 종료한다. 리듀스 단계에서는 주어와 주어, 목적어와 목적어 간의 동일 관계를 생성한다. 여러 노드에서 역 관계 및 동일 관계 추론 작업이 수행되기 때문에, 이 작업의 결과로 중복된 공간 지식들이 생성될 수 있다.

따라서 이 작업 직후 중복된 지식을 제거(Cleaning)함으로써 다음 작업인 이행적 조합 추론(Transitive Composition) 작업의 부하를 줄일 수 있도록 설계하였다.

4.3 이행적 조합 추론

앞선 작업이 완료된 후, 이어서 이행적 조합 추론 작업을 수행한다. 이 작업은 입력으로 들어온 공간 지식들의 조합 관계를 생성한다. 예를 들어, (A northOf B)와 (B northWestOf C)라는 두 공간 지식을 이용해서 (A [northOf | northWestOf] C)라는 새로운 지식을 유도한다.

이행적 조합 추론 작업의 의사 코드는 Fig. 7과 같다. 맵 단계에서는 공통 인자(match point)로 사용될 URI, 즉 주어와 목적어를 각각 출력의 키(key) 값으로 구성한다. 밸류(value) 값에는 키 값이 주어인지 목적어인지를 구분하는 인자와 입력으로 들어온 공간 지식으로 구성한다. 이러한 기법은 공통 인자를 기준으로 지식 분할이 이루어지기 때문에 일종의 조인 작업인 이행적 조합 추론을 수행하기에 적합한 방법이다. 리듀스 단계에서는 입력으로 들어온 밸류 값을 두 개의 서로 다른 집합으로 나눈다. 그다음 Table 1과 Table 2에서 정의한 조합 추론 규칙을 이용하여, 두 집합에서 공통 인자를 갖는 공간 지식들 간의 조합 추론을 수행한다. 그리고 새롭게 생성된 지식의 역 관계도 함께 유도한다. 역 관계 및 동일 관계 추론 작업과 마찬가지로, 이 작업의 결과로 중복된 공간 지식들이 생성될 수 있다. 따라서 이들을 제거(Cleaning)하여 다음 작업의 효율성을 높일 수 있도록 설계하였다.

```
map(key, value) :
 // key : irrelevant
 // value : triple
 write(value.CSD/RCCflag + value.subj, '0' + value)
 write(value.CSD/RCCflag + value.obj, '1' + value)

reduce(key, iterator values) :
 for triple in values
    if (value[0] == 0) then
       join_right.add(triple)
    else
       join_left.add(triple)

 for left in join_left
    for right in join_right
       if (left.obj == right.subj) then
          composedPred = composition_table.get(left.pred,
                                right.pred)
          inversePred = inverse_table.get(composedPred)
          write(null, triple(left.subj, composedPred, right.obj)
          write(null, triple(right.obj, inversePred, left.subj)
```

Fig. 7. Transitive composition

4.4 교차 추론

다음으로 교차 추론(Intersection) 작업을 수행한다. 앞선 이행적 조합 추론 작업의 결과로 주어와 목적어 그리고 공간 관점은 같지만, 공간 서술자가 서로 다른 다양한 공간 지식이 생성될 수 있다. 따라서 이 작업은 위와 같은 조건의 공간 지식들의 교집합을 구하는 작업으로서, 확실성 높은 정제된 공간 지식을 유도한다. 예를 들어, (A [northOf | northWestOf] B)와 (A [northOf | northEastOf] B)라는 두 공간 지식이 존재할 때 (A northOf B)라는 확실성 높은 정제된 공간 지식을 유도한다.

교차 추론 작업의 의사 코드는 Fig. 8과 같다. 맵 단계에서는 공간 지식의 주어와 목적어 그리고 공간 관점을 나타내는 인자를 출력의 키 값으로 구성하고 공간 지식을 밸류 값으로 구성한다. 이러한 기법은 교차 추론 가능한 공간 지식들을 기준으로 지식 분할이 이루어지기 때문에 리듀스 단계에서 정제된 공간 지식을 쉽게 유도해낼 뿐만 아니라, 중복된 공간 지식을 생성하지 않는 장점이 있다. 리듀스 단계에서는 함께 들어온 공간 지식들의 교집합, 즉 정제된 공간 지식을 유도하고 그 결과를 출력한다. 그러나 정제된 공간 지식이 존재하지 않는 경우, 즉 교집합이 공집합인 경우에는 공간 지식 간에 불일치성을 발견한 경우이므로 더 이상 추론을 하지 않고 모든 작업을 종료한다. 예를 들어, (A northOf B)와 (A southOf B)라는 지식이 존재하는 경우는 공간 지식들 간의 불일치성이 존재하는 경우이다.

```
map(key, value) :
 // key : irrelevant
 // value : triple
  write(value.subj + value.obj + value.CSD/RCCFlag, value)

reduce(key, iterator values) :
 for  value  in  values
    refinedTriple = triple(value.subject, pred.intersect
                    (refinedTriple.pred, value.pred), value.object)

 if(refinedTriple.pred == null) then
    exit        // not consistency

 if(isIntersected)    then
    write(null, refinedTriple)
 else
    write(null, value)
```

Fig. 8. Intersection

4.5 변환 추론

다음으로 변환 추론(Conversion) 작업을 수행한다. 이 작업은 두 관점의 공간 지식을 통합적으로 추론하기 위함이다. 예를 들어, (A northOf B)라는 공간 지식에 대해 위상 관점의 새로운 공간 지식인 (A [DC | EC | PO] B)를 생성한다.

변환 추론 작업의 의사 코드는 Fig. 9와 같다. 맵 단계에서는 Table 3과 같은 변환 규칙을 이용하여 서로 다른 관점의 공간 지식을 유도해낸다. 이 작업은 다른 작업들과는 다르게 지식 분할이 불필요한 작업이므로, 맵 단계에서 추론을 수행한다. 그리고 맵의 출력 중, 키 값을 변환된 공간 지식으로 구성함으로써 리듀스 단계에서 중복 제거 효과가 나타나도록 설계하였다. 따라서 이 작업 이후에 중복 제거 작업을 따로 수행하지 않아도 되기 때문에, 알고리즘의 효율성을 높일 수 있다. 이 작업이 완료되면 변환 추론을 통해 얻어진 새로운 공간 지식과 기존의 공간 지식 간의 교집합을 통해 확실성 높은 공간 지식을 유도해야 하기 때문에, 교차 추론 작업을 다시 수행한다. 그다음 새롭게 유도된 지식이 있는지 검사한다. 새롭게 유도된 지식이 있다면 이행적 조합 추론 작업으로 돌아가 다시 5개의 작업을 순차적으로 수행하는 것을 반복한다. 반대로 새롭게 유도된 지식이 없다면 디코딩 작업을 수행하고 추론 작업을 마친다.

```
map(key, value) :
 // key : irrelevant
 // value : triple
 conversionPred = conversion_table.get(value.pred)
 write(triple(value.subj, conversionPred, value.obj), null)

reduce(key, iterator values) :
 write(null, key)
```

Fig. 9. Conversion

5. 구현 및 실험

5.1 공간 지식베이스 구축

본 논문에서 제안한 대용량 공간 추론 방식의 성능 분석 실험을 위하여, 공간 지식 생성기(spatial knowledge generator)를 이용해 대용량 공간 지식베이스를 구축하였다.

공간 지식 생성기를 그래프 이론으로 해석하면 완전 그래프에서 하나의 신장트리를 만드는 것과 비슷하고, 그 원리는 Fig. 10과 같다. 먼저 GeoInstance 개체 수(n)를 결정하고, 각 GeoInstance 개체를 하나의 노드라고 가정한다. 그다음 임의의 한 노드를 출발점(1st)으로 선정하고 해당 노드로부터 임의의 한 노드(2nd)에 간선을 연결한다. 이때 출발 노드는 트리플 문장의 주어, 간선은 서술어, 나머지 노드는 목적어가 된다. 그리고 서술어는 Fig. 2와 Fig. 3에서 가정한 공간 관계 중 하나로 선정한다. 그다음 2nd 노드로부터 임

의의 한 노드(3rd)에 간선을 연결한다. 단 이미 간선으로 연결되어 있는 노드는 임의의 노드 선정 대상에서 제외한다. 이와 같은 작업을 모든 노드에 간선이 연결될 때까지 반복한다.

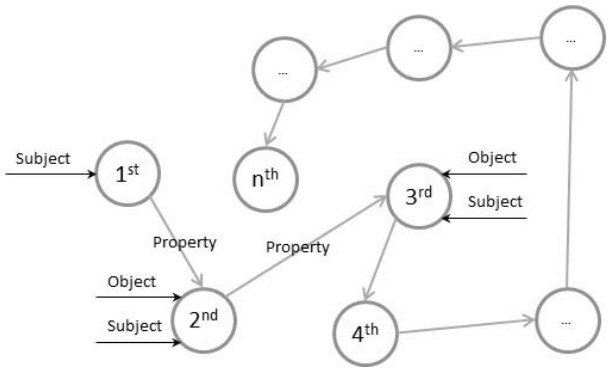

Fig. 10. Principle of the spatial knowledge generator

공간 추론의 특성상 그래프에 사이클이 존재하는 경우 해당 사이클이 어떤 공간 서술자로 구성되어 있느냐에 따라 불일치성을 가진 공간 지식이 될 수 있다. 예를 들어, “캐나다가 미국의 북쪽에 위치(Canada northOf USA)하고 미국이 멕시코의 북쪽에 위치(USA northOf Mexico)하고 있을 때, 멕시코가 캐나다의 동쪽에 위치(Mexico eastOf Canada)한다.”라는 지식은 그래프에 사이클이 형성되어 불일치성을 일으키는 경우이다. 따라서 공간 지식 생성기는 그래프에 사이클이 생기지 않는 방법 중 하나인 신장트리를 만드는 방법을 채택했다.

5.2 성능 실험

본 논문에서 제안한 대용량의 공간 추론 알고리즘의 성능 분석 실험을 위해, 맵리듀스 프레임워크를 이용한 대용량의 공간 추론기 MR_QUSAR을 구현하였다. 구현 환경은 자바 1.6 버전과 하둡 2.2 버전을 사용하였고, 실험 환경은 8개의 테스크 노드로 구성된 하둡 완전 분산 모드 클러스터를 사용하였다. 각 슬레이브 노드는 8 Core CPU와 8GB 메인 메모리, 2TB 하드 디스크로 구성되어 있다.

첫 번째 실험에서는 추론의 결과로 얻어지는 새로운 지식의 양을 평가하기 위해 새로 생성된 트리플(triple)의 개수를 측정하였으며, 실험 결과는 Fig. 11과 같다. 그림을 통해, 우리는 약 600만 개의 트리플로 구성된 대용량의 공간 지식베이스에 대해 약 7700만 개가 넘는 새로운 지식들을 유도하였다는 것을 알 수 있으며, 이를 통해 MR_QUSAR이 새로운 지식을 유도하는 추론 능력이 매우 우수하다는 것을 알 수 있다.

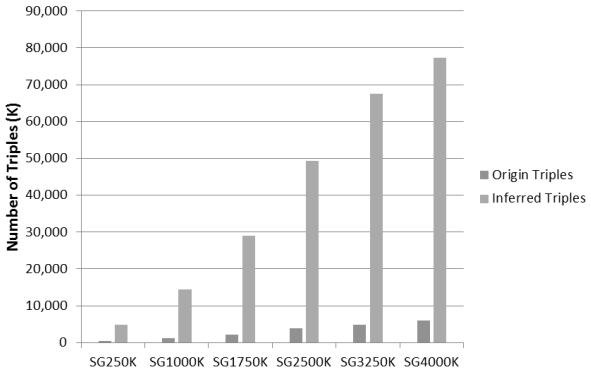

Fig. 11. Number of inferred facts

두 번째 실험에서는 공간 추론기의 추론 시간(reasoning time)과 생산성(productivity)을 평가하였으며, 실험 결과는 Fig. 12와 같다. 추론 시간은 인코딩 작업 이후부터 디코딩 작업 이전까지의 5가지 공간 추론 모듈들의 총 수행 시간을 측정하였고, 단위 시간당 추론 지식의 양을 의미하는 생산성은 추론된 지식의 양을 총 추론 시간으로 나누어 계산하였다. 그림을 통해 우리는 공간 지식베이스의 규모가 증가함에 따라, MR_QUSAR의 추론 시간이 선형적으로 증가함을 확인할 수 있다. 이러한 결과는 MR_QUSAR이 데이터 확장성 면에서 뛰어남을 의미한다. 또한 우리는 약 600만 개의 트리플로 구성된 공간 지식베이스에 대해 초당 12000개 이상의 트리플을 생산하는 것을 확인할 수 있다. WebPIE에서 23개의 규칙을 이용한 OWL Horst 추론의 생산성이 초당 약 470개인 것과 비교해보면, 149개의 규칙을 적용한 MR_QUSAR의 생산성이 매우 높다는 것을 알 수 있다.

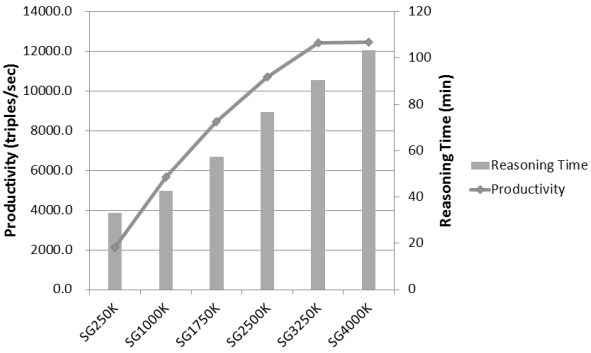

Fig. 12. Reasoning time and productivity

세 번째 실험에서는 공간 추론기의 안정성(stability)을 평가하였으며, 실험 결과는 Fig. 13과 같다. 본 논문에서는 동일한 실험을 반복했을 때 나타나는 추론 시간의 편차로 시

스템의 안정성을 판별할 수 있다고 가정한다. 따라서 각 입력 데이터 집합별로 5번씩 동일한 실험을 수행하여 추론 응답 시간을 측정해보았다. 그림을 통해 우리는 동일한 실험에 대해 MR_QUSAR의 추론 응답 시간의 상대 표준 편차(relative standard deviation) 값이 모두 2% 이하인 것을 알 수 있으며, 이를 통해 MR_QUSAR의 높은 안정성을 확인할 수 있다.

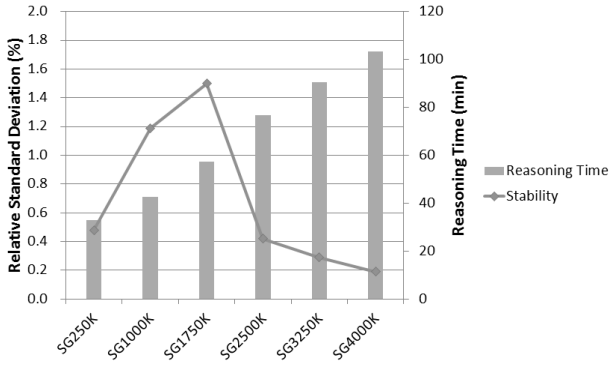

Fig. 13. Reasoning time and stability

6. 결 론

본 논문에서는 맵리듀스 프레임워크를 이용하여 방향 및 위상 관계를 추론하는 효율적인 대용량의 공간 추론 알고리즘을 소개하였다. 이 추론 알고리즘은 CSD-9 방향 관계들과 RCC-8 위상 관계들을 포함한 대용량 공간 지식베이스를 입력으로 가정하며, 이로부터 새로운 방향 관계와 위상 관계들을 추론해내기 위해 지식베이스에 대한 경로 일관성 검사와 교차 일관성 검사를 수행한다. 본 알고리즘은 맵리듀스 프레임워크의 특성을 고려하여 병렬 분산처리의 효과를 높이기 위해, 지식 분할 문제를 맵 단계에서 해결하고, 이것을 토대로 리듀스 단계에서 효과적으로 새로운 공간 지식을 유도하도록 설계하였다. 본 논문에서 제안한 대용량 공간 추론 알고리즘의 성능을 분석하기 위해, 맵리듀스 프레임워크로 구현한 대용량 공간 추론기와 공간 지식 생성기로 만든 샘플 공간 지식베이스를 이용한 성능 분석 실험을 수행하였고 이를 통해 대용량 공간 추론기의 높은 성능을 확인할 수 있었다. 향후 지속적인 작업 최적화를 통해 공간 추론기의 성능을 향상시키고, 새로운 공간 추론 기능들을 추가함으로써 공간 추론기를 확장해나갈 계획이다.

References

[1] D. A. Ferrucci, “This is Watson”, IBM Journal of Research and Development, Vol.56, No.3/4, IBM, 2012.

[2] http://www.jeopardy.com/

[3] D.J. Peuquet, C. X. Zhang, “An Algorithm to Determine the Directional Relationship between Arbitrarily-Shaped Polygons in the Plane”, Pattern Recognition Vol.20, No.1, pp.65-74, 1987.

[4] J. Renz, “Maximal Tractable Fragments of the Region Connection Calculus: A Complete Analysis”, Proceedings of IJCAI, 1999.

[5] A. G. Cohn, S. M. Hazarika, “Qualitative Spatial Representation and Reasoning: An Overview”, Fundam. Inform., Vol.46, No.1, pp.1-29, 2001.

[6] J. Renz, B. Nebel, “Qualitative Spatial Reasoning Using Constraint Calculi”, Handbook of Spatial Logics, pp.161-215, Springer, 2007.

[7] S. Batsakis, E.G.M. Petrakis, “SOWL: A Framework for Handling Spatio-Temporal Information in OWL 2.0”, Proceedings of Int. Symp. on RuleML, pp.242-249, 2011.

[8] M. Stocker, E. Sirin, “PelletSpatial: A Hybrid RCC-8 and RDF/OWL Reasoning and Query Engine”, OWLED, 2009.

[9] G. Christodoulou, “CHOROS: A Reasoning and Query Engine for Qualitative Spatial Information”, Dissertation Thesis, Technical University of Crete, Greece, 2012.

[10] G. Christodoulou, E.G.M. Petrakis, and S. Batsakis, “Qualitative Spatial Reasoning Using Topological and Directional Information in OWL”, Proceedings of the 24th Int. Conf. on Tools with Artificial Intelligence(ICTAI), Vol.1, pp.596-602, 2012.

[11] S. Nam, I. Kim, “Design and Implementation of a Qualitative Spatial Reasoner Based on CSD-9 and RCC-8 Theories”, Proc. of KIISE Fall Conference, pp.652-654, 2013.

[12] I. Horrocks, P. F. Patel-Schneider, H. Boleyet, et al., “SWRL: A Semantic Web Rule Language Combining OWL and RuleML”, W3C Member submission, 2004.

[13] J. Urbani, S. Kotoulas, J. Maassen, et al., “WebPIE: A Web-scale Parallel Inference Engine using MapReduce”, Web Semantics: Science, Services and Agents on the World Wide Web, Vol.10, pp.59-75, 2012.

[14] S. Perera, T. Gunarathne, “Hadoop MapReduce Cookbook”, Packt Publishing, 2013.

남 상 하

e-mail : namsh@kyonggi.ac.kr

2013년 경기대학교 컴퓨터과학과(학사)

2013년~현 재 경기대학교 컴퓨터과학과 석사과정

관심분야 : 인공지능, 기계학습, 시맨틱 웹

김 인 철

e-mail : kic@kyonggi.ac.kr

1985년 서울대학교 수학과(학사)

1987년 서울대학교 전산과학과(이학석사)

1995년 서울대학교 전산과학과(이학박사)

1996년~현 재 경기대학교 컴퓨터과학과 교수

관심분야 : 인공지능, 기계학습, 지능형시스템