J. Internet Comput. Serv.
ISSN 1598-0170 (Print) / ISSN 2287-1136 (Online)
http://www.jics.or.kr

맵리듀스기반 워크플로우 빅-로그 클러스터링 기법☆

A MapReduce-Based Workflow BIG-Log Clustering Technique

진 민 혁[1] 김 광 훈[2*]
Min-Hyuck Jin Kwanghoon Pio Kim

요 약

본 논문에서는 분산 워크플로우 실행 이벤트 로그를 수집하고 분류하기 위한 사전 처리 도구로서 맵-리듀스기반 클러스터링 기법을 제안한다. 특히 우리는 볼륨, 속도, 다양성, 진실성 및 가치와 같은 BIG 데이터의 5V 속성에 만족하고 잘 충족되어 있기 때문에 분산 워크플로우 실행 이벤트 로그를 특별히 워크플로우 빅-로그(Workflow BIG-Logs)라고 정의한다. 이 논문에서 개발하는 클러스터링 기술은워크플로우 빅-로그를 기반으로 하는 특정 워크플로 프로세스 마이닝 및 분석 알고리즘의 사전 처리 단계에 적용하기 위한 목적으로 고안된 것이다. 즉, 맵리듀스(Map-Reduce) 프레임워크를 워크플로우 빅-로그 처리 플랫폼으로 사용하고, IEEE XES 표준 데이터 형식을 지원하며, 결국 본 연구에서 개발중에 있는 구조적 정보제어넷기반 워크플로우 프로세스 마이닝 알고리즘인 ρ알고리즘의 사전 처리 단계 전용으로 사용되도록 구현된 것이다. 보다 자세하게 말하자면, 워크플로우 빅-로그의 클러스터링 패턴은 단위업무 액티비티 기반 클러스터링 패턴과 단위업무 수행자 기반 클러스터링 패턴으로 분류되는데, 특별히 단위업무 액티비티 패턴의 하나인 시간적 워크케이스 패턴과 그의 발생 건수를 재발견하는 맵리듀스 기반 클러스터링 알고리즘을 설계하고 구현하고자 한다. 마지막으로, 우리는 BPI 챌린지에서 공개한 워크플로우 실행 이벤트 로그 데이터세트에 대해 일련의 실험을 수행함으로써 제안된 클러스터링 기법의 기술적 타당성을 검증한다.

☞ 주제어: 워크플로우 프로세스 마이닝, 구조적 정보제어넷 모델, 워크플로우 실행로그, 워크플로우 빅-로그, 시간적 워크케이스, 시간적 업무전달-케이스, IEEE XES 이밴트 스트림 포맷, 하둡 맵리듀스 프레임워크

ABSTRACT

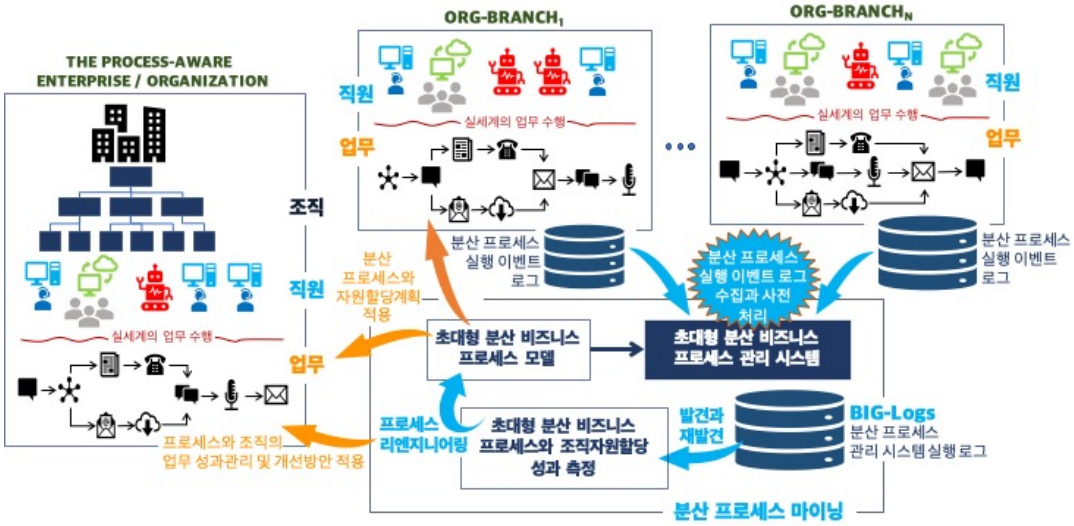
In this paper, we propose a MapReduce-supported clustering technique for collecting and classifying distributed workflow enactment event logs as a preprocessing tool. Especially, we would call the distributed workflow enactment event logs as Workflow BIG-Logs, because they are satisfied with as well as well-fitted to the 5V properties of BIG-Data like Volume, Velocity, Variety, Veracity and Value. The clustering technique we develop in this paper is intentionally devised for the preprocessing phase of a specific workflow process mining and analysis algorithm based upon the workflow BIG-Logs. In other words, It uses the Map-Reduce framework as a Workflow BIG-Logs processing platform, it supports the IEEE XES standard data format, and it is eventually dedicated for the preprocessing phase of the ρAlgorithm that is a typical workflow process mining algorithm based on the structured information control nets. More precisely, The Workflow BIG-Logs can be classified into two types: of activity-based clustering patterns and performer-based clustering patterns, and we try to implement an activity-based clustering pattern algorithm based upon the Map-Reduce framework. Finally, we try to verify the proposed clustering technique by carrying out an experimental study on the workflow enactment event log dataset released by the BPI Challenges.

☞ keyword: workflow process mining, structured information control nets, workflow process enactment event logs, temporal workcase, temporal worktransference, XES event stream data format, Hadoop MapReduce Framework

1. 서 론

본 논문에서는 그리드 및 클라우드 컴퓨팅 환경을 기반으로 하는 분산 워크플로우 관리 시스템[10]에서의 분산 워크플로우 실행로그로부터 워크플로우 프로세스 또는 그와 관련된 지식을 발견하고 재발견하는 프로세스 마이닝 알고리즘을 위한 사전처리 (preprocessing) 기법을 제안한다. 엔터프라이즈 비즈니스 프로세스 또는 워크플로우 모델과 그의 관리 시스템인 비피엠 기술의 등장과

[1] Dept. of Computer Science, Graduate School, Kyonggi Univ., 16227, Korea.
[2] Division of Computer Science and Engineering, Kyonggi Univ., 16227, Korea.
* Corresponding author (kwang@kgu.ar.kr)
[Received 19 November 2018, Reviewed 21 November 2018, Accepted 5 December 2018]
☆ 본 연구는 경기대학교 일반대학원 연구원장학생프로그램의 지원을 받아 수행되었음.



(그림 1) 분산 비즈니스 프로세스 관리 및 마이닝 시스템 환경
(Figure 1) A Conceptual Environment of Distributed Process Management and Mining Systems

발전이 거의 30여년 이상이 지속되면서 그 이 후 많은 조직들이 이 엔터프라이즈 업무프로세스 자동화 기술의 도입과 적용을 성공적으로 완성시켰고, 기술의 확산속도 역시 지속적으로 증가하고 있다. 결과적으로, 프로세스기반 조직의 증가와 함께 업무프로세스 모델의 적용과 그의 실행 이벤트 로그 데이터 역시 급속히 증가하게 됨에 따라 기존의 업무프로세스 모델에 대한 재설계와 리엔지니어링의 필요성과 자원할당에 대한 재계획과 재배치의 긴급성이 대두되는 시점이 도래한 것이다. 즉 다시 말해서, 프로세스 리엔지니어링 기술과 프로세스 마이닝 기술에 대한 연구개발의 시대가 시작된 것이다. 특히, 조직의 대형화와 분산화 그리고 그에 따른 업무프로세스 모델의 초대형화는 그리드 또는 클라우드 컴퓨팅 환경을 기반으로 하는 분산 워크플로우 또는 비즈니스 프로세스 (이하 분산 프로세스) 관리 시스템과 분산 프로세스 마이닝 기술의 필요성을 증가시키고 있다. 그림 1은 이러한 대형화 조직에서의 분산 비즈니스 프로세스 모델[10]과 그의 분산 실행 이벤트 로그[4][5][6][7][8]를 연구배경으로 하는 분산 프로세스 관리 시스템과 분산 프로세스 마이닝 기술의 개념과 범위를 그림으로 나타낸 것이다. 본 논문의 범위는 분산 워크플로우 모델의 실행 이벤트 로그의 수집과 사전처리 기법과 관련된다.

특히, 그림에서 나타내었듯이, 해당 조직의 지사로부터 분산 프로세스 모델의 실행이벤트 로그를 수집하여 사전처리를 하는 작업은 크기·생성속도·다양성·정확성·가치의 고수준 5V로 대표되는 빅 데이터의 기본속성을 만족시키므로, 본 논문에서는 이러한 대규모의 분산 프로세스 실행 이벤트 로그를 **"워크플로우 빅-로그 (BIG-Logs)"**라고 정의한다. 또한, 이러한 워크플로우 빅-로그를 기반으로 하는 워크플로우 프로세스 마이닝 및 분석 알고리즘을 개발하기 위해서 반드시 요구되는 적절한 사전처리 기법을 제안하고 구현하는 것이 본 논문의 연구목표이다. 즉, 본 논문에서 제안하는 사전처리 기법은 분산처리프레임워크인 하둡기반의 맵리듀스[11][12]를 통한 분산처리와 IEEE XES 표준 데이터포맷[8]에 따른 워크플로우 빅-로그을 기반으로 하며, 궁극적으로 대표적인 정보제어넷기반 워크플로우 프로세스 마이닝 알고리즘인 ρ-Algorithm[9]을 지원하기 위한 클러스터링 사전처리 기법이다. 워크플로우 빅-로그의 클러스터링 유형을 액티버티 중심 클러스터링 유형과 수행자 중심 클러스터링 유형으로 구성하고, 각 유형에 따른 맵리듀스기반 클러스터링 알고리즘을 설계 및 구현하는 것이다. 끝으로, 구현된 워크플로우 빅-로그 클러스터링 기법을 프로세스 마이닝 연구를 위해 워크플로우 실행이력 로그 데이터세트 제공하고 있는 4TU.Center[4]로부터 확보한 실제 데이터세트에 적용함으로써 그의 구현내용과 적용가능성을 검증한다.

본 논문의 구성은 다음 장에서 연구주제인 분산 프로세스 마이닝과 대용량 데이터의 분산처리 기술과 관련된

이전의 연구내용을 소개하고, 이어지는 연속된 장에서 본 논문의 제안하는 분산 프로세스 마이닝 프레임워크와 그를 위한 분산데이터처리 플랫폼인 하둡 맵리듀스기반의 사전처리 기법과 그의 핵심 알고리즘을 기술하고, 이의 적용사례와 실행결과를 기술한다.

2. 관련 연구

앞서 기술했듯이, 그리드와 클라우드 컴퓨팅 환경의 확산과 더불어 이를 기반으로 하는 ① 분산 워크플로우 관리 시스템[10][11][12]의 아키텍처와 시스템 연구개발, 이러한 분산 워크플로우 관리 시스템을 기반으로 하는 ② 초대형 조직과 워크플로우 프로세스 모델[10][13]의 증가, 그러한 초대형 조직을 기반으로 하는 초대형 분산 워크플로우 모델들과 그의 운용기간이 장기간 지속됨에 따라 수반되는 ③ 초대형 실행이력과 이벤트로그[6][7][8], 그 초대형 실행이력로그로부터 프로세스 모델 자체 뿐 만 아니라 조직의 자원관리 의사결정에 활용될 핵심적인 경영관리 지식을 발견·재발견하는 ④ 프로세스 마이닝 알고리즘과 분석 기법[1][2][3][5][9]들이 본 논문과의 주요 핵심연구기술과 직·간접적인 관련성이 매우 높은 연구개발 이슈들이다. 따라서, 본 장에서는 이러한 세부 관련연구들에 대한 연구개발 현황과 본 논문에서 제안하는 클러스터링 기법과의 관련성과 상호적용범위에 대한 사전조사결과를 기술하고자 한다.

① 분산 워크플로우 관리 시스템: 본 논문의 연구주제와 밀접한 관련이 있는 분산 프로세스 마이닝 기술은 클라우드 컴퓨팅 환경에서 실행되는 분산 워크플로우 관리 시스템을 기반으로 한다. K. Kim [10]의 연구에서는 초대형 워크플로우 모델을 분산 워크플로우 엔진에 합리적으로 배포하기 위한 워크플로우 모델 배포 방법을 제안하였으며, 이는 특히 클라우드 컴퓨팅 환경에서 작동하는 분산 워크플로우 관리 시스템에서의 워크플로우 배포 방법론의 하나로 워크플로우 모델을 조각화하는 방법과 이를 통한 분산 워크플로우 엔진 아키텍처 및 시스템을 구축하는 방법으로 구성되어 있다. 결과적으로, 이와 같은 클라우드 컴퓨팅 기반의 분산 워크플로우 엔진들로부터 실행이력 즉 이벤트로그들을 수집하고, 수집된 초대형 이벤트로그들(워크플로우 빅-로그)을 특성에 따라 클러스터링하는 방법이 본 논문의 궁극적 연구목표이다.

② 초대형 조직과 워크플로우 프로세스 모델: 워크플로우 프로세스 모델의 초대형 속성은 하나의 프로세스를 구성하는 액티버티의 수, 하나의 프로세스로부터 생성되는 인스턴스 (또는 워크케이스)의 수, 그리고 하나의 프로세스에 참여하는 수행자의 수 등이 있을 수 있는데, K. Kim[10]의 연구에서는 하나의 프로세스를 구성하는 액티버티의 수가 수 십 여개에서 수 백 여개에 달하는 초대형 워크플로우 프로세스 모델을 횡적 또는 종적으로 단편화하여 분산시키는 방법과 그에 따른 분산 워크플로우 아키텍처를 제안하였고, K. Kim, H.J. Ahn[13]은 하나의 프로세스로부터 생성되는 인스턴스의 수가 수십만 또는 수백만 여개에 달하는 초대형 워크플로우 프로세스 모델을 효율적으로 실행시키기 위한 EJB기반 분산 워크플로우 아키텍처와 시스템을 설계하고 구현하였다. 본 논문에서는 이러한 초대형 분산 워크플로우 관리 시스템을 기반으로 백 여개의 액티버티들로 구성되는 프로세스가 수만 여개의 인스턴스들을 생성하고 백 여명의 수행자들의 의해 수행되는 초대형 워크플로우 프로세스 모델을 실행하고 처리하면서 저장시킨 초대형의 이벤트 로그로부터 프로세스 및 그의 관련 지식을 발견하고 재발견하는데 있어서 반드시 선행되어야 하는 대용량 데이터 사전처리 과정을 지원하는 효율적인 기법을 제안하는 것이 본 논문의 궁극적인 목표이다.

③ 초대형 워크플로우 실행이력과 이벤트 로그: 분산 워크플로우 엔진의 로깅 및 감사 컴포넌트[5]는 워크플로우 인스턴스 및 액티버티의 실행 이벤트를 로그 데이터 센터에 기록하고, 기록 된 이벤트 로그는 워크플로우 프로세스 모델, 워크플로우 인스턴스의 이벤트 트레이스 그리고 단위 액티버티의 이벤트 로그를 축으로 한 3차원 데이터 큐브를 구축할 수 있다. 지금까지 워크플로우 실행 이벤트 로그의 구조와 형식에 대한 여러 연구가 있었습니다. M. Park, K. Kim[6]의 연구에서는 워크플로우 마이닝을 위한 XML 기반 워크플로우 실행 이벤트 로그 언어로서 XML기반의 XWELL을 제안하였다. 워크플로우 기술의 국제 표준화기구인 WfMC는 표준화 된 감사 및 로그 포맷에 관한 규격으로 BPAF[7]를 발표했다. 최근 IEEE에서는 XES[8]를 발표했으며, 이 XML기반의 이벤트 로그 표준 포맷의 목표는 정보 시스템 설계자에게 이벤트 로그 및 이벤트 스트림을 통해 시스템의 동작을 캡처할 수 있는 통합되고 확장 가능한 이벤트 로그 저장 방법론을 제공하는데 있다. 본 논문에서는 이 XES 이벤트 로그 포맷을 기반으로 하는 실제 데이터세트를 대상으로 실험을 수행하고자 한다. 현재까지 워크플로우 실행 이벤트 로그 데이터세트를 제공하고 관리하는 기관은 4TU.Centre for Research Data[4]이다.

④ 프로세스 마이닝 알고리즘과 분석 기법: 본 논문의 주요 연구주제인 분산 프로세스 마이닝과 분석 기법과 관련되어 지금까지 여러 연구사례가 수행되어 왔다. 특히, 프로세스 마이닝 알고리즘들과 관련된 대표적인 연구결과는 W.M.P. van der Aalst, A.J.M.M. Weijters[1]의 알파-알고리즘(αAlgorithm)과 K. Kim, C.A. Ellis[3]의 시그마-알고리즘(σAlgorithm) 그리고 K.-S Kim, et al.[2]의 로-알고리즘(ρAlgorithm) 등이 있으며, 프로세스 기반 지식 발견 기법에 관한 대표적인 연구결과는 M. Park, et al.[14]의 워크플로우기반 소셜네트워크 지식 발견 기법 등이 있다. 알파-알고리즘은 페트넷기반의 워크플로우 프로세스 모델을 실행이력 이벤트 로글로부터 재발견하는 알고리즘이고, 시그마-알고리즘과 로-알고리즘은 정보제어넷기반의 워크플로우 프로세스 모델을 실행이력 이벤트 로그로부터 재발하는 알고리즘이다. 특히, 로-알고리즘은 워크플로우 프로세스를 구성하는 4가지 기본적인 프로세스 패턴, 즉 순차적, 선택적, 병렬적, 반복적 프로세스 패턴을 재발견하는데 있어서 각 액티버티의 실행 회 수를 활용하는 접근방법으로서 본 논문의 클러스터링 접근방법을 통한 사전처리 결과를 가장 적절히 활용할 수 있는 알고리즘이다. 본 논문에서 제안하는 클러스터링 기법은 바로 이러한 프로세스 마이닝 알고리즘과 지식 발견 기법의 입력데이터, 즉 워크플로우 실행이력 이벤트 로그에 대한 매우 시의적절한 사전처리 기법으로 활용될 수 있다.

3. 맵리듀스기반 클러스터링 기법

본 논문의 핵심 연구내용은 앞서 제시한 바와 같이 분산 컴퓨팅 환경을 기반으로 하는 초대형 워크플로우 프로세스 모델의 실행이력 이벤트 로그를 특별히 “워크플로우 빅-로그”라고 칭하고, 이 워크플로우 빅-로그로부터 프로세스나 관련 지식을 마이닝하는 워크플로우 발견 및 재발견 기법들을 지원하기 위한 이벤트 로그 사전처리 기법의 하나로 하둡의 맵리듀스 플랫폼을 이용하는 접근방법을 제안하고 구현하는 것이다. 특히, 워크플로우 빅-로그는 초대형 워크플로우 프로세스 모델의 실행이력 이벤트로그로서 초대형 여부를 판단하는 정량적 기본 속성을 만족한다고 가정하는데, 그 기본 속성은 조직내의 워크플로우 모델의 수, 각 워크플로우 모델을 구성하는 액티버티의 수, 각 워크플로우 모델의 실행 인스턴스의 수, 각 워크플로우 모델의 실행에 참여하는 수행자의 수 그리고 각 워크플로우 모델의 실행시 발생되는 이벤트의 수이다. 본 장에서는 제안하는 하둡 맵리듀스기반의 워크플로우 빅-로그 클러스터링 기법을 구성하는 알고리즘을 비롯한 각 구성요소에 대한 정형적 정의와 세부 포맷을 자세히 기술하고자 한다.

3.1 워크플로우 빅-로그의 정형적 정의

초대형 워크플로우 프로세스 모델의 인스턴스가 실행될 때 워크플로우 엔진의 로깅 함수는 각 액티버티의 실행 이벤트를 로그 저장소에 기록하는데, 본 논문에서는 기록된 각 이벤트의 XML 포맷으로 IEEE XES 표준 이벤트 스트림 태그 언어를 전제로 한다. 그림 2에서 나타낸 바와 같이, 워크플로우 빅-로그를 구성하는 각 워크플로우 프로세스 인스턴스의 단위 액티버티들의 실행 이벤트 로그에 대한 정형적 정의는 다음과 같다.

[정의 1] 워크플로우 액티버티 인스턴스 이벤트 로그: $we = (\alpha, pc, wf, wc, ac, p^*, t, s)$

- α = 액티버티 (activity instance identifier) 식별자
- pc = 패키지 식별자
- wf = 워크플로우 프로세스 식별자
- ac = 워크플로우 인스턴스 식별자
- p^* = 수행자 식별자
- t = 타임스탬프 (timestamp)
- s = 액티버티 인스턴스의 상태 (state): *ready, assigned, reserved, running, completed, cancelled*

동일한 워크플로우 인스턴스 식별자를 갖는 워크플로우 액티버티 인스턴스 이벤트 로그들은 결과적으로 해당 워크플로우 인스턴스의 실행이력 트레이스 (workflow instance activity event trace)를 구성하게 되는데, 이를 워크플로우 인스턴스 액티버티 이벤트 트레이스라고 저의하며, 프로세스 마이닝 기법의 기본적인 처리단위가 된다. 워크플로우 인스턴스 액티버티 이벤트 트레이스에 대한 정형적 정의는 다음과 같다.

[정의 2] 워크플로우 인스턴스 액티버티 이벤트 트레이스: $WT(c) = (we_1, \cdots, we_n)$, c = 워크플로우 인스턴스,

$\{ we_i \mid we_i.wc = c \wedge we_i.t \le we_j.t \wedge we_i.pc = we_j.pc \wedge we_i.wf = we_j.wf \wedge we_i.wc = we_j.wc \wedge i < j \wedge 1 \le i,j \le n \}$

궁극적으로, 각 이벤트 로그의 속성인 타임스탬프와 액티버티 인스턴스의 상태 속성을 고려하여 워크플로우 인스턴스 액티버티 이벤트 로그를 타임스탬프 시간 순서대로 정렬시킨 이벤트 로그 시퀀스를 워크플로우 인스턴

스 액티버티 이벤트 트레이스라고 정의할 수 있다. 여기서 의미있는 시간 순서는 다음과 같은 타임스탬프의 유형에 따라 정의될 수 있다.

- 예정 시간 : 작업 항목의 상태가 준비 (READY)에서 할당 (ASSIGNED)로 변경될 때 취해진다.
  wet.s ⇒ (*t* = *we.t* ∧ *s* = *we.s* ∧ *s* = '*assigned*')
- 할당 시간 : 작업 항목의 상태가 준비에서 예약 (RESERVED)로 변경될 때 취해진다.
  wet.e ⇒ (*t* = *we.t* ∧ *s* = *we.s* ∧ *s* = '*assigned*')
- 시작 시간 : 작업 항목의 상태가 예약에서 수행 (RUNNING)로 변경될 때 취해진다.
  wet.u ⇒ (*t* = *we.t* ∧ *u* = *we.s* ∧ *u* = '*running*')
- 완료 시간 : 작업 항목의 상태가 수행에서 완료 (COMPLETED)로 변경될 때 취해진다.
  wet.o ⇒ (*t* = *we.t* ∧ *o* = *we.s* ∧ *o* = '*assigned*')

3.2 맵리듀스 기반 클러스터링 알고리즘

앞서 정의한 워크플로우 빅-로그에 대한 정형적 정의를 이론적 기반으로 하는 워크플로우 실행이력 이벤트 로그로부터 발견 또는 재발견할 수 있는 지식의 유형에 따라 다양한 형태의 프로세스기반 데이터를 구축해야 하고, 이를 위한 사전처리를 필요로 한다. 본 논문에서는 이러한 사전처리 방법의 하나로서 하둡기반의 맵리듀스 플랫폼을 적용하고자 한다. 즉, 워크플로우 빅-로그로부터 프로세스 모델을 재발견하는 알고리즘을 설계하는데 있어서 중요한 사전처리 단계의 하나인 각 이벤트 트레이스를 시간적 워크케이스로 변형시키고, 동일한 시간적 워크케이스들을 하나의 패턴으로 클러스터링시킨 후, 각 패턴의 발생 회 수를 발견하는 단계를 바로 맵리듀스 플랫폼을 이용하여 수행하는 방법과 알고리즘을 제시하고자 한다.

[정의 3] 시간적 워크케이스 모델, TWCϕ (c): 특정 워크플로우 인스턴스 c에 해당하는 시간적 워크케이스 모델은 이벤트 로그의 타임스탬프 유형에 따라 4가지 유형의 시간적 워크케이스 모델이 정의될 수 있다.

- $\phi \in$ { s, e, u, o }
- TWCϕ (c) = $(we_{\alpha_1}^{\tau[.\phi]}, \cdots, we_{\alpha_m}^{\tau[.\phi]})$:
  $\{ \mathrm{we}_{\alpha}^{\tau[.\phi]} | \alpha = \mathrm{we.ac} \wedge \tau = \mathrm{we.t} \wedge \phi \in \{\mathrm{s,e,u,o}\} \wedge we_{\alpha}.wc = c \wedge (we_{\alpha_i}^{\tau_i} \prec we_{\alpha_j}^{\tau_j})^{10} \wedge \tau_i \prec \tau_j \wedge i < j \wedge 1 \le i,j \le m \}$,

정의 3은 바로 각 이벤트 트레이스에 대한 시간적 워

ω-MapperReducer: The Workflow BIG-Log Clustering Algorithm

```
Input  All Temporal Workcases (TWC), T;  // From the Workflow BIG-Logs
Output A set of Temporal Workcase Patterns, P; // 시간적 워크케이스 패턴
       Counts of the Patterns, all sum_p (p ∈ P); // 시간적 워크케이스 패턴의 발생 건수

Begin Procedure
      class ω-Mapper
      class ω-Reducer
End Procedure

class ω-Mapper:
    Organizing key and value of the mapper  // key: line#-TWC, value: string-TWC
    method Map ( TWC-key a, TWC-value t )
        for all t do
           clustering p ← t;
           call Emit ( pattern p, count weight );  // weight = 1 (default)
        done;

class ω-Reducer:
    method Reduce ( pattern-key p, pattern-values W )  // W = all of the count weight
        sum_p ← 0;
        for all w ∈ W do
           sum_p ← sum_p + w;
           call Emit ( pattern p, count sum_p );
        done;
```

(그림 2) ωMapperReducer: 시간적 워크케이스 기반 워크플로우 빅-로그 클러스터링 알고리즘
(Figure 2) A Clustering Algorithm for Clustering Temporal Workcases from Workflow BIG-Logs

크케이스 모델에 대한 정형적 정의를 나타낸 것이다. 즉, 타임스탬프를 중심으로 시간적으로 정렬된 워크플로우 인스턴스 액티버티 이벤트 시퀀스에 대한 정형적 모델인 시간적 워크케이스 모델을 정의한 것이다. 결과적으로, 시간적 워크케이스 모델은 예정 시간, 할당 시간, 시작 시간 및 완료 시간과 같은 타임스탬프의 유형에 따라 다음과 같이 4 가지 유형의 시간적 워크케이스 모델로 구분될 수 있다.

- 예정시간 (ϕ = s) 기반 시간적 워크케이스 모델
- 할당시간 (ϕ = e) 기반 시간적 워크케이스 모델
- 시작시간 (ϕ = u) 기반 시간적 워크케이스 모델
- 종료시간 (ϕ = o) 기반 시간적 워크케이스 모델

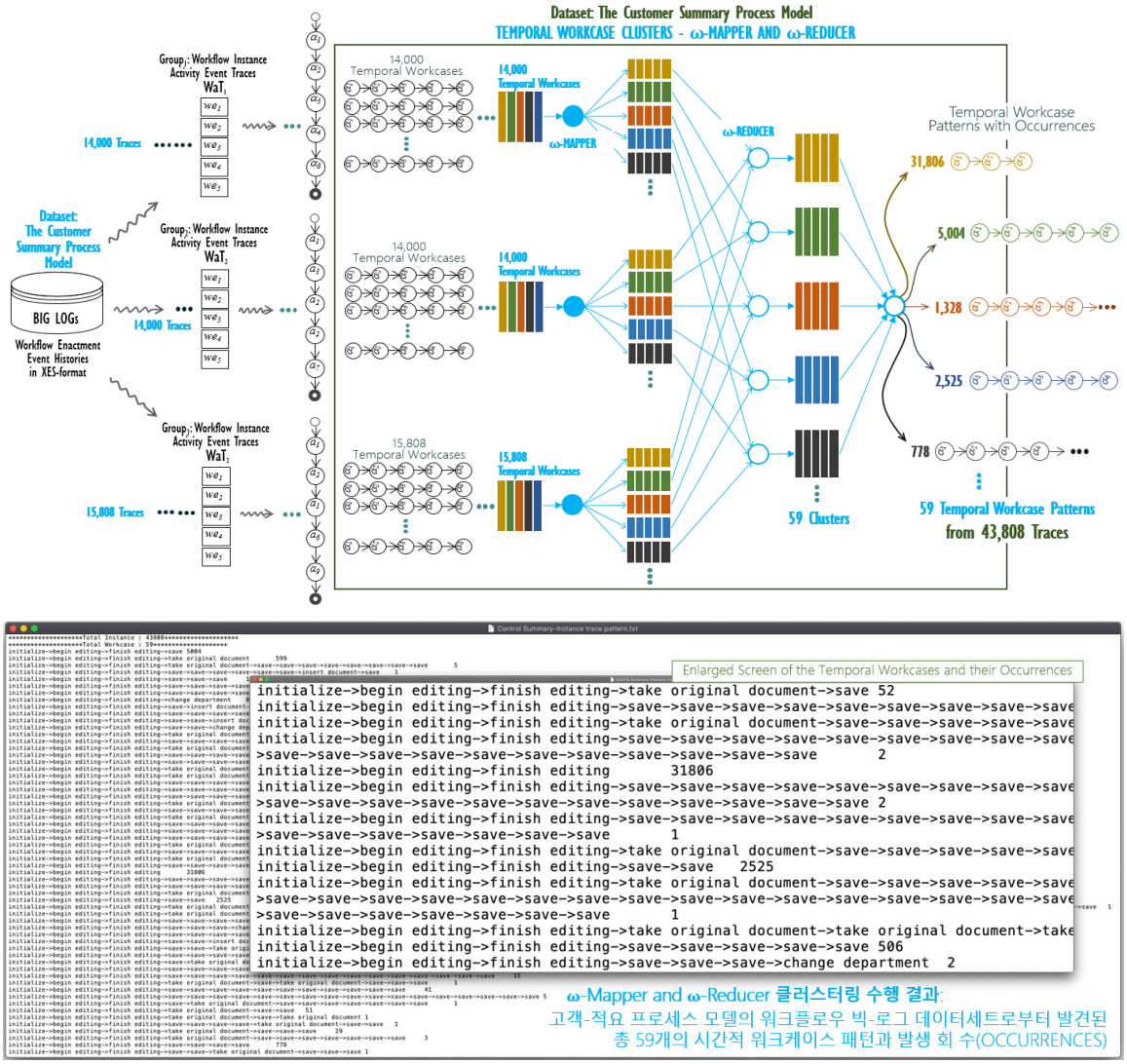
본 논문에서 제안하는 그림 2의 워크플로우 빅-로그 클러스터링 알고리즘(ω-MapperReducer)의 기반기술인 맵리듀스 플랫폼[15]은 구글에서 대용량 데이터 처리를 분산 병렬 컴퓨팅에서 처리하기 위한 목적으로 개발된 소프트웨어 프레임워크로서 일반적으로 사용되는 Map과 Reduce라는 함수 기반으로 구동된다. 맵리듀스의 입력은 각 레코드가 (Key, Value)의 쌍으로 구성되는 일련의 리스트형태를 갖는다. 따라서, 본 논문의 알고리즘의 입력으로 워크플로우 빅-로그를 구성하는 시간적 워크케이스를 Value(TWC-value)로 하고, 그의 파일내 라인번호를 Key(TWC-key)로 하는 일련의 리스트들을 조직한다. **Map()** 함수는 이 (TWC-key, TWC-value) 쌍의 리스트를 읽어 다시 (pattern-key, count-value) 형태의 중간 결과를 **Emit()** 함수를 통해 출력한다. 이 중간 결과들은 pattern-key를 기준으로 동일한 값으로 클러스터화시킨 (pattern-key, count-values[]) 쌍의 리스트 형태로 구성된다. **Reduce()** 함수는 각 pattern-key 값이 해당되는 (pattern-key, pattern-values) 쌍의 리스트들에 대해 집계연산을 수행하고, 최종 결과, 즉 각 패턴 (시간적 워크케이스 패턴)과 그의 발생 총합인 (pattern p, count sum_p)을 **Emit()** 함수를 통해 출력한다. 참고로, **Map()** 함수의 수행결과인 (pattern-key, count-values[]) 쌍의 리스트에서 count-value는 가중치 (weight) 개념을 통해 각 패턴-키 리스트 값에 대한 가치 수준을 달리 적용할 수 있다. 본 논문의 알고리즘에서는 모든 패턴-키 리스트 값의 가중치는 **1**로 정하여 그 가치 수준이 동일하다고 가정한다.

맵리듀스기반 워크플로우 빅-로그 클러스터링 알고리즘의 입력과 출력 파일은 블록기반 분산파일시스템을 이용한다. 즉, 블록기반 분산파일시스템에서 각 입출력 파일은 고정크기의 블록단위(기본 64MB)로 관리되며, 각 블록은 장애시의 복구를 위해 기본적으로 2개의 복사본을 갖는다. 맵리듀스 작업은 기본적으로 **Map()** 함수를 수행하는 맵-단계와 Reduce() 함수를 수행하는 리듀스-단계로 구성된다. 맵-단계에서 입력 파일의 각 블록은 **Map()** 함수를 수행하는 각 태스크 즉 매퍼(ω-Mapper)에게 전달되고, 각 매퍼는 **Map()** 함수를 각 블록내의 리스트 (TWC-key, TWC-value)들에 적용한다. **Map()** 함수의 수행결과는 이를 수행한 각 컴퓨팅 노드의 로컬디스크에 기록되는데, 이는 시스템 장애 시에 중간결과에 대한 장애복구를 지원하기 위한 것이다. **Reduce()** 함수를 수행하기에 앞서 매퍼는 중간결과에 대하여 pattern-key를 기준으로 클러스터화를 수행하는데, 이것은 pattern-key 값에 대한 정렬을 통해 이루어진다. 그리고 정렬을 통해 클러스터화된 (pattern-key, pattern-values[]) 리스트들에 대하여 Reduce() 함수를 수행할 각 태스크 즉 리듀서(ω-Reducer)로의 전달에 앞서 내부적으로 **Combine()** 함수를 적용한다. 이 **Combine()** 함수는 집계연산을 맵-단계에서 미리 수행함으로써, 매퍼에서 리듀서로 데이터를 전달하는데 요구되는 입출력비용을 감소시키기 위함이다.

참고로, 리듀서들은 모든 매퍼가 종료된 시점에 HTTPS를 통해 각 매퍼들의 로컬디스크에 기록된 중간결과들을 입력받는다. 이 때 각 리듀서는 자신이 처리하도록 할당된 패턴-키(pattern-key) 값에 대한 모든 리스트들을 모든 매퍼로부터 전달받는다. 이 과정은 마치 카드놀이의 패를 섞는 행위와 유사해 셔플링(shuffling)이라고 칭하며, 이렇게 모인 각 패턴-키에 대한 모든 값들은 하나의 리스트로 병합된다. 이 후, 각 패턴-키 값에 대하여 **Reduce()** 함수를 적용하고 그 결과를 분산 파일시스템에 기록한다. 결과적으로, 본 논문에서는 이와 같은 원리에 의해 구동하는 시간적 워크케이스 기반의 워크플로우 빅-로그 클러스터링 알고리즘을 구현하였으며, 이 구현된 알고리즘을 기반으로 실제 워크플로우 프로세스 모델의 실행이력 이벤트 로그 데이터세트에 대한 실험적 적용사례연구를 수행하고자 한다.

3.3 제안된 클러스터링 알고리즘의 구현과 실험

앞서 기술한 맵리듀스기반 워크플로우 빅-로그 클러스터링 알고리즘을 자바언어와 리눅스기반의 하둡 개발도구를 통해 구현하였으며, 이를 이용하여 4TU.Centre for Research에서 제공한 BPI Challenges 2018[4]의 공개 데이터세트의 하나인 고객-적요 서브프로세스 모델(Customer Summary SubProcess Model)의 실행이력 이벤트 로그 데



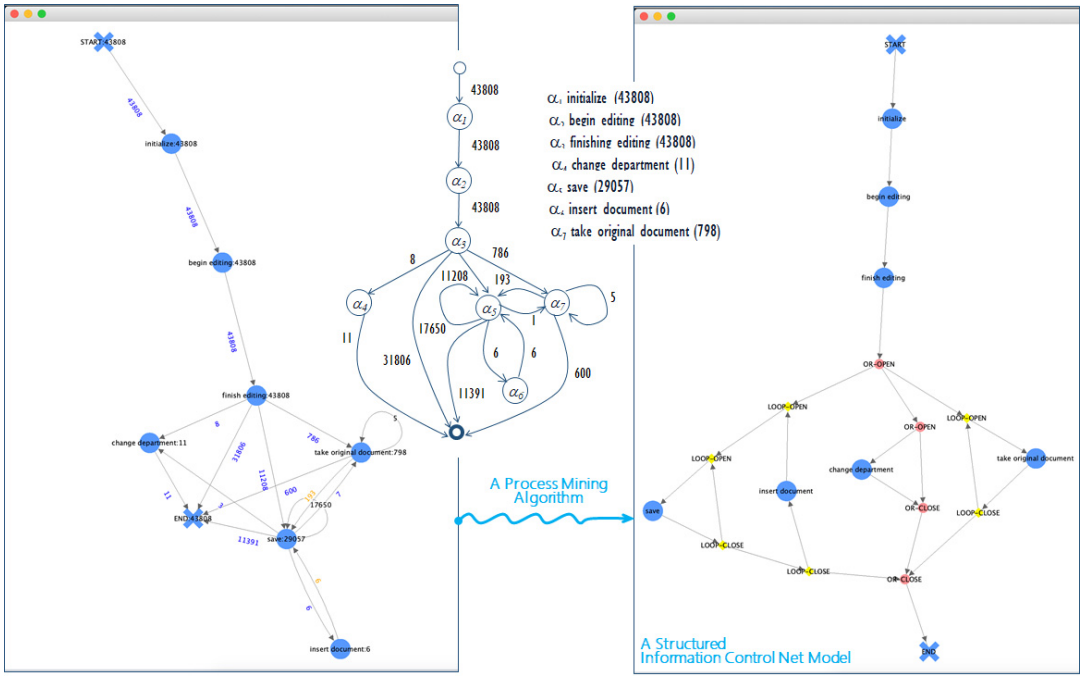
(그림 3) 고객-적요 프로세스 모델의 워크플로우 빅-로그에 대한 실제 클러스터링 실행 결과

(Figure 3) A Clustering Result from Workflow BIG-Logs of the Customer Summary Process Model

이터세트에 대한 실험적 적용사례연구를 수행하였다. 이 데이터 세트에는 총 43,808개의 워크플로우 인스턴스 실행에 따른 이벤트 트레이스를 기록하고 있으며, 기록된 모든 워크플로우 인스턴스 트레이스에는 해당 액티버티 이벤트 로그가 포함되어 있다. 또한, 이 서브프로세스 모델은 7개의 단위업무 액티버티로 구성되어 있다. 그림 3은 이러한 데이터세트에 적용한 시간적 워크케이스기반 워크플로우 빅-로그 클러스터링 알고리즘의 맵리듀스 플랫폼기반의 구현 상황과 그의 실행결과인 총 59개의 시간적 워크케이스 패턴들과 각 패턴의 발생 회 수를 나타낸 것이다.

3.4 실험적 적용사례연구 결과의 활용

본 논문에서 제안한 맵리듀스기반 시간적 워크케이스 워크플로우 빅-로그 클러스터링 알고리즘과 그의 구현 그리고 실험적 적용사례연구의 궁극적 목표는 프로세스



(그림 4) 고객-적요 프로세스 모델의 워크플로우 빅-로그 사전처리 결과(시간적 워크케이스 패턴과 발생 건수)와 이를 이용한 정보제어넷기반 워크플로우 프로세스 재발견 결과

(Figure 4) A Rediscovered Information Control Net Model from the Preprocessing Results on the Workflow BIG-Log Dataset of the Customer Summary Process Model

마이닝 기술의 핵심요소인 프로세스 발견 및 재발견 알고즘을 개발하기 위한 데이터 사전처리 기법을 제안하는데 있다. 즉 다시 말해서, 제안한 알고리즘을 기반으로 하는 사전처리 기법은 본 논문의 저자들이 궁극적으로 개발하고자 하는 정보제어넷(information control nets) 기반의 워크플로우 프로세스 재발견 알고리즘을 완성하는데 반드시 필요로 되는 워크플로우 빅-로그 사전처리 방법이다. 그림 4는 앞서 수행한 BPI Challenge 2018 데이터 세트의 사전처리 결과, 즉 시간적 워크케이스 패턴과 그의 발생 건수를 이용한 워크플로우 프로세스 재발발견 알고리즘을 적용하여 궁극적으로 재발견한 정보제어넷 기반 워크플로 프로세스 모델을 나타낸 것이다. 그림에서 볼 수 있듯이, 궁극적으로 재발견된 구조적 정보제어넷기반 고객-적요 프로세스 모델은 2개의 이벤트(시작/끝) 액티버티와 7개의 단위업무 액티버티 (α_1 ~ α_7) 그리고 3개의 반복적 게이트웨이(loop-split/join gateway) 액티버티와 2개의 선택적(exclusive-OR split/join) 게이트웨이 액티버티로 구성되어 있음을 알 수 있다.

4. 결 론

본 논문에서는 초대형 워크플로우 프로세스 모델의 실행이력 이벤트 로그로부터 워크플로우 프로세스 모델을 발견 및 재발견 하기위한 프로세스 마이닝 기술의 사전처리 기법의 하나인 맵리듀스기반 워크플로우 빅-로그 클러스터링 기법을 제안하였다. 특히, 초대형 워크플로우 프로세스 모델의 실행이력 이벤트 로그가 갖는 초대형적 속성, 즉 조직내의 워크플로우 모델의 수, 각 워크플로우 모델을 구성하는 액티버티의 수, 각 워크플로우 모델의 인스턴스 수 측면에서 빅 데이터의 5V 속성을 만족하고 있어, 이를 특별히 워크플로우 빅-로그로 정의하였으며, 이로부터 시간적 워크케이스 기반의 효율적 사전처리 방법으로 워크플로우 빅-로그 클러스터링 알고리즘을 제안하였다. 결과적으로, 제안된 알고리즘의 구현 가능성을 검증하기 위하여 대용량 데이터 처리를 위한 핵심기술로 큰 인기를 누리고 있는 하둡기반의 맵리듀스 플랫폼을 이용하여 구현하였고, 구현된 알고리즘을 실제

워크플로우 프로세스 모델의 실행이력 데이터세트에 적용하는 실험적 적용사례연구를 성공적으로 수행하였다.

참고문헌(Reference)

[1] W. M. P. van der Aalst and A. J. M. M. Weijters, “Process mining: a research agenda,” Journal of Computers in Industry, Vol. 53, Issue 3, 2004.

[2] Kyoungsook Kim, et al., “A Conceptual Approach for Discovering Proportions of Disjunctive Routing Patterns in a Business Process Model,” KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS, Vol. 11, No. 2, pp. 1148-1161, 2017.

[3] Kim, Kwanghoon and Ellis, Clarence A., “σ-Algorithm: Structured Workflow Process Mining Through Amalgamating Temporal Workcases,” The Proceedings of PAKDD2007, Advances in Knowledge Discovery and Data Mining, Lecture Notes in Artificial Intelligence, Vol. 4426, pp. 119-130, 2007.

[4] BPI Challenge 2012, 2013, 2014, 2015, 2016, 2017, 2018, 4TU.Centre for Research Data, https://data.4tu.nl/repository/collection:event-logs-real.

[5] Kim, Kwanghoon, “A XML-BasedWorkflow Event Logging Mechanism for Workflow Mining,” The Proceedings of the International Workshop on APWeb, pages 132-136, 2006.

[6] Minjae Park and Kwanghoon Kim, “XWELL: A XML-Based Workflow Event Logging Mechanism and Language for Workflow Mining Systems,” Lecture Notes in Computer Science, Vol. 4707, pp. 900-909, 2007.

[7] Michael zur Muehlen and Keith D. Swenson, “BPAF: A Standard for the Interchange of Process Analytics Data,” Lecture Notes in Business Information Processing, Vol. 66, pp. 170-181, 2011.

[8] IEEE, “IEEE Standard for eXtensible Event Stream (XES) for Achieving Interoperability in Event Logs and Event Streams,” IEEE 1849-2016, 2016. https://doi.org/10.1109/IEEESTD.2016.7740858

[9] Kim, Kyoungsook, Lee, Youngkoo, Ahn, Hyun., and Kim, Kwanghoon, “An Experimental Mining and Analytics for Discovering Proportional Process Patterns from Workflow Enactment Event Logs,” Proceedings of the International Conference on Big Data Technologies and Applications, Exeter, England, Great Britain, Sept. 4rd-5th, 2018.

[10] Kwanghoon Kim, “A Model-Driven Workflow Fragmentation Framework for Collaborative Workflow Architectures and Systems,” Journal of Network and Computer Applications, Volume 35, Issue 1, pp. 97-110, January 2012.

[11] K. Lee, Y. Lee, H. Choi, Y. F. Chung and B. Moon, “Parallel Data Processing with MapReduce: A Survey,” SIGMOD Record, Vol. 40, No. 4, pp. 11-20, December 2011.

[12] C. Goncalves, L. Assuncao, j. C. Cunha, “Flexible MapReduce Workflows for Cloud Data Analytics,” International Journal of Grid and High Performance Computing, Vol. 5, No. 4, pp. 48-64, 2013.

[13] Kim KH., Ahn HJ., “An EJB-Based Very Large Scale Workflow System and Its Performance Measurement,” In: Fan W., Wu Z., Yang J. (eds) Advances in Web-Age Information Management. WAIM 2005, Lecture Notes in Computer Science, Vol. 3739. pp. 526-535, Springer, Berlin, Heidelberg, 2005.

[14] Minjae Park, Hyun, Ahn, and Kwanghoon Pio Kim, “Workflow-supported social networks: Discovery, analyses, and system,” Journal of Network and Computer Applications, Vol, 75, pp. 355-373, Nov. 2016.

[15] K.-H. Lee, W.J. Park, K.S. Cho, W.Ryu, “The MapReduce framework for Large-scale Data Analysis: Overview and Research Trends,” Electronics and telecommunications trends, vol. 28, No. 6, pp. 156-166, 2013. http://dx.doi.org/10.22648/ETRI.2013.J.280616

◐ 저 자 소 개 ◑

진 민 혁(Min-hyuck Jin)

2017년 2월 경기대학교 이과대학 컴퓨터과학과 학사
2017년 3월～현재 경기대학교 일반대학원 컴퓨터과학과 석사(재학)
관심분야 : 워크플로우/비피엠, Process Discovery/Rediscovery
E-mail : happytoh@kyonggi.ac.kr

김 광 훈(Kwang-hoon Kim)

1984년 2월 경기대학교 이과대학 전자계산학과 학사
1986년 2월 중앙대학교 일반대학원 전자계산학과 석사
1986년 2월～1991년 8월 한국전자통신연구원 연구원
1994년 5월 University of Colorado Boulder, Department of Computer Sciences, MS
1998년 5월 University of Colorado Boulder, Department of Computer Sciences, Ph.D
2005년 3월～2010년 2월 Univ. of Colorado Boulder, Department of Computer Science, 방문교수
2007년 7월～2010년 6월 경기대학교 콘텐츠융합소프트웨어연구센터장
1998년 3월～현재 경기대학교 컴퓨터과학과 교수
2000년 1월～현재 한국인터넷정보학회 이사, 부회장
2002년 3월～현재 비피엠코리아포럼 부회장
2003년 1월～현재 WfMC ERC Vice-chair
2003년 1월～현재 TTA 정보통신국제표준전문가
관심분야 : 워크플로우/비피엠, Process-Aware Information Systems, Process Discovery/Rediscovery, Workflow-supported Social Networks
E-mail : kwang@kyonggi.ac.kr