

연관 규칙 마이닝을 이용한 제품 저장 위치 할당 방식에 관한 연구

정재원^{1†}, 이상엽¹

¹한양대학교 산업공학과

Location assignment using association rule mining in a warehouse

Jaewon Jeong^{1†}, SangYeop Lee¹

¹Department of Industrial Engineering, Hanyang University

Order picking costs account for a significant portion of the logistics system of the warehouse. To increase the efficiency of warehouse operation, order picking operations in the warehouse should be carefully designed. In this work, we study the location assignment problem to study the applicability of the data mining approach in determining the storage locations. Specifically, we present an association rule-based storage assignment algorithm denoted by the Apriori algorithm and the Affinity algorithm. The objective of this work is to minimize the order picker's travel distance by allocating the storage locations using the association rule-based approach. We observe the details for analyzing the efficiency of the proposed approach with the actual online transaction data. The results from the experiments show that the proposed methods outperform the turnover-based assignments.

Keywords: Location assignment, Order picking, Association rule mining, Apriori algorithm, Affinity algorithm

논문접수일 : 2021.10.06.

심사완료일 : 2021.12.16.

게재확정일 : 2021.12.21.

† Corresponding Author: its1129@hanyang.ac.kr

1. 서론

물류 관리 시스템 및 창고 관리는 많은 노동력을 필요로 하는 프로세스이다. 주문된 제품을 보관하고 고객에게 반출하는 프로세스는 기업의 비용에 직접적인 영향을 미치기 때문에, 창고 관리는 그 중요성이 매우 크다(Cergibozan and Tasan, 2019). 그 중에서도 오더 피킹(Order picking)은 창고 운영비에서 높은 비중을 차지하는 작업으로, Tompkins et al.(2010)은 창고 운영비 중 55%까지 오더 피킹 비용으로 소요될 수 있다고 추정하고 있다. 이에 따라 기업들은 시장에서 우위를 점하고자 물류 비용을 줄이기 위한 노력으로 오더 피킹 효율성 개선에 힘쓰고 있다. 또한 오더 피킹은 물류센터 분야에서 고객 만족도 향상 및 창고 운영에 관해 효율적인 상품 반출을 위해 가장 큰 비중을 차지한다. 그러나 오더 피킹의 효율성을 높이기 위해서는 근본적으로 저장 위치 할당 문제의 최적화가 필요하다.

최근 물류 관리 시스템은 주로 컴퓨터 기반의 시스템으로 작업이 이루어지고 있다. 컴퓨터 기반의 물류 관리 시스템은 작업 요청, 자원 관리 또는 제조 기록 데이터를 저장하고 접근하는 장치로 사용된다. 기업에서 재정적인 이익을 극대화하기 위해서는 데이터를 효율적으로 기록하고 유지할 수 있는 컴퓨터 기반의 물류 관리 시스템을 구축하는 것이 필수적이다(Abisoye et al., 2013; Ahmadi et al., 2019). 다양한 기업과 조직에서 이러한 컴퓨터 기반의 시스템을 구축하게 됨에 따라, 데이터 마이닝(Data mining) 분야의 연구는 빠르게 성장하였으며, 갈수록 그 중요성이 대두되고 있다. 데이터 마이닝은 데이터베이스 속의 지식을 발견하는 이론(Knowledge-discovery in databases: KDD)으로, 데이터베이스 기술(Database technology), 정보 검색(Information retrieval), 패턴 인식(Pattern recognition), 머신러닝(Machine learning), 통계(Statistics), 인공지능(Artificial intelligence), 데이터 시각화(Data visualization) 및 고성능 컴퓨팅(High-performance computing)과 같은 여러 분야의 학문에서 연구되고 있다(Han et al., 2011).

본 연구에서는 수리적인 모델에 기반한 창고 내 제품 위치 할당(Storage assignment) 방법론에 데이터 마이닝의 기법 중 연관 규칙 마이닝(Association rule mining)을 적용한 새로운 제품 위치 할당 방법론을 제안하고자 한다. 연관 규칙 마이닝은 장바구니 분석(Market basket analysis)이라고도 불리는 데이터 마이닝 기법으로, 최근에는 추천 시스템 기법으로써 활용되고 있다(Son et al., 2015). 본 연구에서는 고객 주문 데이터 내에서 패턴을 찾고 제품끼리 어떤 관계를 맺는지 분석하기 위해 이 기법을 선택하였다. 또한 실제 온라인 리테일러 회사의 주문 데이터에 이를 적용하여 입출 빈도 기반 보관 방식에 비해 우리가 제안하는 새로운 방식이 오더 피킹 거리를 감소시키는데 효과가 있음을 실험으로 확인하였다. 연구에 사용된 고객 주문 데이터는 UCI에서 제공하고 있는 'Online Retail Data Set' 데이터로써, 영국의 온라인 리테일 회사의 2010년 1월부터 2011년 9월까지의 고객 주문 데이터이다(Chen et al., 2012). 본 연구의 실험에서는 전체 데이터 중 2010년 3월부터 2010년 5월까지의 주문 기록에서 주문 빈도가 가장 높은 120개의 품목에 대한 해당 기간 동안의 고객 주문 데이터를 선택하여 수치실험에 사용하였다.

본 연구의 구성은 다음과 같다. 2절에서는 제품 위치 할당 문제, 오더 피킹 문제, 그리고 데이터마이닝 분야에서의 선행 연구들에 대해 검토하였다. 3절의 본론에서는 실험 구성, 데이터 전처리 및 구축 과정, 제품 위치 할당 알고리즘 및 실험 결과에 대하여 설명하였다. 4절에서는 연구의 결론 및 추후 연구 과제에 대해 논의하였다.

2. 선행 연구 고찰

2.1 위치 할당 문제 (Storage Location Assignment Problem)

창고(Warehouse)의 저장 위치에 제품을 할당하는 문제를 ‘위치 할당 문제(Storage Location Assignment Problem)’이라고 정의한다. 위치 할당 문제는 저장 위치에 제품을 할당하는 데 있어 오더 피킹의 효율성 또는 공간 활용도를 최대화하는 것에 목적을 두고 있다. Hausman et al.(1976)은 처음으로 자동 창고를 위한 저장 위치 할당 방법론을 제시하였다. 해당 연구에서는 세 가지 저장 위치 할당 방법론을 제시하고 있으며, 각각은 랜덤 보관 방식(Random storage assignment), 입출 빈도 기반 보관 방식(Turnover-based assignment), 그리고 클래스 기반 보관 방식(Class-based storage assignment)이다.

Brynzér and Johansson(1996)는 주문 빈도(Frequency)에 따라 동일한 특성을 가진 제품들을 VG(Variant group)으로 그룹화하는 전략을 제안하였다. 제품을 빈도에 따라 분류하여 저장 위치를 할당하는 저장 위치 할당 전략으로 오더 피킹 시간을 단축할 수 있음을 실험을 통해 제시하였다.

Li et al.(2016)은 새로운 동적 할당 문제(Dynamic storage assignment problem)를 정의하고 제품 저장 위치 할당 최적화를 위한 통합 체계를 연구하였다. 또한, 데이터마이닝을 기반으로 하는 새로운 접근법과 전통적 방법론인 ABC 분류법과의 성능 비교를 진행하였다. 해당 연구와 같이 데이터 마이닝 기술을 제품 저장 위치 할당 시스템에 적용한 연구를 통해 위치 할당 시스템의 운영 성능을 개선하기 위한 효과적인 의사결정 개발이 가능할 것으로 관측되고 있다(Reyes et al., 2019).

본 연구에서는 입출 빈도 기반 보관 방식을 제품 저장 위치 할당 방법론으로 적용하여 3.4절의 실험(1)을 진행했다.

2.2 오더 피킹 문제(Order picking problem) 및 라우팅 문제(Routing problem)

Reyes et al.(2019)에 따르면 위치 할당 방법론이 연구되는 가장 일반적인 목적은 오더 피킹 거리의 최소화를 포함한 창고 내 공간 활용과 이동 거리와 관련된 문제라고 정의하고 있다. 이처럼 제품 위치 할당 문제와 오더 피킹 문제에서 주된 연구 목적은 이동 시간 또는 이동 거리의 최소화이다. Pan and Wu(2009)는 마르코프 체인(Markov Chain) 방법을 기반으로 작업자가 작업 라인으로 이동하는 거리를 예측하는 모델을 개발하였다. 또한 제안된 분석 모델을 기반으로 저장 위치 할당 문제의 특성을 분석하고, 단일 피킹 구역(Single picking zone), 크기가 동일하지 않은 구역(Unequal-sized zones), 크기가 동일한 구역(Equal-size zones)의 세 가지 Picking line 형태에서 각각의 제품 위치 할당 최적화 알고리즘을 제시하였다. Ene and Öztürk(2012)는 정수계획(Integer programming)과 유전 알고리즘(Genetic algorithm)을 이용하여 자동차 산업의 저장 위치 할당과 오더 피킹 시스템을 최적화하기 위한 연구를 진행하였다. 이 연구에서는 자동차 산업의 모든 창고 레이아웃에 통합하여 적용할 수 있는 유전 알고리즘 기반의 방법론을 제시하였다.

라우팅(Routing) 문제에 대한 선행 연구로는 다음의 연구들이 있다. Applegate et al. (2007)은 처음으로 외판원 문제(Traveling Salesman Problem: TSP)를 통해 라우팅 조합 최적화 문제의 솔루션을 제시하였다. 이 연구에 따르면 모든 창고 레이아웃에서 라우팅은 TSP 공식을 사용하여 나타낼 수 있다. 또한, TSP에 적용할 수 있는 알고리즘은 특정 조건을 갖추면 오더 피킹 문제에서도 효율적으로 솔루션을 도출할 수 있기 때문에, TSP 문제에서 적용되는 휴리스틱 기법을 오더 피킹 문제를 해결하는 데 사용할 수 있다고 설명한다.

현실에서의 오더 피킹 작업자의 라우팅에는 보편적으로 간단한 휴리스틱 방법을 이용한다. 이것은 모든 레이아웃에 적용 가능한 라우팅 알고리즘이 존재하지 않으며(De Koster et al., 2007), 휴리스틱 방법을 사용하는 것이 최적화된 라우팅 알고리즘을 사용하는 것보다 통로의 혼잡성 면에서 더 좋은 성능을 보여주기 때문이다(Petersen, 1997).

S-shape 라우팅 정책(S-shape routing policy)은 가장 간단한 라우팅 정책 중 하나로, Goetschalckx and Ratliff(1988)에 의해 도입되었으며, 오더 피킹 작업자가 한쪽 끝 통로(Aisle)로 들어가 다른 쪽 끝으로 나가는 방법이다. S-shape 라우팅 정책에서는 오더 피킹 작업자는 요청된 제품이 있는 통로에 도달하면 그 통로에 진입해 완전히 횡단한 이후 그 다음 통로로 이동한다(Cano et al., 2017). 또 다른 라우팅 정책인 반환 정책(Return policy)은 오더 피킹 작업자가 제품을 처리할 때마다 전면 교차 통로(Front cross-aisle)만을 통하여 통로에 드나드는 라우팅 정책이다.(Van Gils et al., 2016).

본 연구의 실험에서는 단일 선택 구역에서 S-shape 라우팅 정책과 반환 정책을 결합한 결합 라우팅 정책(Combined routing policy)을 적용하였다.

2.3 연관 규칙 마이닝 (Association rule mining)

데이터 마이닝은 데이터를 분석하는 도구이며, 다양한 관점에서 데이터를 분석하고 유용한 정보로 요약하는 프로세스이다. 데이터 마이닝 기법에는 이상 탐지(Anomaly detection), 군집화(Clustering), 연관 규칙 마이닝(Association rule learning), 회귀 분석(Regression), 요약(Summarization), 분류(Classification) 알고리즘 등이 존재한다(Han et al., 2011). 이 중 연관 규칙 마이닝은 함께 발생하는 빈도가 높은 항목들 간의 관계를 찾아내는 방법이다(Kumbhare and Chobe, 2014).

Chuang et al.(2012)는 제품 연관(Item-associated) 클러스터링 기법을 통해 위치 할당 문제의 피킹 거리(Picking distance)를 최소화하는 연구를 진행하였다. Ming-Huang Chiang et al.(2014)의 연구에서는 제품 간 관계의 강도와 특성 모두를 측정하는 연관 규칙 마이닝을 위해 'Weighted Support Count(WSC)'라는 새로운 연관성 지표를 제안하였다. 해당 연구에서는 실제 식품 유통 센터의 데이터를 사용한 실험을 통해 이 연관 규칙 마이닝의 적용으로 이동 거리를 최대 13%까지 감소시킬 수 있음을 증명하였다. Bevilacqua et al.(2019)의 연구에서는 연관 규칙 마이닝을 통해 창고의 부품 및 완제품 보관 및 검색 작업 시 낭비되는 시간을 최소화하기 위해 AS(Automated Storage)/RS(Retrieval System)의 제품 저장 위치 할당 방법론을 개발하였고, 실제 사례를 통해 이 방법론의 성능을 검증하였다. Chuang et al.(2014)는 Apriori 알고리즘의 연관 웹 통계(Web statistics) 및 연관 규칙 마이닝을 활용한 인접 구역 지정(Zoning adjacency) 및 저장 위치 할당 정책을 개발하였다.

1) Apriori 알고리즘을 이용한 연관 규칙 마이닝

Apriori 알고리즘은 Agrawal and Srikant(1994)에서 처음으로 제시한 알고리즘으로, 빈발 항목 집합(Frequent itemsets) 마이닝 및 연관 규칙 마이닝에 사용된다. 빈발 항목 집합은 사용자가 설정한 최소 지지도(support) 이상의 지지도를 갖는 항목 집합들이다. Apriori 알고리즘에서는 연관 규칙을 생성하기 위해 먼저 트랜잭션(Transaction) 데이터에서 빈발 항목 집합을 탐색한다. 빈발 항목 집합을 탐색할 때는 먼저 k-항목 집합(k 개의 항목을 가지는 항목 집합)을 사용하여 (k+1)-항목 집합(k+1) 개의 항목을 가지는 항목 집합을 생성한다. 생성된 빈발 항목 집합을 조건 집합(Antecedent)과 결과 집합(Consequent) 두 개의 항목 집합으로 나누어 연관 규칙을 생성하는데, 각각의 연관 규칙은 조건 집합이 발생할 때, 결과 집합 또한 발생한다는 규칙으로, 주문 기록에서의 연관 규칙은 조건 집합의 제품이 주문되면, 결과 집합의 제품도 주문

된다는 의미를 가진다. 또한 생성되는 연관 규칙은 지지도(Support), 신뢰도(Confidence), 향상도(Lift)의 성능 지표를 가지며, 모든 지표는 각각 최소 지지도, 최소 신뢰도, 최소 향상도 이상의 값을 가진다.

각각의 지표들은 수식 (1), (2), (3) 으로 표현할 수 있다. 수식에서 X와 Y는 각각 조건 집합과 결과 집합을 의미한다. 이 때 조건절과 결과절은 항상 상호배반이다. X의 지지도는 데이터 내에서 X가 발생할 확률이고, 규칙의 신뢰도는 X가 발생했을 때 Y가 발생할 조건부 확률이며, 향상도는 X와 Y가 동시에 발생할 확률을 X와 Y가 각각 발생할 확률로 나눈 값으로 X와 Y가 독립일 때 대비 몇 배의 연관성이 있는지를 나타내는 값이다.

$$Support(X) = \frac{freq(X)}{N}, \quad Support(X \rightarrow Y) = \frac{freq(X, Y)}{N} \quad (1)$$

$$Confidence = \frac{freq(X, Y)}{freq(X)} \quad (2)$$

$$Lift = \frac{Support(X \rightarrow Y)}{Support(X) \times Support(Y)} \quad (3)$$

본 연구에서는 Apriori 알고리즘을 이용한 연관 규칙을 입출 빈도 기반 보관 방식에 접목한 제품 위치 할당 방법론으로 적용하여 3.4절의 실험(2)을 진행하였다.

2) Affinity measure을 이용한 연관 규칙 마이닝

Affinity measure를 이용한 연관 규칙 마이닝은 안광일 등(2000)이 처음으로 제안한 연관 규칙 마이닝 알고리즘이다. 이 연관 규칙 마이닝 알고리즘은 Apriori 알고리즘과는 달리 수량에 대한 정보를 사용한다. 알고리즘의 내용은 다음과 같다. 먼저 발생 수량이 일정 수량 이상인 항목들을 Affinitive 항목 집합으로 선택한다. 선택된 affinitive 항목 집합을 결합하여 2 개의 항목을 가지는 2-항목 집합을 생성한 후, 각 트랜잭션에서 가능한 2-항목집합들의 Affinity coefficient를 구하여 전체 트랜잭션에서 각 항목집합들의 Affinity coefficient의 합을 구한 결과로 Affinity ratio를 계산한다. 이 때, 사용자가 설정하는 파라미터인 최소 Affinity ratio 이상인 2-항목집합들을 Affinitive 항목 집합으로 선택한다. 같은 방식으로 상위 항목 집합에서 Affinitive 항목 집합을 구한다. 생성한 Affinitive 항목 집합들로 생성된 연관 규칙 중 사용자가 설정한 파라미터인 최소 Mixture 이상의 Mixture 값을 가지는 연관 규칙들을 유의미한 연관 규칙으로 선택한다. 연관 규칙을 생성하는데 사용되는 지표인 Affinity coefficient, Affinity ratio와 Mixture는 수식 (4), (5), (6) 과 같이 정의한다.

$$Affico = \min\left(\frac{n_{1,2..k}}{n_1}, \frac{n_{1,2..k}}{n_2}, \dots, \frac{n_{1,2..k}}{n_k}\right) \times (n_1 + n_2 + \dots + n_k) \quad (4)$$

항목집합 $\{1, 2, \dots, k\}$ 의 Affinity coefficient는 수식 (4)로 계산할 수 있다. n_k 는 트랜잭션에서 발생하는 k 항목의 개수를 의미하며, $n_{1,2,..k}$ 는 트랜잭션에서 만들 수 있는 항목들의 순서쌍의 개수이다. 또한 항목집합이 1개의 항목을 가지는 집합인 경우 Affinity coefficient의 값은 데이터 중 해당 항목의 총 발생 수량이다.

$$Affinity\ ratio = \frac{Affico_{1,2..k}}{Affico_{1,1} + Affico_{2,2} + \dots + Affico_{k,k}} \quad (5)$$

상기 항목집합의 Affinity ratio는 수식 (5)로 계산할 수 있으며, 이 값은 항목 간의 연관성 정도가 항목 내의 연관성 대비 어느 정도로 강한지를 판단할 수 있는 지표이다(안광일 등, 2000).

$$mixture = \frac{Affico_{x,y}}{Affico_x} \quad (6)$$

연관 규칙 X→Y의 Mixture는 수식 (6)과 같이 정의되며 이 값은 해당 규칙이 유의한 정도를 나타낸다.

본 연구에서는 안광일 등(2000)이 처음으로 제안한 Affinity measure를 사용한 연관 규칙 마이닝을 입출 빈도 기반 보관 방식에 적용한 제품 저장 위치 할당 방법론으로 적용하여 3.4절의 실험(3)을 진행하였다.

3. 본론

3.1 실험 구성

세 가지 위치 할당 방법들을 3월 주문기록, 4월 주문기록, 5월 주문기록에 대해 실험하였다. 본 연구에서 실험한 제품 위치 할당 방식은 1) 입출 빈도 기반 보관 방식(Turnover-based assignment), 2) Apriori 알고리즘을 사용한 연관 규칙 마이닝을 입출 빈도 기반 보관 방식에 접목한 방식, 3) Affinity measure를 사용한 연관 규칙 마이닝을 입출 빈도 기반 보관 방식을 혼합한 방식 세 가지이다. 첫 번째 방식은 보편적으로 널리 쓰이는 전통적 저장 위치 할당 방법론인 입출 빈도 기반 보관 방식이고, 데이터마이닝 기법 중 하나인 연관 규칙 마이닝을 입출 빈도 기반 보관 방식에 접목한 두 번째, 세 번째 방식과 성능을 비교하여 본 연구에서 제안하는 방식의 성능을 검증하고자 함께 제시하였다. Apriori 알고리즘을 이용하는 연관 규칙 마이닝은 현재까지 사용되는 연관 규칙 마이닝 방법 중 비교적 간단하면서도 좋은 성능을 보이는 보편적인 연관 규칙 마이닝 방법이며, Affinity measure를 이용하는 Affinity 알고리즘은 수량 정보를 고려하여 연관 규칙을 탐색하는 방법이다.

각 위치 할당 방식을 앞서 언급한 UCI에서 제공하고 있는 영국의 온라인 리테일 회사의 2010년 1월부터 2011년 9월까지의 고객 주문 데이터 중에서 3월, 4월, 5월의 주문 기록에 대해 실험하여 월별 평균 오더 피킹 거리(order picking distance)와 월별 오더 피킹 거리 총합을 도출하였다. 최종적으로는, 월별 평균 오더 피킹 거리의 평균과 함께, 월별 오더 피킹 거리의 총합을 각 월의 주문 개수를 가중치로 한 가중평균을 실험 결과로 제시하여 각 위치 할당 방식의 성능을 비교 및 평가하였다. 추가로, 연관 규칙 마이닝을 입출 빈도 기반 보관 방식에 접목하기 위해 주문 빈도가 높은 특정 개수의 제품을 출입구에 가까운 위치에 먼저 할당한 뒤, 연관 규칙에 따라 나머지 제품들을 할당하는 방식을 채택하였는데, 이 때 먼저 할당하는 제품의 개수를 20, 40, 60, 80개로 실험해보고 그 변화에 따라 오더 피킹 거리가 어떻게 변화하는지를 측정하여, 우리가 제안하는 새로운 제품 위치 할당 방법론을 실제 창고 관리에서 적용하는데 필요한 가이드라인과 인사이트를 제공하고자 한다.

3.2 실험 환경

실험 환경은 다음과 같다. Python 3.6.9의 Pandas 라이브러리, Mlxtend 라이브러리 등을 통해 연관 규칙 마이닝을 구현하였고, 이 외에는 마이크로소프트 Excel을 사용하였다. 하드웨어는 Intel® Xeon® Silver 4210R, DDR4 RAM 128.0GB 사양의 하드웨어를 사용하였다.

3.3 데이터 전처리 및 구축

1) 전처리

원본 데이터는 2010년 1월부터 2011년 9월까지의 기간 주문기록이며, 본 연구에서는 2010년 3월, 4월, 5월의 주문 기록에 대하여 실험하였다. 데이터의 전처리 과정은 다음과 같다. 먼저 폐기, 출고 실패 등으로 기록되어있는 주문 기록은 모두 삭제하였다. 그리고 주문량을 의미하는 Quantity의 값이 0 이하인 주문은 취소 주문으로 간주하고, 전체 주문 기록에서 취소된 주문을 제거하기 위해 취소 시점 이전의 주문 기록에서 같은 고객이 같은 항목을 주문한 건을 탐색하여 대조해 취소된 수량을 차감하는 과정을 거쳤다. 이때, 취소 건과 동일한 고객이 주문한 기록이 존재하지 않으면 취소 주문만 삭제하였다. 또한 주문량이 지나치게 높은 경우는 주문에 실수가 있는 경우가 포함될 수 있다고 판단하여 주문량이 $Q1(1사분위) - 1.5 * IQR(interquartile\ range)$ 이하와 $Q3(3사분위) + 1.5 * IQR$ 이상인 주문 기록은 모두 삭제하였다.

2) 월별 주문 기록 데이터 구축

원본 데이터의 형태는 각 행이 하나의 제품에 대한 주문이고, 열은 주문 번호를 의미하는 'InvoiceNo', 제품의 번호를 의미하는 'StockCode', 제품의 이름을 의미하는 'Description', 주문 수량을 의미하는 'Quantity', 주문 시각을 의미하는 'InvoiceDate', 제품 단가를 의미하는 'UnitPrice', 주문한 고객의 ID인 'CustomerID', 마지막으로 주문된 국가를 의미하는 'Country' 열로 구성되어 있다. 이를 연관 규칙 마이닝을 적용할 수 있는 형태로 바꾸기 위해 'InvoiceNo' 열의 값을 행으로 설정, 'StockCode' 열의 값을 열로 설정하여 해당 주문 번호와 제품 번호에 해당하는 주문량을 각각의 셀에 대입하였다. 또한 본 연구에서 설정한 창고 레이아웃은 최대 120개의 제품을 저장할 수 있으므로, 월별 주문 기록에서 주문 빈도가 가장 높은 120개의 품목을 선정하여 실험에 사용하였다. 결과적으로 생성된 주문 기록 데이터 형태는 Table 1과 같다. 또한 Apriori 알고리즘을 사용한 연관 규칙 마이닝을 위해 각 셀의 값을 0 이상인 셀은 1로, 주문량이 0인 셀 (해당 제품을 주문하지 않은 경우)은 0으로 one-hot coding하여 Table 2와 같은 이진 주문 기록의 데이터를 생성하였다.

Table 1. Transaction record of March

StockCode InvoiceNo	22423	22720	85123A	...	22385	22982	23182
545220	0	0	0	...	0	0	0
545221	0	0	0	...	0	0	0
545222	0	24	0	...	0	0	0
...
548547	0	0	0	...	0	0	0
548548	1	0	0	...	0	0	0
548549	0	0	0	...	0	0	0

Table 2. Binary transaction record of March

StockCode InvoiceNo	22423	22720	85123A	...	22385	22982	23182
545220	0	0	0	...	0	0	0
545221	0	0	0	...	0	0	0
545222	0	1	0	...	0	0	0
...
548547	0	0	0	...	0	0	0
548548	1	0	0	...	0	0	0
548549	0	0	0	...	0	0	0

Table 3. Statistics of transaction records

Attributes Month	Number of orders	Average items per order	Minimum number of items per order	Maximum number of items per order	Average order frequency
March	1360	7.3998	1	51	74.5667
April	1151	7.2664	1	53	62.8250
May	1325	7.5758	1	61	84.6500

3월, 4월, 5월 각각 주문 기록 데이터의 주문 건수(Number of orders), 하나의 주문 당 제품 가짓수의 평균값(Average number of items per order), 하나의 주문 당 제품 가짓수의 최솟값(Minimum number of items per order), 하나의 주문 당 제품 가짓수의 최댓값(Maximum of number items per order), 제품 당 주문 빈도의 평균값(Average of order frequency)은 Table 3의 내용과 같다.

3) 창고 layout

창고 layout은 Figure 1과 같은 전형적인 사각형 형태로서 총 6개의 통로(Aisle)와 하나의 통로 당 양쪽에 10개, 총 20개의 저장 위치를 가지도록 구성하였다. 결과적으로 12열의 위치가 존재하게 되며, 좌측부터 1번에서 12번으로 번호를 매겼다. 실험의 연산 시간의 한계에 따라 비교적 간단한 창고 레이아웃을 설정하기 위해, 입구와 출구가 대칭적으로 존재하는 작은 규모의 레이아웃을 설정하였다. 피킹 시에 작업자는 Point I(입구)에서 출발해서 피킹을 시작하여 피킹을 끝내면 Point O(출구)로 나가게 된다. 따라서 피킹 경로가 가장 짧은 통로는 그림에서 가장 진하게 색칠된 구역의 통로이고, 그다음으로 피킹 경로가 짧은 통로는 중간 밝기로 색칠된 구역의 통로, 가장 피킹 경로가 긴 통로는 색칠되지 않은 저장 위치가 설정된 구역의 통로이다. 본 연구에서는 각 Lane의 높이와 깊이가 모두 1인 2차원적 창고 레이아웃을 선택하였다. 각 셀에는 한 가지의 제품만 저장하며, 저장하는 제품의 개수는 고려하지 않는다.

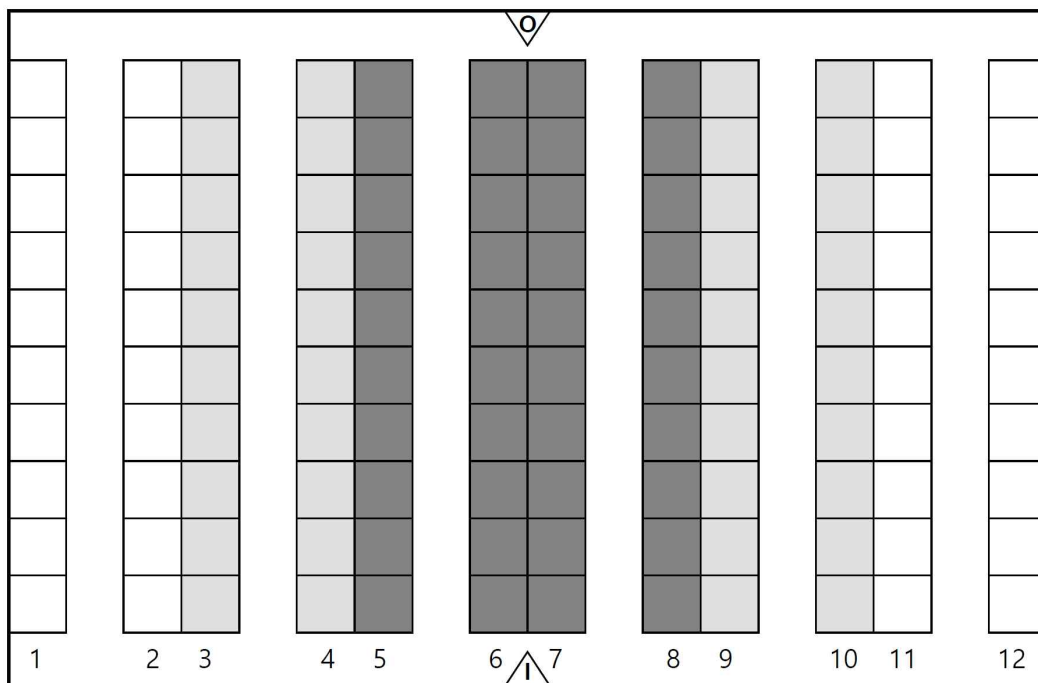


Figure 1. Warehouse layout

3.4 제품 저장 위치 할당 알고리즘

1) 입출 빈도 기반 보관 방식(Turnover-based assignment)

입출 빈도 기반 제품 위치 할당 방법론은 주문 빈도에 따라 위치를 할당하는 방식이다. 이 방식은 입출고 빈도가 높은 제품을 출입구에 가까운 통로에 배치함으로써 이동 시간을 최소화한다(장석화, 2007). 이 실험에서는 총 120가지의 제품을 주문 빈도에 따라 출입구가 위치한 가운데를 기준으로 대칭적으로 할당하였다. 구체적으로, 주문 빈도순 상위 10개의 제품을 입구와 출구에 가장 가까운 통로의 6번 위치에 할당하고, 그 다음 10개를 7번 위치에, 그 다음으로는 5번, 8번, 4번, 9번, 3번, 10번, 2번, 11번, 1번, 12번 위치 순서로 위치를 할당하였다. 그리고 통로 내에서는 입구에 가까운 위치부터 차례대로 채워나가는 방식을 선택하였다.

2) Apriori-연관 규칙 마이닝을 활용한 재고 위치 할당 방식

Apriori-연관 규칙 마이닝을 활용한 재고 위치 할당 방식은 입출 빈도 기반 보관 방식에 연관 규칙 마이닝을 접목한 방식으로, 연관 규칙 마이닝 방법으로는 Apriori 알고리즘을 적용한 방법이다. Apriori 알고리즘을 활용한 연관 규칙 마이닝은 2절에서 설명한 바와 같이 최소 지지도, 최소 신뢰도, 최소 향상도를 조절하여 유의미한 연관 규칙을 탐색하는 데이터마이닝 기법이다.

이 실험에서는 python의 mlxtend 라이브러리의 Apriori 함수와 Association_rules 함수를 이용해 연관 규칙 마이닝을 구현하였다. 적절한 파라미터 값을 찾기 위해 여러 조합의 최소 지지도, 최소 신뢰도, 최소 향상도를 실험해본 뒤, 적정 수준의 rule 개수가 도출되는 파라미터의 조합을 선택하였다. 빈발 항목 집합의 생성에는 최소 지지도(Minimum support) 0.005를 적용하였고, 연관 규칙 생성에는 최소 신뢰도(Minimum confidence) 0.23과 최소 향상도(Minimum lift) 2를 파라미터로 적용하였다. 또한, 실험의 일관성을 위해 3월, 4월, 5월 주문 데이터를 대상으로 한 실험에 전부 동일한 값을 적용하였다. Table 4는 사용된 파라미터의 값과 도출된 rule의 개수(Number of rules)를 정리한 표이다. Table 5에서는 3월의 주문 기록 데이터로 생성된 연관 규칙의 일부를 제시하였다.

Table 4. Parameters used for Apriori algorithm

Parameters Month	Minimum support	Minimum confidence	Minimum lift	Number of rules
March	0.005	0.23	2	1150
April	0.005	0.23	2	1029
May	0.005	0.23	2	1191

Table 5. Details for Association rules using Apriori algorithm (March)

Association rules	Antecedents	Consequents	Support	Confidence	Lift
1	22722	22720	0.1475	0.5507	3.7330
2	23182	22379	0.0092	0.2340	4.9859
3	22697	22699	0.0771	0.7500	10.4041
4	22699	22697	0.0721	0.8023	10.4041
5	85099B	22386	0.0520	0.4336	5.6222

Algorithm 1 How to assign items at the shelf

```
1:  $i = 6, p = 7/2, q = 7/2$ 
2: while  $p \geq 10$  and  $q \geq 10$  do:
3:   if  $i \leq 6$  then:
4:     Assign 10 items at shelf  $i$ ;
5:      $p = p - 10$ ;
6:      $i = i + 2(6 - i) + 1$ ;
7:   end if
8:   if  $i \geq 7$  then
9:     Assign 10 items at shelf  $i$ ;
10:     $q = q - 10$ ;
11:     $i = i - (2(7 - i) + 2)$ ;
12:  end if
13: end while
14: for item  $j$  in shelf do:
15:   for item  $k$  in items associated with item  $j$  do:
16:    if  $i \leq 6$  then:
17:      Assign item  $k$  at shelf  $(i - 1)$ ;
18:      if items in shelf  $(i - 1) > 10$  then:
19:        Assign the left items at the shelf  $(i - 2)$ ;
20:      end if
21:    end if
22:    if  $i \geq 6$  then:
23:      Assign item  $k$  at shelf  $(i + 1)$ ;
24:      if items in shelf  $(i + 1) > 10$  then:
25:        Assign the left items at the shelf  $(i + 2)$ ;
26:      end if
27:    end if
28:  end for
29: end for
30: for item  $j$  not assigned do:
31:   for item  $i = 6, 7, 5, 8, 4, 9, 3, 10, 2, 11, 1, 12$  do:
32:    for  $l = 1; l + 1; l \leq 10$  do:
```

Figure 2. Storage assignment algorithm

연관 규칙을 활용한 제품 위치 할당의 알고리즘은 다음과 같다. 먼저 주문 빈도가 가장 높은 20개 또는 40개, 60개, 80개의 제품을 출입구에 가까운 위치들에 먼저 할당한다. 예를 들어, 80개를 할당하는 경우에는 주문 빈도 순서대로 10개씩 6번, 7번, 5번, 8번, 4번, 9번, 3번, 10번 위치에 할당하고, 20개를 할당하는 경우에는 10개씩 6번과 7번 위치에 순서대로 제품을 할당한다. 그 후에 생성된 연관 규칙에 따라, 출입구를 기준으로 왼쪽에 있는 위치들에 할당된 전체 제품들과 연관된 제품들은 왼쪽의 남은 위치에 차례대로 할당한다. 이 과정을 연관 규칙을 모두 이용할 때까지 반복한다. 출입구 기준 오른쪽 위치에 할당된 제품들에 대해서도 같은 방식으로 제품을 할당한다. 이 때 왼쪽에 공간에 부족할 시에는 12번 위치부터 채우고, 반대로 오른쪽에 공간이 부족하면 1번 위치부터 채운다. 모든 연관 규칙을 이용한 뒤에도 할당되지 못한 제품들은 주문 빈도가 높은 제품들 순서대로 출입구에 가까운 위치부터 대칭적으로 할당한다. 연관 규칙을 활용한 위치 할당 방식의 알고리즘을 Figure 2에 Pseudo-code 형태로 제시하였다. 이 알고리즘은 Affinity-연관 규칙 마이닝을 활용한 제품 위치 할당 방식에도 동일하게 적용한다.

3) Affinity-연관 규칙 마이닝을 활용한 제품 위치 할당 방식

Affinity-연관 규칙 마이닝을 활용한 제품 위치 할당 방식은 Apriori-연관 규칙 마이닝을 활용한 제품 위치 할당 방식과 동일하게 입출 빈도 기반 보관 방식에 연관 규칙 마이닝을 접목한 방식이지만, 연관 규칙을 탐색할 때 안광일 등(2000)의 연구에서 새롭게 제시한 Affinity measure를 사용한다는 점에서 차이점이 있다. 안광일 등(2000)의 연구에서 제시한 연관 규칙 마이닝에는 최소 수량, 최소 Affinity ratio, 최소 Mixture의 파라미터가 필요하다. 본 연구의 실험에서는 (2)의 Apriori-연관 규칙 마이닝을 활용한 제품 위치 할당 방식과 같은 제품들을 대상으로 실험하기 위해 최소 수량을 0으로 설정하여 선정한 120개의 제품 중 탈락하는 제품들이 없도록 하였다. 또한 연관 규칙의 조건 집합과 결과 집합이 각각 하나의 항목만을 가지도록 했으므로, 2개 이상의 항목을 가지는 항목 집합을 생성할 때 필요한 파라미터인 최소 Affinity ratio는 설정하지 않았다. 유의미한 연관 규칙의 선정을 위해 설정하는 파라미터인 최소 Mixture는 (2)의 Apriori-연관 규칙 마이닝을 활용한 제품 위치 할당 방식의 3월 주문 기록 데이터에 대한 실험에서 도출된 rule의 개수와 가능한 비슷한 개수의 Rule이 도출되는 값으로 설정하였다. 이렇게 설정한 최소 Mixture(Minimum mixture) 값을 마찬가지로 실험의 일관성을 위해 3월, 4월, 5월 주문 데이터를 대상으로 한 실험에 전부 동일한 값으로 적용하였다. 사용된 파라미터의 값과 도출된 Rule의 개수(Number of rules)는 Table 6과 같다. Table 7은 3월의 주문 기록 데이터로 생성된 연관 규칙의 예시이다. 연관 규칙을 활용한 위치 할당 방식의 알고리즘은 (2)에서 설명한 내용과 동일하다.

Table 6. Data using minimum mixture

Parameters Month	Minimum mixture	Number of rules
March	0.245	1144
April	0.245	1160
May	0.245	1271

Table 7. Details for Association rules using Affinity algorithm (March)

Association rules	Antecedents	Consequents	Mixture
1	22423	47566	0.2694
2	22423	21212	0.2538
3	22720	22993	0.3168
4	22982	22979	0.8545
5	23182	84991	0.3840

3.5 오더 피킹 정책(Order picking policy) 및 라우팅 정책(Routing policy)

본 연구에서는 간단한 오더 피킹 알고리즘을 위해 오더 피킹 정책은 하나의 주문 만을 출하하는 개별처리(Single order picking) 방식을 선택했다. 라우팅 정책은 S-shape 라우팅 정책과 Return 라우팅 정책을 혼합한 Combined 라우팅 정책을 적용했다. 구체적인 내용은 다음과 같다. Point I(입구)에서 출발한 작업자는 Pick list에 포함된 제품이 위치한 통로(Aisle) 중 가장 왼쪽에 있는 통로부터 방문한다. 첫 번째 제품을 처리한 뒤, 그 다음의 제품들은 가장 거리가 짧은 이동 경로를 선택하여 움직인다고 가정한다. Pick list에 포함된 모든 제품을 처리한 뒤에는 마지막 제품의 위치에서 출구로 이동한다. Figure 3은 표시된 제품 4가지가 하나의 Pick list를 구성하고 있을 때 오더 피킹 경로의 예시이다.

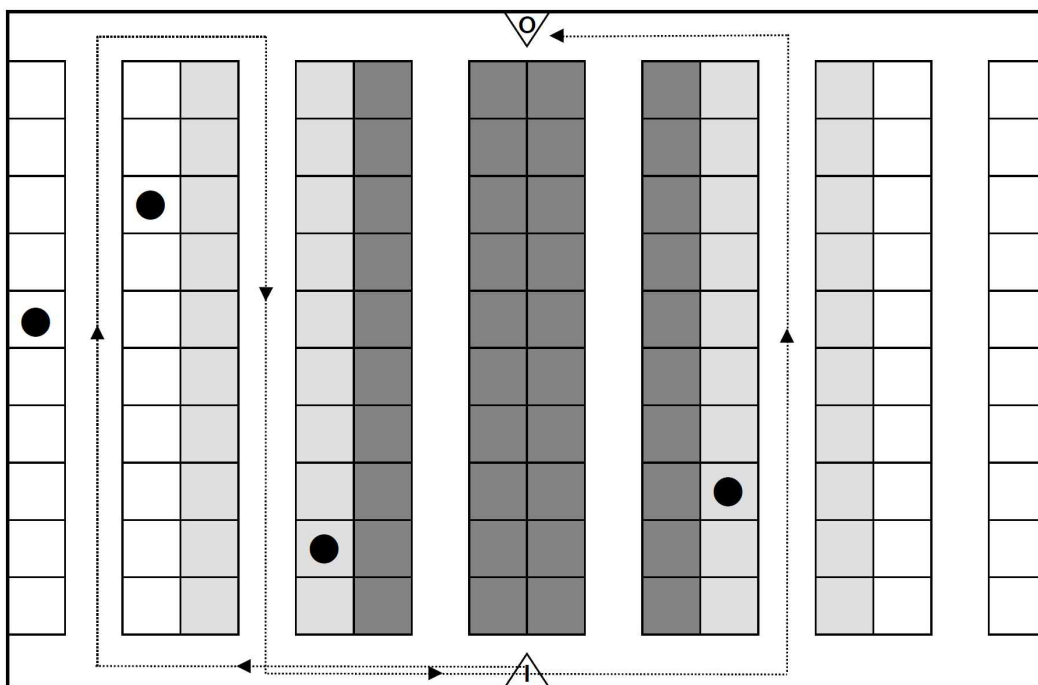


Figure 3. Example of order picking path

3.6 실험 결과 및 분석

본 연구에서는 데이터마이닝 기법 중 연관 규칙 마이닝을 제품 위치 할당 방법론에 적용하여 Order picking distance의 변화를 확인하고, 고전적 제품 위치 할당 방법론인 입출 빈도 기반 보관 방식에서의 Order picking distance와 비교하는 실험을 진행하였다. 실험 데이터는 영국의 e-commerce 회사의 2010년 1월부터 2011년 9월까지의 주문 기록 데이터 중 2010년 3월, 2010년 4월, 2010년 5월의 주문 기록 데이터로, 총 세 가지 실험 데이터를 이용해서 실험하였다. Figure 4의 그래프 중 'Weighted average (Apriori)'는 Apriori-연관 규칙 마이닝을 활용한 제품 위치 할당 방식을 3월, 4월, 5월의 주문 기록 데이터에 대해 실험한 결과로, 월별 오더 피킹 거리의 총합을 주문 개수를 가중치로 가중 평균한 값이다. Figure 3의 그래프 중 'Weighted average (Turnover-based assignment)'는 입출 빈도 기반 보관 방식을 적용했을 때의 월별 오더 피킹 거리의 합을 같은 방법으로 도출한 결과이다. 가로 축의 숫자는 입출 빈도 기반 보관 방식과 연관 규칙 마이닝을 결합할 때, 입출 빈도 기반 보관 방식으로 먼저 창고에 할당하는 제품의 수를 의미한다.

Figure 5의 그래프는 Affinity-연관 규칙 마이닝을 활용한 제품 위치 할당 방식에서의 오더 피킹 거리의 합을 마찬가지로 도출하여 나타낸 결과이다. 주문 당 오더 피킹 거리의 평균값에 대해서도 같은 방법으로 도출하여 Figure 6와 Figure 7으로 함께 제시하였다. 밑의 Table 8은 세 가지 제품 저장 위치 할당 방식의 모든 실험에서의 Order picking distance의 평균치와 모든 실험 데이터에 대한 Order picking distance 합의 가중평균치를 나타내며, Table 9는 모든 실험에서의 주문 당 Order picking distance의 평균치와 모든 실험 데이터에 대한 주문 당 Order picking distance의 평균치를 나타낸다.

1) 제품 위치 할당 방식에 따른 order picking distance의 변화

실험 결과에 따르면, 입출 빈도 기반 보관 방식을 사용했을 때 평균적으로 오더 피킹 거리의 값이 가장 크고, Affinity-연관 규칙 마이닝을 활용한 제품 위치 할당 방식이 두 번째, Apriori-연관 규칙 마이닝을 활용한 제품 위치 할당 방식이 가장 오더 피킹 거리의 값이 작았다. 입출 빈도 기반 보관 방식 대비 Apriori-연관 규칙 마이닝을 활용한 방식에서 한 달 동안의 총 오더 피킹 거리는 0.8652% 더 낮았고, 주문 당 오더 피킹 거리는 1.0641% 더 낮았다. 그리고 Affinity-연관 규칙 마이닝을 활용한 방식에서 한 달 동안의 총 오더 피킹 거리는 입출 빈도 기반 보관 방식에 비해 0.3839% 더 낮았고, 주문 당 오더 피킹 거리는 0.4583% 더 낮았다. 이 결과는 제품 위치 할당 방식에 연관 규칙을 적용하는 것이 이동 거리를 줄이는 데 효과가 있음을 나타낸다. 또한 주문 수량을 고려하여 생성한 연관 규칙을 적용했을 때보다, 수량 정보를 고려하지 않는 Apriori 알고리즘을 적용했을 때 이동 거리가 더 감소했음을 알 수 있다. 하지만 각 방식에서의 이동 거리 감소폭은 모두 2% 미만으로, 성능의 차이가 크게 나타나지 않았다.

Table 8. Weighted average of total order picking distance

Storage assignment Month	Turnover-based assignment	Apriori-ARM	Affinity-ARM
March	109,037.5	107,142.8	109,388.5
April	93,151.5	91,349.3	92,201.3
May	122,643	123,206.3	121,936.0
Weighted average	109,548.3	108,600.5	109,127.8

Table 9. Average of order picking distance per order

Storage assignment Month	Turnover-based assignment	Apriori-ARM	Affinity-ARM
March	91.3977	89.8095	91.6920
April	91.2356	89.4704	90.3049
May	92.5608	92.9859	92.0272
Average	91.7314	90.7553	91.3413

2) 입출 빈도 기반 보관 방식과 연관 규칙 마이닝의 접목

본 연구에서는 연관 규칙 마이닝 적용의 효과와 더불어, 입출 빈도 기반 보관 방식에 연관 규칙 마이닝을 접목할 때, 입출 빈도 기반 보관 방식으로 위치에 먼저 할당하는 제품의 수에 따라 이동 거리가 어떻게 변화하는지를 실험하였다. 결론적으로 Apriori-연관 규칙 마이닝을 적용한 방식에서는 40개의 제품, 즉 총 저장 위치의 1/3을 먼저 할당하였을 때 이동 거리가 가장 작았고, Affinity-연관 규칙 마이닝을 적용한 방식에서는 총 저장 위치의 1/6인 20개의 제품을 먼저 할당하였을 때 이동 거리가 가장 작았다. 실험 데이터가 세 가지로 매우 적어 이 수치가 정확히 가장 좋은 정도라고 결론지을 수는 없으나, 입출 빈도 기반 보관 방식에 따라 할당하는 제품의 개수에 적절한 정도가 존재한다고 해석할 수 있다. 또한 입출 빈도 기반 보관 방식으로 할당한 제품의 개수가 적을수록 연관 규칙에 따라 할당한 저장 위치가 더 많은데, 그래프에서 확인할 수 있듯이 연관 규칙에 따라 할당한 저장 위치의 비율이 높을수록 평균적으로 이동 거리가 더 짧았다.

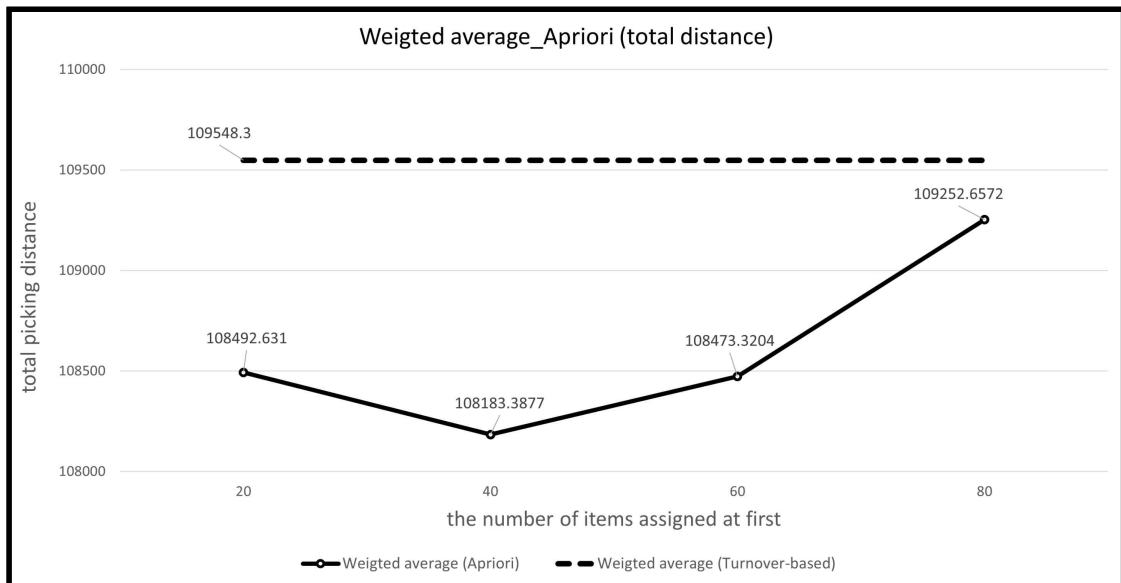


Figure 4. Weighted average of total picking distance using Apriori algorithm

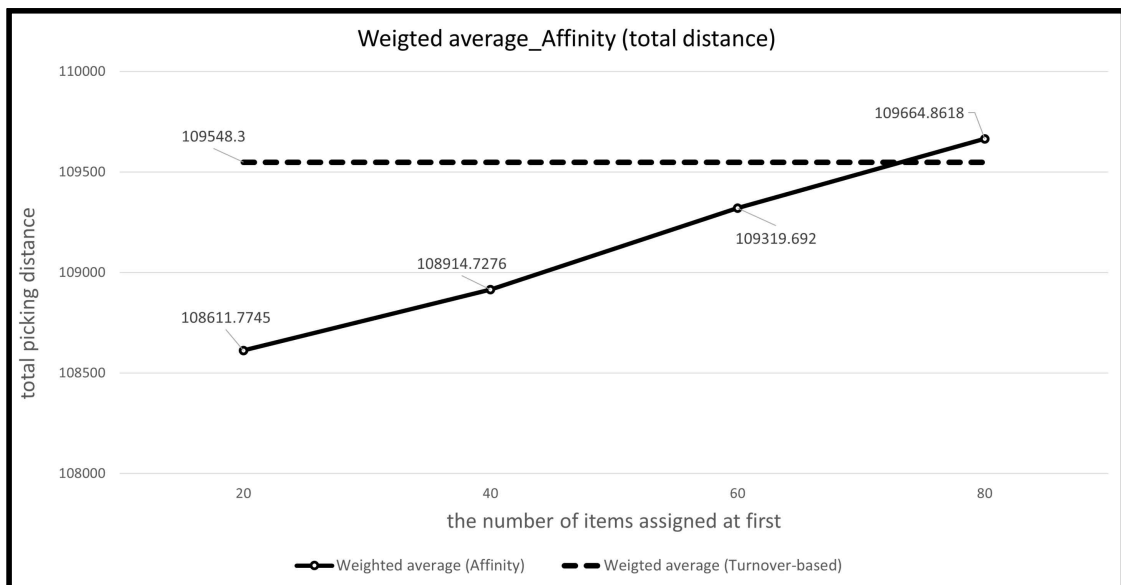


Figure 5. Weighted average of total picking distance using Affinity algorithm

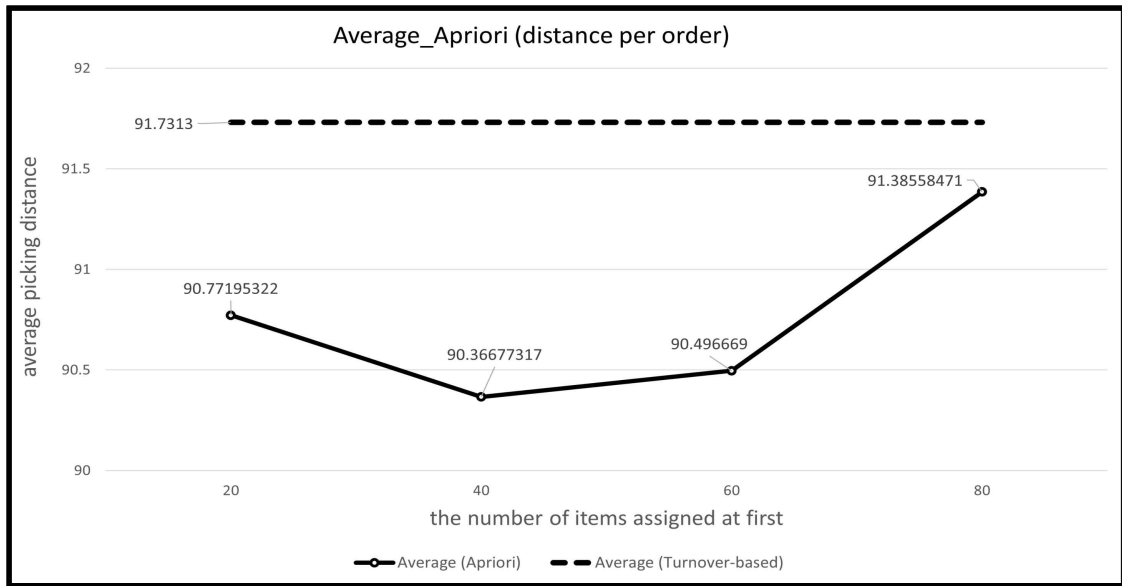


Figure 6. Average of picking distance per order using Apriori algorithm

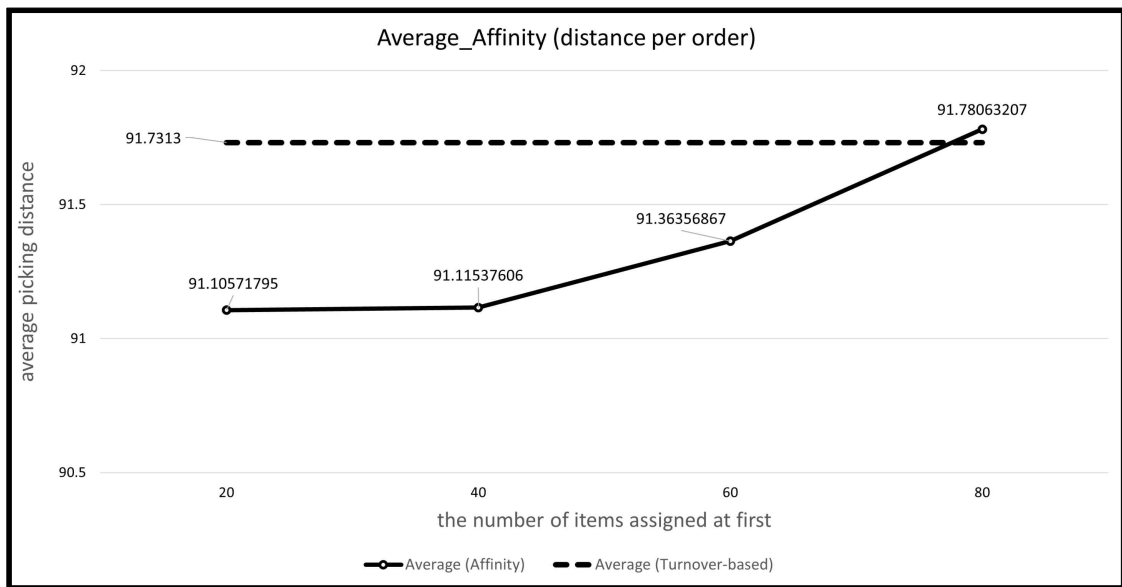


Figure 7. Average of picking distance per order using Affinity algorithm

4. 결론

본 연구에서는 제품 저장 위치 할당 방법론에 데이터마이닝 기법인 연관 규칙 마이닝을 적용한 새로운 제품 저장 위치 할당 알고리즘을 제안한다. 연관 규칙 마이닝에도 항목 집합의 발생 빈도를 기반으로 연관 규칙을 탐색하는 Apriori 알고리즘을 활용한 연관 규칙 마이닝과 발생 빈도와 함께 발생 수량에 대한 정보를 활용하는 Affinity measure를 활용한 연관 규칙 마이닝을 모두 적용하여 다양한 실험을 진행하였다. 실험 결과 및 분석 (1)에서는 제안한 저장 위치 할당 방식의 성능을 평가하기 위해, 같은 실험 데이터에 고전적 저장 위치 할당 방법론인 입출 빈도 기반 보관 방식을 적용한 결과를 비교하여 제시하였다. 그리고 실험 결과 및 분석 (2)에서는 입출 빈도 기반 보관 방식에 연관 규칙 마이닝을 접목할

시에 입출 빈도 기반 보관 방식으로 할당된 저장 위치의 비율에 따른 성능의 차이를 실험 결과로 제시하였다.

3.4절의 (1)에서 밝힌 바와 같이, 이 연구에서 제안하는 저장 위치 할당 방식이 보편적으로 사용되는 입출 빈도 기반 보관 방식보다 더 좋은 성능을 보임을 확인할 수 있었으나, 성능의 차이가 크지 않았다. 이 결과는 사용한 창고 레이아웃이 120개의 매우 적은 저장 위치를 가지고 있어 이동 거리의 차이가 크게 나타날 수 없었기 때문이라고 판단되며, 더 큰 규모의 현실의 창고에서는 이 차이가 증폭되어 유의미한 정도의 차이로 나타날 가능성이 크다. 또한 3.4절의 (2)의 실험 결과에서는, 연관 규칙 마이닝을 적용한 저장 위치 할당 방식 두 가지 모두 입출 빈도 기반 보관 방식으로 할당된 저장 위치의 비율 중 가장 높은 성능을 보이는 특정 개수가 존재함을 확인하였다. 그와 동시에, 연관 규칙에 따라 할당된 저장 위치의 비율이 높을수록 평균적으로 이동 거리가 더 짧음을 알 수 있는데, 이 결과는 연관 규칙 마이닝을 활용하여 제품 저장 위치를 할당하는 방식에 정당성을 부여한다. 따라서 이 연구의 가장 큰 시사점은 창고 내 제품 저장 위치 할당 방식에 연관 규칙 마이닝을 활용하는 것이 충분한 성능의 향상을 이뤄낼 수 있는 가능성이 있으며, 연관 규칙을 적용하는 비율에 있어 가장 높은 성능을 보이는 지점이 있다는 것을 실제 데이터에 대한 실험 결과로서 제시했다는 점이다. 또한 제품 저장 위치를 다루어 왔던 기존 연구는 과거의 수요 패턴을 전제로 하여 주문 빈도에 따라 제품 저장 위치를 결정하는 문제를 주로 다루었으나, 본 연구에서는 이를 확장하여 수요 패턴에서의 제품 간 수요 연관성을 반영한 확장된 규칙을 저장 위치 결정에 활용한 연구라는 점에서 의의가 있다.

이 연구의 한계점은 첫 번째로 데이터의 크기와 실험 데이터의 부족함으로 인해 제안하는 방식의 유의미한 성능 향상을 볼 수 없었다는 점에 있다. 만약에 현실적인 크기의 레이아웃과 개수의 제품들을 포함한 다수의 실험 데이터에 대해 실험한다면 더 유의미한 결과를 제시할 수 있을 것이다. 또한 본 연구의 실험에서는 120개의 제품 주문 빈도 상위 품목들로 선택하였는데, 이것은 이 실험에서 사용한 데이터가 희소성(Sparsity) 문제를 가지고 있었기 때문이다. 이로 인해 이미 연관성이 높은 제품들이 주로 선택되어, 연관 규칙에 따라 저장 위치를 할당하는 것이 큰 성능 향상을 보이지 못했다고 해석된다. 따라서 주문 기록 데이터를 임의적으로 생성하여 실험하거나, 상위, 중위, 하위 주문 빈도를 가지는 제품들을 섞어서 생성한 주문 기록 데이터에 대해 실험하면 입출 빈도 기반 보관 방식과 더욱 큰 성능의 차이를 볼 수 있을 것이다.

두 번째로, 본 연구의 실험에서는 실험의 일관성을 유지하기 위하여 연관 규칙 마이닝을 위한 파라미터를 모든 실험 데이터에 대해 같은 값으로 적용하였는데, 데이터의 특성에 따라 파라미터를 다른 값으로 적용한다면 더 정확한 연관 규칙들을 도출할 수 있다. 같은 맥락으로, 연관 규칙 마이닝의 파라미터 선택에 좋은 성능을 보이는 알고리즘들을 사용한다면 성능의 향상과 더불어, 이 알고리즘을 현실의 창고 관리에서 상용하는데 도움이 될 것이다. 마지막으로, 우리의 실험에서는 연관 규칙을 탐색할 때 조건 집합과 결과 집합이 모두 한 개의 제품만을 포함하도록 연관 규칙을 생성하였는데, 추후의 연구에서 두 개 이상의 항목을 포함하는 항목집합들 간의 연관 규칙들을 탐색한다면, 더욱 다양하고 연관성이 높은 연관 규칙들을 적용한 제품 저장 위치 할당 알고리즘을 사용할 수 있을 것이다.

추후 연구 과제로, 창고의 규모를 확장하여 취급하는 제품의 수를 늘려 검증함과 동시에 lane의 깊이와 높이가 1이상인 3차원적 구조의 창고 레이아웃을 설정하고, 제품의 부피 정보를 고려하여 capacity 제약을 추가하는 방안과 저장 위치에 여러 제품을 저장하는 방안을 포함하는 것을 제시한다. 또한, 개별 처리가 아닌 다수의 주문을 처리하는 오더 배치(Order batch) 정책을 적용함으로써 이 연구의 범용성을 제고할 수 있을 것이라고 생각한다. 마지막으로 이 연구에서는 오더 피킹 작업자가 1명일 때를 가정했는데, 여러 명의 작업자가 동시에 오더 피킹을 하는 상황을 가정하여 제품 위치 할당 알고리즘을 개발하는 것도 연구 범용성 제고에 큰 도움이 될 것이다.

참고문헌

- 안광일, 김성집, & 김재련 (2000), 수량을 고려한 연관규칙 탐색, 대한산업공학회 추계학술대회 논문집, 264-267.
- 장석화(2007), 지정식 보관방식 창고시스템에서 보관위치 할당계획, 한국산업경영시스템학회지, 30(1), 8-14.
- Abisoye, O. A., Boboye, F., and Abisoye, B. O. (2013), Design of a computerized inventory management system for supermarkets, *International Journal of Science and Research*, 2 (2013), pp. 340 ~ 344.
- Adhikary, D., and Roy, S. (2015), Trends in quantitative association rule mining techniques, *Proceedings of IEEE International Conference on Recent Trends in Information System*, pp. 126 ~ 131.
- Agrawal, R., and Srikant, R. (1994), Fast algorithms for mining association rules, In *Proc. 20th int. conf. very large data bases, VLDB (Vol. 1215, pp. 487-499)*.
- Ahmadi, E., Masel, D. T., Metcalf, A. Y., and Schuller, K. (2019), Inventory management of surgical supplies and sterile instruments in hospitals: a literature review, *Health Systems*, 8(2), 134-151.
- Applegate, D. L., Bixby, R. E., Chvátal, V., and Cook, W. J. (2011), *The traveling salesman problem*, Princeton university press.
- Bevilacqua, M., Ciarapica, F. E., and Antomarioni, S. (2019), Lean principles for organizing items in an automated storage and retrieval system: an association rule mining-based approach, *Management and Production Engineering Review*, 10(1), 29-36.
- Brynzér, H., and Johansson, M. I. (1996), Storage location assignment: Using the product structure to reduce order picking times, *International Journal of Production Economics*, 46, 595-603.
- Cano, J. A., Correa-Espinal, A. A., and Gómez-Montoya, R. A. (2017), An evaluation of picking routing policies to improve warehouse efficiency, *International Journal of Industrial Engineering and Management*, 8(4), 229-238.
- Chuang, Y. F., Chia, S. H., and Wong, J. Y. (2014), Enhancing order-picking efficiency through data mining and assignment approaches, *WSEAS Transactions on Business and Economics*, 11(1), 52-64.
- Chuang, Y. F., Lee, H. T., and Lai, Y. C. (2012), Item-associated cluster assignment model on storage allocation problems, *Computers & industrial engineering*, 63(4), 1171-1177.
- Cergibozan, Ç., and Tasan, A. S. (2019), Order batching operations: an overview of classification, solution techniques, and future research, *Journal of Intelligent Manufacturing*, 30(1), 335-349.
- Chen D., Sain S. L., and Guo K. (2012), Data mining for the online retail industry: A case study of RFM model-based customer segmentation using data mining, *Journal of Database Marketing and Customer Strategy Management*, Vol. 19, No. 3, pp. 197-208.
- De Koster, R., Le-Duc, T., and Roodbergen, K. J. (2007), Design and control of warehouse order picking: A literature review, *European journal of operational research*, 182(2), 481-501.
- Ene, S., and Öztürk, N. (2012), Storage location assignment and order picking optimization in the

- automotive industry, *The international journal of advanced manufacturing technology*, 60(5), 787-797.
- Goetschalckx, M., and Donald Ratliff, H. (1988), Order picking in an aisle, *IIE transactions*, 20(1), 53-62.
- Han, J., Pei, J., and Kamber, M. (2011), *Data mining: concepts and techniques*, Elsevier.
- Hausman, W. H., Schwarz, L. B., and Graves, S. C. (1976), Optimal storage assignment in automatic warehousing systems, *Management science*, 22(6), 629-638.
- Kumbhare, T. A., and Chobe, S. V. (2014), An overview of association rule mining algorithms, *International Journal of Computer Science and Information Technologies*, 5(1), 927-930.
- Li, J., Moghaddam, M., and Nof, S. Y. (2016), Dynamic storage assignment with product affinity and ABC classification—a case study, *The International Journal of Advanced Manufacturing Technology*, 84(9), 2179-2194.
- Ming-Huang Chiang, D., Lin, C. P., and Chen, M. C. (2014), Data mining based storage assignment heuristics for travel distance reduction, *Expert Systems*, 31(1), 81-90.
- Pan, J. C. H., and Wu, M. H. (2009), A study of storage assignment problem for an order picking line in a pick-and-pass warehousing system, *Computers & Industrial Engineering*, 57(1), 261-268.
- Petersen, C. G. (1997), An evaluation of order picking routeing policies, *International Journal of Operations & Production Management*.
- Reyes, J., Solano-Charris, E., and Montoya-Torres, J. (2019), The storage location assignment problem: A literature review, *International Journal of Industrial Engineering Computations*, 10(2), 199-224.
- Tompkins, J. A., White, J. A., Bozer, Y. A., and Tanchoco, J. M. A. (2010), *Facilities planning*, John Wiley & Sons.
- Van Gils, T., Braekers, K., Ramaekers, K., Depaire, B., and Caris, A. (2016), Improving order picking efficiency by analyzing combinations of storage, batching, zoning, and routing policies, In *International Conference on Computational Logistics* (pp. 427-442), Springer, Cham.

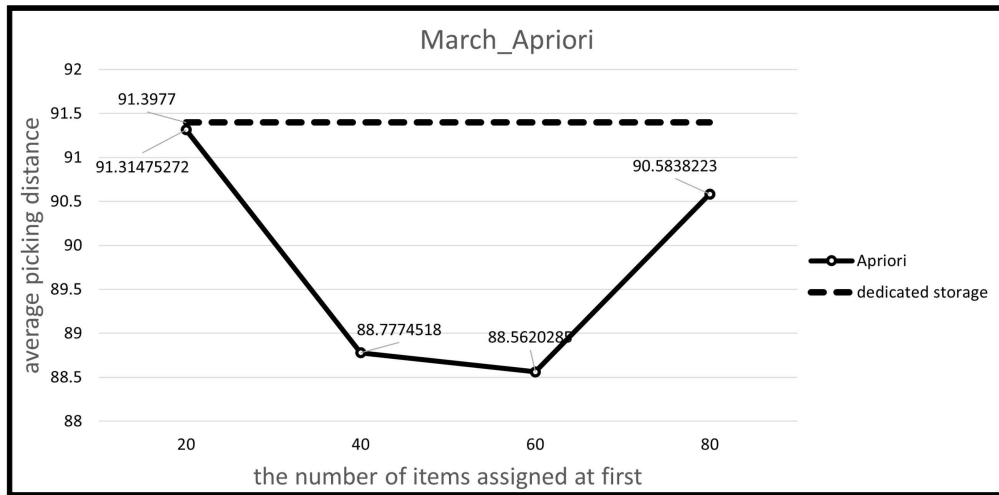
5. 부록(Appendix)

Appendix [a]. Average order picking distance using Apriori - association rule mining

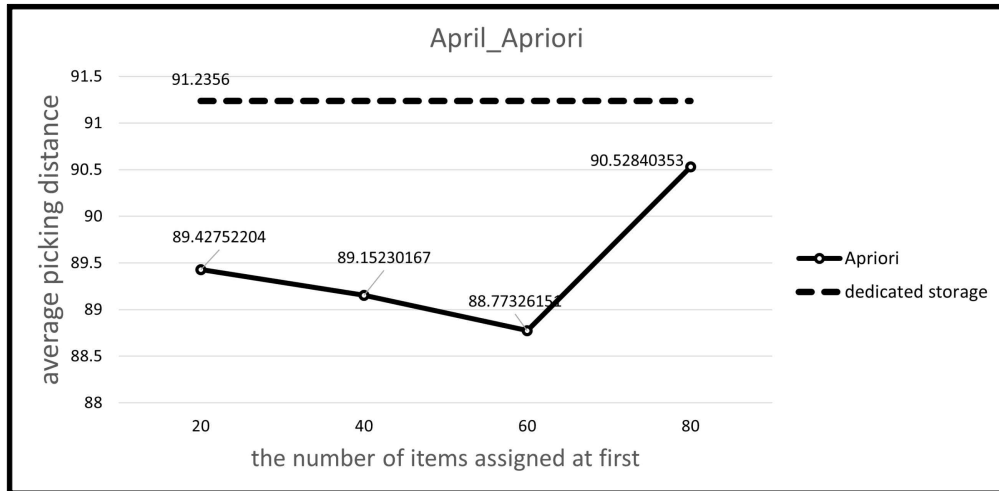
Month \ #items assigned at first	20	40	60	80
March	91.3148	88.7775	88.5620	90.5838
April	89.4275	89.1523	88.7733	90.5284
May	91.5736	93.1706	94.1547	93.0445

Appendix [b]. Average order picking distance using Affinity - association rule mining

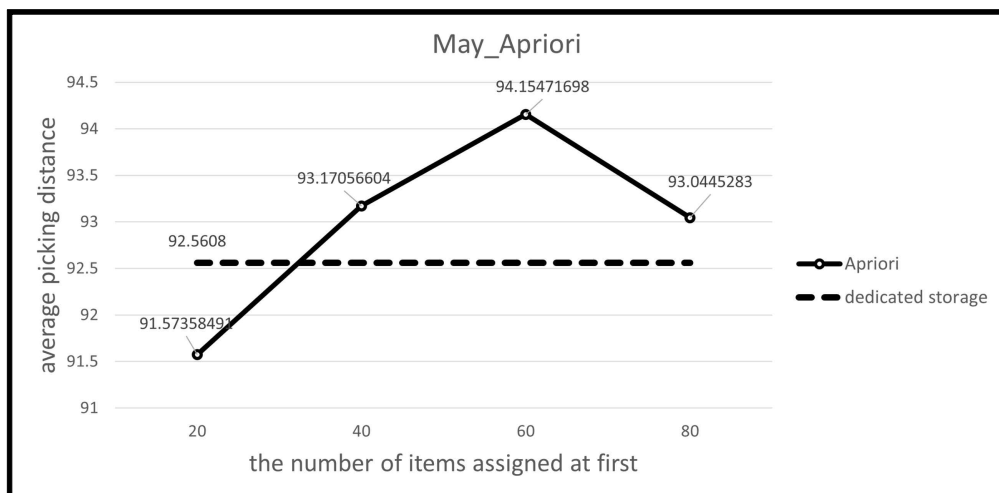
Month \ #items assigned at first	20	40	60	80
March	90.8881	92.2603	91.9233	91.6961
April	91.5000	89.4285	89.4285	90.8820
May	90.9291	91.6574	92.7585	92.7638



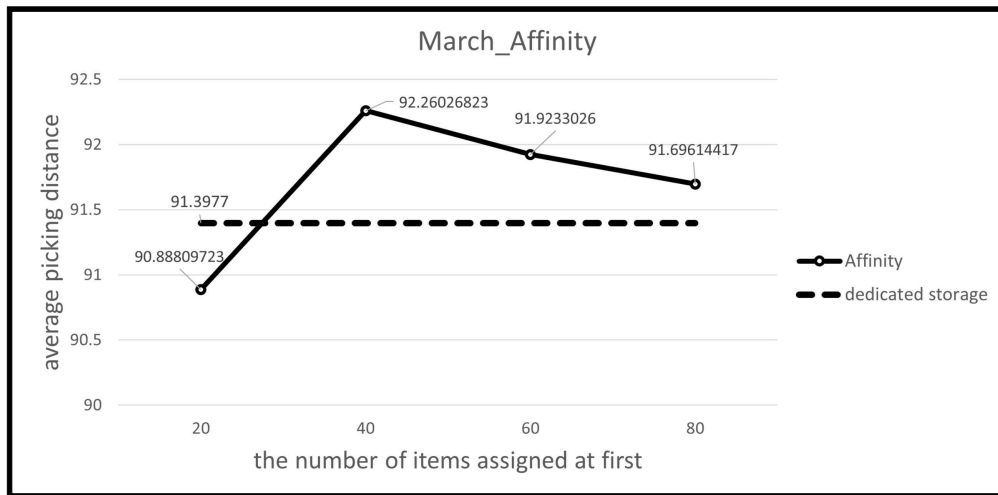
Appendix [c]. Average order picking distance using Apriori-association rule mining (March)



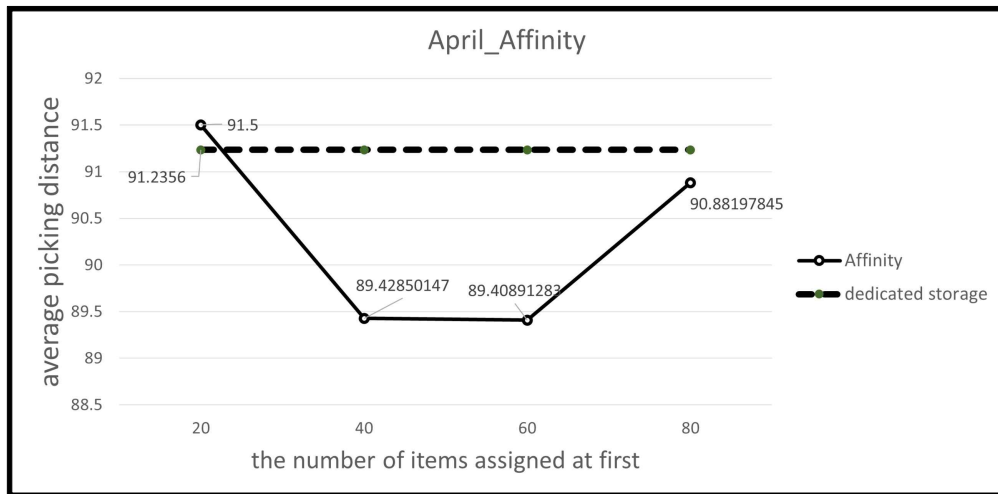
Appendix [d]. Average order picking distance using Apriori-association rule mining (April)



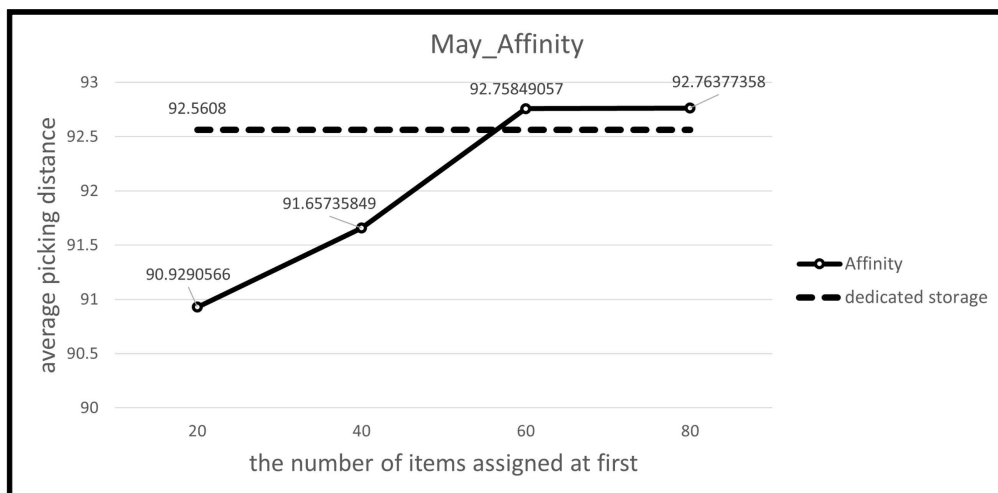
Appendix [e]. Average order picking distance using Apriori-association rule mining (May)



Appendix [f]. Average order picking distance using Affinity-association rule mining (March)



Appendix [g]. Average order picking distance using Affinity-association rule mining (April)



Appendix [f]. Average order picking distance using Affinity-association rule mining (May)