



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2024-0110330
(43) 공개일자 2024년07월15일

(51) 국제특허분류(Int. Cl.)
G06F 16/34 (2019.01) G06F 16/33 (2019.01)
G06F 18/10 (2023.01)
(52) CPC특허분류
G06F 16/34 (2019.01)
G06F 16/3347 (2019.01)
(21) 출원번호 10-2023-0002273
(22) 출원일자 2023년01월06일
심사청구일자 2023년01월06일

(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
청주대학교 산학협력단
충청북도 청주시 청원구 대성로 298 (내덕동)
(72) 발명자
이주현
충청남도 당진시 밤절로 104, 303동 804호
박상성
충청북도 청주시 청원구 공항로 159-12, 526호
이준석
경기도 김포시 유현로 19, 121동 1502호
(74) 대리인
김등용

전체 청구항 수 : 총 10 항

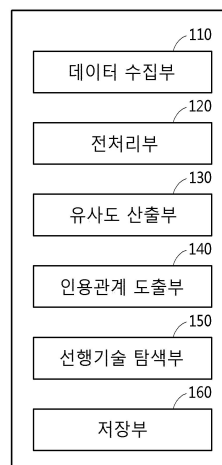
(54) 발명의 명칭 특허정보에 기반한 기술 관계 분석 장치 및 방법

(57) 요약

특허정보에 기반한 기술 관계 분석 장치 및 방법이 개시된다. 이를 위한 선행기술 탐색 장치는 복수의 특허들 각각에 대한 정보를 수집하는 데이터 수집부, 상기 복수의 특허들 각각에 대한 정보를 전처리하는 전처리부, 상기 복수의 특허들 간의 유사도를 산출하는 유사도 산출부, 상기 복수의 특허들 간의 인용관계를 도출하는 인용관계 도출부, 및 상기 복수의 특허들 중에서 유사성이 인정되지만 직접적인 인용관계가 존재하지 않는 특허쌍을 도출하는 선행기술 탐색부를 포함한다.

대표도 - 도3

100



(52) CPC특허분류

G06F 18/10 (2023.01)

G06F 2216/11 (2013.01)

Y10S 707/933 (2013.01)

Y10S 707/937 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호 1345353675

과제번호 2022R1I1A1A01069422

부처명 교육부

과제관리(전문)기관명 한국연구재단

연구사업명 이공학학술연구기반구축

연구과제명 자연어 이해를 활용한 비즈니스 서비스를 위한 오픈 플랫폼 연동형 양자 기반
시스템 개발

기 여 율 1/1

과제수행기관명 고려대학교

연구기간 2022.06.01 ~ 2023.02.28

명세서

청구범위

청구항 1

복수의 특허들 각각에 대한 정보를 수집하는 데이터 수집부;
상기 복수의 특허들 각각에 대한 정보를 전처리하는 전처리부;
상기 복수의 특허들 간의 유사도를 산출하는 유사도 산출부;
상기 복수의 특허들 간의 인용관계를 도출하는 인용관계 도출부; 및
상기 복수의 특허들 중에서 유사성이 인정되지만 직접적인 인용관계가 존재하지 않는 특허쌍을 도출하는 선행기술 탐색부를 포함하는 선행기술 탐색 장치.

청구항 2

제1항에 있어서,
상기 복수의 특허들 각각에 대한 정보는 상기 복수의 특허들 각각의 제목, 요약, 및 인용특허의 식별 정보를 포함하는,
선행기술 탐색 장치.

청구항 3

제2항에 있어서,
상기 전처리부는 사전학습 언어 모델(Pre-Trained Language Model, PTL)을 이용하여 상기 복수의 특허들 각각에 대한 임베딩을 생성하는,
선행기술 탐색 장치.

청구항 4

제3항에 있어서,
상기 유사도 산출부는 상기 임베딩을 기초로 상기 복수의 특허들 간의 유사도를 산출하고, 미리 정해진 임계치를 초과하는 유사도를 갖는 특허 간에는 유사성이 있는 것으로 판단하는,
선행기술 탐색 장치.

청구항 5

제4항에 있어서,
상기 유사도 산출부는 코사인 유사도를 산출하는,
선행기술 탐색 장치.

청구항 6

제5항에 있어서,
상기 인용관계 도출부는 상기 복수의 특허들 간에 인용관계를 나타내는 행렬인 인용관계 행렬을 생성하는,
선행기술 탐색 장치.

청구항 7

제6항에 있어서,

상기 선행기술 탐색부는 상기 복수의 특허들 중에서 유사성이 인정되지만 직접적인 인용관계를 가지지 않는 특허쌍을 도출하는,

선행기술 탐색 장치.

청구항 8

제7항에 있어서,

상기 선행기술 탐색부는 상기 복수의 특허들 중에서 유사성이 인정되는 두 개의 특허를 선택하고, 선택된 두 개의 특허 중 출원 일자가 앞서는 특허를 선행특허로 결정하고, 선택된 두 개의 특허 중 출원 일자가 느린 특허를 특허를 후행특허로 결정하고, 인용 관계로 연결된 상기 두 개의 특허 사이의 인용의 횟수를 산출하는,

선행기술 탐색 장치.

청구항 9

제8항에 있어서,

상기 인용의 횟수는 BFS(Breadth-First Search) 기법을 이용하여 산출되는,

선행기술 탐색 장치.

청구항 10

적어도 프로세서를 포함하는 컴퓨팅 장치에 의해 수행되는 선행기술 탐색 방법에 있어서,

복수의 특허들 각각에 대한 정보를 수집하는 단계;

상기 복수의 특허들 각각에 대한 정보를 전처리하는 단계;

상기 복수의 특허들 간의 유사도를 산출하는 단계;

상기 복수의 특허들 간의 인용관계를 도출하는 단계; 및

상기 복수의 특허들 중에서 유사성이 인정되지만 직접적인 인용관계가 존재하지 않는 특허쌍을 도출하는 단계를 포함하는 선행기술 탐색 방법.

발명의 설명

기술 분야

[0001] 본 발명은 선행기술 탐색을 위한 것으로, 기술 분야에 따라 유사하지만 인용관계가 없는 선행특허를 탐색할 수 있는 장치 및 방법에 관한 것이다.

배경 기술

[0002] 특허는 빠르게 생성되며 축적된 양이 방대하고 다양한 형태의 정보를 포함하는 빅데이터(big data)이다. 특허 빅데이터의 평가는 많은 시간과 비용이 소모된다. 또한, 전문가 기반 선행기술 탐색은 전문가 의견에 편향된 결과가 도출될 수 있다. 이를 개선하기 위해 데이터 기반 특허 분석이 필요하다. 그리고 특허에 존재하는 서지정보와 인용관계를 사용할 필요가 있다.

[0003] 이에 대한 대안으로써 본 발명에서 제안하는 선행기술 탐색 방법을 통해 특허의 서지정보로 특허 간의 유사한 정도를 산출하고 이를 행렬 형태로 구조화한다. 그리고 특허의 특허의 인용관계도 행렬 형태로 구조화한 뒤, 복수이 행렬을 결합하여 유사한 특허들에 대해 서지정보를 고려하여 먼저 출원된 특허를 결정하고, 인용관계를 고려하여 유사하지만 인용관계가 없는 특허 쌍을 탐색하는 것이 가능하다.

선행기술문헌

특허문헌

[0004] (특허문헌 0001) 대한민국 공개특허 제2015-0102477호 (2015.09.07. 공개)

(특허문헌 0002) 대한민국 공개특허 제2017-0098607호 (2017.08.30. 공개)

(특허문헌 0003) 대한민국 공개특허 제2016-0047112호 (2016.05.02. 공개)

(특허문헌 0004) 대한민국 등록특허 제1625124호 (2016.05.27. 공고)

발명의 내용

해결하려는 과제

[0005] 본 발명은 상기한 바와 같이 데이터 기반의 선행기술 탐색을 통해 제반 문제점을 개선하기 위해 제안된 것으로써, 기존에는 선행문헌 조사나 전문가 집단에 의해 진행하였으나, 본 시스템을 통해 시간과 비용을 절감하면서 정확한 선행기술 탐색이 가능하다.

과제의 해결 수단

[0006] 본 발명의 일 실시예에 따른 선행기술 탐색 장치는 복수의 특허들 각각에 대한 정보를 수집하는 데이터 수집부, 상기 복수의 특허들 각각에 대한 정보를 전처리하는 전처리부, 상기 복수의 특허들 간의 유사도를 산출하는 유사도 산출부, 상기 복수의 특허들 간의 인용관계를 도출하는 인용관계 도출부, 및 상기 복수의 특허들 중에서 유사성이 인정되지만 직접적인 인용관계가 존재하지 않는 특허쌍을 도출하는 선행기술 탐색부를 포함한다.

[0007] 또한, 본 발명의 일 실시예에 따른 선행기술 탐색 방법은 적어도 프로세서를 포함하는 컴퓨팅 장치에 의해 수행되고, 복수의 특허들 각각에 대한 정보를 수집하는 단계, 상기 복수의 특허들 각각에 대한 정보를 전처리하는 단계, 상기 복수의 특허들 간의 유사도를 산출하는 단계, 상기 복수의 특허들 간의 인용관계를 도출하는 단계, 및 상기 복수의 특허들 중에서 유사성이 인정되지만 직접적인 인용관계가 존재하지 않는 특허쌍을 도출하는 단계를 포함한다.

발명의 효과

[0008] 본 발명의 특허 간의 유사한 정도와 인용관계를 고려한 선행기술 탐색은 연구개발 전에 유사한 기술의 유무와 권리범위를 침해하고 있는 선행기술의 조기 탐색을 통한 특허 분쟁의 조기 대응이 가능한 장점이 있다.

도면의 간단한 설명

[0009] 본 발명의 상세한 설명에서 인용되는 도면을 보다 충분히 이해하기 위하여 각 도면의 상세한 설명이 제공된다.

도 1은 특허 검색 방법의 분류를 설명하기 위한 도면이다.

도 2는 인용과 유사도 네트워크의 예를 도시하는 도면으로, (a)는 10개의 특허들의 관계를 보여주는 G_c 의 예이고, (b)는 p_1 과 유사한 특허들의 연결(linking) 결과를 도시하고, (c)는 G_c 에 포함되는 두 개의 노드들 사이의 최단 거리(shortest path) 검색하는 과정을 도시한다.

도 3은 본 발명의 일 실시예에 따른 선행기술 탐색 장치의 기능 블록도이다.

도 4는 도 3에 도시된 선행기술 탐색 장치에 의해 수행되는 선행기술 탐색 방법을 설명하기 위한 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0010] 본 명세서에 개시되어 있는 본 발명의 개념에 따른 실시예들에 대해서 특정한 구조적 또는 기능적 설명들은 단지 본 발명의 개념에 따른 실시예들을 설명하기 위한 목적으로 예시된 것으로서, 본 발명의 개념에 따른 실시예들은 다양한 형태들로 실시될 수 있으며 본 명세서에 설명된 실시예들에 한정되지 않는다.

[0011] 본 발명의 개념에 따른 실시예들은 다양한 변경들을 가할 수 있고 여러 가지 형태들을 가질 수 있으므로 실시예들을 도면에 예시하고 본 명세서에서 상세하게 설명하고자 한다. 그러나, 이는 본 발명의 개념에 따른 실시예들을 특정한 개시 형태들에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물, 또는 대체물을 포함한다.

[0012] 제1 또는 제2 등의 용어는 다양한 구성 요소들을 설명하는데 사용될 수 있지만, 상기 구성 요소들은 상기 용어

들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성 요소를 다른 구성 요소로부터 구별하는 목적으로만, 예컨대 본 발명의 개념에 따른 권리 범위로부터 벗어나지 않은 채, 제1 구성 요소는 제2 구성 요소로 명명될 수 있고 유사하게 제2 구성 요소는 제1 구성 요소로도 명명될 수 있다.

[0013] 어떤 구성 요소가 다른 구성 요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성 요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성 요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성 요소가 다른 구성 요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는 중간에 다른 구성 요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성 요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.

[0014] 본 명세서에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로서, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 본 명세서에 기재된 특징, 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0015] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

[0016] 이하, 본 명세서에 첨부된 도면들을 참조하여 본 발명의 실시예들을 상세히 설명한다. 그러나, 특허출원의 범위가 이러한 실시예들에 의해 제한되거나 한정되는 것은 아니다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.

[0017] 도 1은 특허 검색 방법의 분류를 설명하기 위한 도면이다.

[0018] 도 1은 Shalaby and Zadrozny(Shalaby, W., and Zadrozny, W. (2019). Patent retrieval: a literature review. Knowledge and Information Systems, 1-30.)가 제안한 특허 검색에 대한 분류를 도시한다. 특허 검색 방법 중 첫 번째는 키워드(keyword) 기반의 방법이다. 선행 기술은 특허 데이터베이스(database)에서 쿼리(query)를 이용하여 검색할 수 있다. 이 방법에 기반한 연구들의 목적은 쿼리를 확장하거나 축소하여 선행 기술 조사의 효율성을 높이는 것이다.

[0019] 두 번째 방법은 특허의 고유한 특징을 활용하는 것이다. 특허는 출원될 때 심사관에 의해 IPC 코드(international patent classification code)가 할당된다. 그리고 특허는 선행 기술을 인용하며, 다른 특허에 의해 인용되기도 한다. 메타데이터(metadata) 기반의 방법은 특허 검색을 효율적으로 하기 위해 기술을 분류한 IPC 코드나 특허를 형태학적(Morphology)으로 구분한 F-term을 활용한다.

[0020] 마지막 방법은 텍스트 마이닝이나 자연어 처리를 통해 특허의 시멘틱(semantic) 정보를 활용하는 것이다. 일반적으로 이 방법은 문서의 유사성을 측정하여 선행 기술 여부를 판단한다.

[0021] A. KEYWORD-BASED APPROACHES

[0022] 쿼리는 특허 데이터베이스에서 검색할 때 사용되는 키워드의 조합이다. 연구자는 기술에 대한 지식을 기반으로 관련 있는 키워드를 선정하고 그것들을 조합하여 쿼리를 만든다. 너무 적은 키워드를 사용한 쿼리는 매우 넓은 범위의 특허를 검색한다. 이러한 검색결과를 기반한 분석의 결과는 예러가 포함될 수 있다. 반대로 너무 많은 키워드를 사용한 쿼리는 매우 좁은 범위의 특허를 검색한다. 이러한 검색결과를 기반한 분석은 편향된 결과를 야기할 수 있다.

[0023] 선행 기술 조사는 적절한 범위의 쿼리가 필요하다.

[0024] Jones(Jones, S. (1998, November). Graphical query specification and dynamic result previews for a digital library. In Proceedings of the 11th annual ACM symposium on User interface software and technology (pp. 143-151).)는 벤다이어그램(venn diagram)을 통해 쿼리 결과를 표기하는 방법은 제안했다. 이 방법은 검색되는 데이터를 구조화하여 쿼리를 공식화했다. Anick et al.(Anick, P. G., Brennan, J. D.,

Flynn, R. A., Hanssen, D. R., Alvey, B., and Robbins, J. M. (1989, December). A direct manipulation interface for boolean information retrieval via natural language query. In Proceedings of the 13th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 135-150).)와 Ressel-Rose et al.(Russell-Rose, T., Chamberlain, J., and Shokrane, F. (2019, March). A visual approach to query formulation for systematic search. In Proceedings of the 2019 Conference on Human Information Interaction and Retrieval (pp. 379-383).)들은 정확하고(accurate) 반복가능하며(repeatable) 명확한(transparent) 선행 기술 조사를 위해 쿼리 방식의 툴(tool)을 제안했다. Tiwana and Horowitz(Tiwana, S., and Horowitz, E. (2009, November). Findcite: automatically finding prior art patents. In Proceedings of the 2nd international workshop on Patent information retrieval (pp. 37-40).)들은 데이터베이스에서 검색된 특허와 유사한 특허를 찾기 위해 쿼리에 사용된 키워드를 이용하는 방법을 제안했다.

[0025] 쿼리 기반의 선행 기술 조사는 전통적인 방법으로 오래 전부터 연구된 분야이다. 그러나 이 방식은 두 가지 한계점이 있다. 첫 번째, 쿼리 기반의 방식은 키워드에 의존한다. 다시 말하면, 이 방법은 심사관에 따라 수집된 선행기술 대상이 다를 수 있다. 두 번째, 쿼리 작성에 필요한 키워드를 선정하기 위해서는 대상 기술에 대한 이해가 필수적이다. 따라서 심사관은 새로운 기술의 선행 기술을 조사하기 위해 많은 시간을 할애해야 하므로 업무 효율이 낮을 수 있다.

[0026] B. METADATA-BASED APPROACHES

[0027] 특허는 기술의 분야에 따라 화학, 기계, 전기 등으로 분류된다. 이러한 특허 분류 코드는 선행 기술 조사에 활용된다. Harris et al.(Harris, C. G., Arens, R., and Srinivasan, P. (2011). Using classification code hierarchies for patent prior art searches. In Current challenges in patent information retrieval (pp. 287-304). Springer, Berlin, Heidelberg.)은 IPC의 계층(hierarchy) 구조를 활용하여 선행 기술을 조사하는 방법을 제안했다. IPC와 같은 특허 분류 코드를 사용한 선행 기술 조사는 일관된 결과가 도출될 수 있는 장점이 있다. 그러나 이 방식은 새로운 기술 분야에 대한 조사가 어려운 단점이 있다. 그리고 또 다른 한계점은 두 개 이상의 기술이 합쳐진 경우, 특허 분류 코드가 갖는 의미가 모호해질 수 있어 선행 기술의 범위를 결정하기 어렵다는 것이다.

[0028] 특허는 분류 코드 외에도 인용과 같은 다양한 메타데이터(metadata)를 포함한다. 특허의 인용은 법적인 권리와 연관된 정보로 해석될 수 있어 인용에 대한 기술 및 법적 해석을 기반으로 특허의 품질을 평가하고 이를 기반으로 선행 기술을 조사할 수 있다. Couteau(Couteau, O. (2014). Forward searching-A complement to keyword- and class-based patentability searches. World Patent Information, 37, 33-38.)는 유사한 특허를 찾기 위해 forward searching과 citation analysis를 결합한 방법을 제안했다. 그리고 Von Wartburg et al.(Von Wartburg, I., Teichert, T., and Rost, K. (2005). Inventive progress measured by multi-stage patent citation analysis. research Policy, 34(10), 1591-1607.)들은 유사한 특허를 탐색하기 위해 패밀리 특허를 가중치로 사용한 인용 네트워크를 이용했다. 그들은 네트워크를 통해 기술의 cumulative inventive progress를 측정하는 방법을 제안했다.

[0029] 공인용(co-citation)은 서로 다른 선행 기술이 나중에 등록된 특허에 의해 동시에 인용된 것을 의미한다(Small, H. (1973). Co-citation in the scientific literature: A new measure of the relationship between two documents. Journal of the American Society for information Science, 24(4), 265-269.). 기존의 연구는 선행 기술을 조사하기 위해 co-citation의 관계를 점수화하는 방법을 제안했다. 또 다른 연구에서는, co-citation 관계를 통해 기술의 지식-전이(knowledge-transfer)를 추적하는 방식을 제안했다. 그러나 기존의 메타데이터-기반 접근법들은 새로운 기술 분야에 적용하기 어려울 뿐만 아니라, 인용의 흐름을 반영하지 못하는 한계점이 있다. 인용의 흐름은 기술의 발전 동향을 파악할 수 있는 정보이다. 기존의 연구들에서 특허의 전방 또는 후방 특허만을 활용했기 때문에, 개선된 선행 기술 조사 방법은 인용의 흐름을 반영할 필요가 있다.

[0030] C. SEMANTIC-BASED APPROACHES

[0031] 특허는 발명의 명칭, 요약(abstract), 청구항 등으로 구성된 문서이다. 즉, 특허는 자연어 처리 방법을 통해 유사도를 측정할 수 있다. 최근, 자연어 처리 알고리즘이 발전함에 따라 많은 연구들이 선행 연구 조사에 시맨틱(semantic) 정보를 사용하고 있다. No et al.과 Nakamura et al.들은 co-citation 정보와 co-word의 빈도를 병합한 매트릭스 기반 방식을 제안했다. 이들은 특허의 유사성을 판단할 때, 메타데이터(metadata)나 시맨틱(semantic) 정보 중 하나만을 사용하는 것의 문제점을 언급했다. 그러나 이들의 방법은 새로운 기술에 사용되는

co-word에 적용하기 어렵다.

[0032] Sanguri et al.들은 딕셔너리(dictionary) 기반의 시멘틱-기반 접근법(semantic-based approaches)의 문제점을 개선하기 위해 잠재 의미 분석(latent semantic analysis)을 활용한 방법을 제안했다. 그들은 텍스트의 잠재(latent) 정보를 추출하고 co-citation 관계와 함께 문서의 유사도를 계산했다. Cammarano et al.들은 인용 관계와 문서의 잠재 의미(latent semantic)를 병합한 상관관계를 제안했다.

[0033] 기존의 연구들을 검토한 결과, 선행 기술 조사를 위한 알고리즘은 키워드(keyword), 메타데이터(metadata) 그리고 시멘틱(semantic) 정보를 각각 사용해왔다. 자연어 처리 방법이 발전함에 따라, 키워드-기반 접근법(keyword-based approach)보다 시멘틱-기반 접근법(semantic-based approach)의 필요성이 증가하고 있다. 특히 시멘틱-기반 접근법은 키워드-기반의 방법이 갖는 한계점을 보완할 수 있는 장점이 있다. 따라서 개선된 선행 기술 조사 알고리즘은 시멘틱(semantic) 정보와 메타데이터(metadata)를 함께 사용해야 한다. 그러면 개선된 방법은 기존보다 더 효율적일 것이다. 따라서 본 발명에서는 메타데이터(metadata)와 시멘틱(semantic) 정보를 활용할 수 있는 선행 기술 조사 알고리즘을 제안한다.

[0034] 이하에서는, 본 발명에서 제안하는 선행기술 검색 방법을 설명한다.

[0035] 본 발명에서는 피인용 네트워크와 유사도 네트워크를 이용해서 선행기술을 탐색(또는 검색)하는 방법을 제안한다.

[0036] n (n 은 임의의 자연수)개의 등록된 특허 중 i 번째 특허인 p_i 의 등록 날짜(또는 출원 날짜)를 $DATE_i$, p_i 를 인용한 특허들의 집합을 $FC(p_i)$ 라 하자. 피인용 네트워크가 $G_c = (V_c, E_c)$ 일 때, V_c 는 특허(특허의 집합)이고 E_c 는 인용 관계를 의미한다. 예를 들어, $i \neq j$ 인 두 개의 특허 $p_i, p_j \in V_c$ 에 대해, $p_j \in FC(p_i)$ 이면 p_j 가 p_i 를 인용한 것이며, 본 발명에서는 이를 $e_c(p_i, p_j) \in E_c$ 라 표현한다.

[0037] 특허들의 유사한 관계를 표현한 네트워크(유사도 네트워크)가 $G_s = (V_s, E_s)$ 일 때, p_i 와 p_j 가 수학적 1을 만족하면, $e_s(p_i, p_j) \in E_s$ 가 성립한다.

[0038] [수학식 1]

[0039]
$$SIM(f(p_i), f(p_j)) \geq SIM_{min}$$

[0040] $f: p_i \rightarrow \mathbb{R}^d$ 를 특허의 텍스트를 단어의 빈도 또는 문맥 정보를 반영한 임베딩 벡터로 매핑하는 함수라 하자. $SIM()$ 은 벡터로 매핑된 두 특허의 유사도를 계산하는 함수이다. 두 특허의 유사도가 임계값 SIM_{min} 보다 크면 두 특허는 G_s 에서 서로 연결된다. $SIM()$ 가 코사인 유사도인 경우, 수학식 2로 계산되며, 값의 범위는 -1부터 1까지이다. 두 벡터는 코사인 유사도가 1에 가까울수록 양의 상관관계가 강해지며, -1에 가까울수록 음의 상관관계가 강해진다.

[0041] [수학식 2]

[0042]
$$SIM(f(p_i), f(p_j)) = \frac{\sum^d f(p_i) \times f(p_j)}{\sqrt{\sum^d f(p_i)^2} \sqrt{\sum^d f(p_j)^2}}$$

[0043] 도 2의 (a)는 G_c 의 예시이다. 예제에서 n 은 네트워크에서 노드(특허)의 개수인 10이다. G_c 에서 p_2 는 p_1 을 인용했기 때문에 $p_2 \in FC(p_1)$ 가 성립한다. 도 2의 (b)는 G_c 와 G_s 를 결합한 네트워크이다. 도 2의 (b)에서 p_1 은 $p_3, p_5, p_6, p_9, p_{10}$ 과 유사하다. 예컨대, $SIM(f(p_1), f(p_3))$ 은 SIM_{min} 보다 크며, $SIM(f(p_1), f(p_7))$ 은 그렇지 않다.

[0044] A. PIC-EXPLORER

[0045] 발명가는 기술개발을 효율적으로 진행하고 자신들의 기술을 특허로 인정받기 위하여 유사 기술을 조사한다. 만약 조사한 특허 중, 인용이 필요한 특허나 논문이 있을 경우, 그 기술내용을 인용하고 출원할 때 그 내용을 기술한다. 이러한 활동들은 선행 기술보다 고도화된 특허를 발명함으로써 특허성(patentability)을 높일 수 있다. 따라서, 발명가들에게 유사한 특허를 빠르고 정확하게 찾을 수 있는 방법은 매우 필요하다고 할 수 있다. 또한, 대리인과 심사관들도 특허의 완결성을 높이기 위해 유사한 특허를 신속히 찾을 수 있어야 한다.

[0046] 위와 같은 문제를 해결하기 위한 여러 알고리즘이 제안되었다(J. Lee, S. Park, and J. Kang, "Introducing Patents with Indirect Connection (PIC) for Establishing Patent Strategies," Sustainability., vol. 13, no. 2, 820, Jan. 2021.). 그들은 서로 유사하지만 직접적인 인용 관계가 없는 특허쌍을 PIC (Patents with Indirect Connection)이라 정의했다. PIC-Explorer (PIC-E)는 PIC을 탐색하기 위해 특허의 인용과 유사도를 이용한다. PIC-E는 도 2의 (b)에서 p_1 과 p_3 또는 p_1 과 p_5 의 관계를 탐색한다. p_1 과 p_3 은 직접적인 인용 관계는 없지만 서로 유사하다. 즉, p_1 과 p_3 는 $p_1 \rightarrow p_2 \rightarrow p_3$ 와 같은 기술 발전(또는, 인용 관계)의 흐름을 갖는다. 이를 탐지하기 위해 PIC-E는 p_1 를 인용한 특허들인 $FC(p_1)$ 을 찾는다. 그리고 $FC(p_1)$ 에 속한 특허 중 p_3 이 인용한 특허가 있는지를 확인한다. 그러나 PIC-E는 다음과 같은 한계점이 있다.

[0047] · PIC-E는, $p_1 \rightarrow p_2 \rightarrow p_3$ 과 같이, 하나의 특허만을 공유하는 PIC만을 찾을 수 있다. 즉, 이 알고리즘은 $p_1 \rightarrow p_2 \rightarrow p_3 \rightarrow p_6$ 과 같은 PIC을 탐색할 수 없다.

[0048] · PIC-E는 매우 느리다. 이 방법은 개선한다면 p_1 과 p_6 같이 두 개 이상의 특허를 공유하는 PIC을 찾을 수 있다. 그럼에도 불구하고 이 알고리즘을 사용하면 느리다.

[0049] · PIC-E의 방식은 애매모호한 결과를 도출할 수 있다. 도 2의 (b)에서 p_1 과 p_5 는 PIC이다. PIC-E를 사용한다면, 두 특허는 $p_1 \rightarrow p_4 \rightarrow p_5$ 의 흐름을 갖는 PIC이다. 그러나 두 특허는 $p_1 \rightarrow p_2 \rightarrow p_3 \rightarrow p_4 \rightarrow p_5$ 의 흐름으로도 연결될 수 있어 정확한 경로를 파악하는 것에는 한계가 있다.

[0050] B. K-STEP PIC-EXPLORER

[0051] 앞서 언급한 PIC-EXPLORER 알고리즘의 한계점을 해결하기 위해서는 다음의 개선이 필요하다. (1) 더 긴 경로를 거치는 선행 기술을 탐색할 수 있어야 한다. (2) 탐색 시간이 짧으며 더 넓은 확장성을 갖어야 한다. (3) PIC 도출 과정을 명확히 해야 한다. 본 발명에서는 위 문제점을 해결하기 위하여 PIC-EXPLORER를 개선한 K-Step PIC-Explorer 알고리즘을 제안한다. 본 발명에서는 p_1 과 p_3 와 같이 하나의 특허(p_2)를 공유하면서 유사하지만 직접적인 인용이 없는 관계를 1step PIC이라 정의한다. 새로운 방법은 k개의 특허를 공유하면서 유사하지만 인용이 없는 선행 기술을 탐색한다. 본 발명에서는 이를 "k-step PIC-Explorer (k-step PIC-E)"로 명명한다.

[0052] 알고리즘 1은 k-step PIC-E의 수도 코드(pseudo code)이다. 먼저, 제안하는 방법은 두 개의 특허가 유사한지 판단한다. 만약, p_i 와 p_j 가 유사하다면, $e_s(p_i, p_j) \in E_s$ 가 성립한다. 그리고 두 특허의 인용이 없다면 $e_c(p_i, p_j) \notin E_c$ 를 만족한다. 즉, k-step PIC-E의 입력(input)은 유사하지만 인용 관계가 없는 특허쌍이다. k-step PIC은 두 특허 중 먼저 등록된(또는 먼저 출원된) 특허를 판단하기 위해 등록 일자(또는 출원 일자)의 차이를 계산한다. 등록 일자(또는 출원 일자)의 차이(TD)는 수학식 3으로 계산되는 시차이다.

[0053] [수학식 3]

$$[0054] \quad TD = DATE_i - DATE_j$$

[0055] 예를 들어, $DATE_i$ 가 2021년 1월 1일이고 $DATE_j$ 가 2021년 7월 30일일 때, TD는 -210이므로 음수이다. 따라서 알고리즘은 TD를 통해 p_i 가 p_j 보다 선행 기술임을 알 수 있다. 우리는 등록 일자(또는 출원 일자)가 빠른 특허를 P_E , 더 늦게 등록된(또는 출원된) 특허를 P_L 이라고 정의한다. 제안된 방법은 등록 일자(또는 출원 일자)가 같은 경우도 고려하기 위해서 TD가 0이면 p_i 를 P_E 로 입력한다. 그리고 k-step PIC-E는 두 등록 일자의 차이의 최대값을 TD_{max} (day)로 제한한다. 왜냐하면, 연구자가 선행 기술을 조사할 때, 기술의 특성에 따라 검색 범위를 조절할 필요가 있기 때문이다. 만약, 그들이 모든 선행 기술에 대해 k-step PIC을 탐색한다면 TD_{max} 를 7,300으로 설정하면 된다. 왜냐하면 특허의 권리는 최대 20년(7,300일)동안 보호받기 때문이다.

[0056] [알고리즘 1]

Algorithm 1 The k-step PIC-Explorer algorithm,

Input: $G_c = (V_c, E_c), G_s = (V_s, E_s)$

Output: list of k-step PIC

Set: k, TD_{max}

1: $e_s(p_i, p_j) \in E_s$ and $e_c(p_i, p_j) \notin E_c, i < j$

LOOP Process

2: **for** $i, j = 1$ to n **do**

3: $TD = DATE_i - DATE_j$

4: **if** $|TD| \leq TD_{max}$ **then**

5: **if** $TD \leq 0$ **then**

6: $P_E \leftarrow p_i, P_L \leftarrow p_j$

7: **else**

8: $P_E \leftarrow p_j, P_L \leftarrow p_i$

9: **end if**

10: $SP = BFS_{G_c}(P_E \rightarrow P_L)$

11: **if** $Length(SP) = k + 1$ **then**

12: P_E and P_L are k-step PIC

13: **end if**

14: **end if**

15: **end for**

[0057]

[0058] P_E 와 P_L 가 결정되면, k-step PIC-E는 G_c 에서 P_E 부터 P_L 까지 가는 최단 거리(shortest path, SP)를 찾는다. SP는 그래프(graph)에서 두 노드를 연결하는 가장 짧은 루트이다. 본 발명은 G_c 에서 SP를 찾기 위해, 대표적인 그래프 순회(graph traversal) 방법 중 하나인, Breadth-First Search (BFS)를 사용한다(E. F. Moore, "The shortest path through a maze," in Proc. Int. Symp. Switching Theory, Cambridge, MA, USA, 1959, pp. 285-292., C. Y. LEE, "An algorithm for path connections and its applications," IRE Trans. Electron Computers., vol. EC-10, no. 3, pp. 346-365, Sept. 1961, 10.1109/TEC.1961.5219222., S. S. Skiena, "Sorting and Searching," The Algorithm Design Manual, Springer, London, UK, 2012, pp. 103-144.). BFS의 특징은 다음과 같다.

[0059]

· BFS는 큐(queue)를 이용하는 검색 알고리즘으로 시작 노드로부터 인접한 모든 노드를 우선 방문한다.

[0060]

· BFS는 시작 노드에서 시작하여 다음 깊이 수준의 노드로 이동하기 전에 현재 깊이의 모든 노드를 탐색한다.

[0061]

· BFS는 시작 노드부터 목표 노드까지 SP를 보장한다.

[0062]

· k-step PIC-E는 BFS를 활용하여 두 특허 간의 SP를 빠르게 찾는다. 즉, G_c 에서 SP는 $BFS_{G_c}(P_E \rightarrow P_L)$ 이다. 예를 들어, 도 2의 (b)에서 p_1 와 p_5 는 1step PIC이다. 도 2의 (a)에서 p_1 부터 p_5 까지의 경로는 $p_1 \rightarrow p_2 \rightarrow p_3 \rightarrow p_4 \rightarrow p_5$ 와 $p_1 \rightarrow p_4 \rightarrow p_5$ 이다. 그러나, 도 2의 (c)처럼, BFS는 두 특허의 SP를 $p_1 \rightarrow p_4 \rightarrow p_5$ 로 반환한다.

[0063]

k-step PIC-E는 P_E 와 P_L 의 SP의 길이를 측정한다. 도 2의 (c)에서 $BFS_{G_c}(p_1 \rightarrow p_5)$ 의 길이는 2이다. 만약, 두 특허가 k-step PIC이면, $BFS_{G_c}(p_1 \rightarrow p_5)$ 의 길이가 k+1이다. 따라서 p_1 과 p_5 는 1step PIC이다. 표 1은 도 2의 (b)에서 k-step PIC을 탐색한 결과이다. 1step PIC은 두 개, 2step과 3step PIC은 각각 한 개씩 탐색되었다. 예를 들어, p_1 과 p_9 는 3step PIC이다. 그 의미는 두 특허가 서로 유사함에도 불구하고 인용이 없다는 것이다. 그래서 k-step PIC-E는 G_c 에서 p_1 과 p_9 의 SP를 찾는다. 탐색 결과, SP는 $p_1 \rightarrow p_2 \rightarrow p_7 \rightarrow p_8 \rightarrow p_9$ 이고, 그 길이는 4(=k+1)이다. 따라서 두 특허는 3step PIC의 관계가 성립한다.

[표 1]

P_E	P_L	k	Shortest Path with BFS
p_1	p_3	1	$p_1 \rightarrow p_2 \rightarrow p_3$
p_1	p_5	1	$p_1 \rightarrow p_4 \rightarrow p_5$
p_1	p_6	2	$p_1 \rightarrow p_2 \rightarrow p_3 \rightarrow p_6$
p_1	p_9	3	$p_1 \rightarrow p_2 \rightarrow p_7 \rightarrow p_8 \rightarrow p_9$

도 3은 본 발명의 일 실시예에 따른 선행기술 탐색 장치의 기능 블록도이고, 도 4는 도 3에 도시된 선행기술 탐색 장치에 의해 수행되는 선행기술 탐색 방법을 설명하기 위한 흐름도이다.

선행기술 탐색 장치(100)는 선행기술 검색 장치, 관계 분석 장치, 기술 관계 분석 장치, 특히 관계 분석 장치 등으로 명명될 수도 있다. 선행기술 탐색 장치(100)는 적어도 프로세서(processor) 및/또는 메모리(memory)를 포함하는 컴퓨팅 장치로 구현될 수 있다. 따라서, 선행기술 검색 방법, 관계 분석 방법, 기술 관계 분석 방법, 특히 관계 분석 방법 등으로 명명될 수도 있는 선행기술 탐색 방법의 단계들 중 적어도 일부는 컴퓨팅 장치에 포함되는 프로세서의 동작으로 이해될 수도 있다. 컴퓨팅 장치는 PC(Personal Computer), 서버(server), 랩탑 컴퓨터, 노트북 등을 포함할 수 있다.

선행기술 탐색 장치(100)는 데이터 수집부(110), 전처리부(120), 유사도 산출부(130), 인용관계 도출부(140), 선행기술 탐색부(150), 및 저장부(160) 중 적어도 하나를 포함할 수 있다.

데이터 수집부(110)는 유무선 통신망을 통하여 소정의 데이터를 수집할 수 있다(S110). 실시예에 따라, 데이터 수집부(110)는 소정의 입출력 인터페이스를 통하여 USB 메모리 장치와 같은 저장 장치로부터 데이터를 수신할 수도 있다. 또다른 실시예로, 소정의 데이터는 선행기술 탐색 장치(100)의 저장부(160)에 미리 저장되어 있을 수도 있다.

수집되는 데이터는 특히 데이터 및/또는 인용관계 정보를 포함할 수 있다. 특히 데이터는 복수의 특허들 각각에 대한 데이터로써, 해당 특허 문서 자체를 의미하거나 해당 특허 문서에 포함된 적어도 일부의 데이터를 의미할 수 있다. 해당 특허 문서에 포함된 적어도 일부의 데이터라 함은 제목, 요약, 인용특허(즉, 해당 특허가 인용하는 특허)에 대한 정보(예컨대, 출원번호, 공개번호, 또는 등록번호 등과 같은 식별 정보를 의미할 수 있음) 중 적어도 하나를 포함할 수 있다. 인용관계 정보는 복수의 특허들 간의 인용 여부에 대한 정보로써, 특허 데이터에 포함되어 있을 수도 있으나, 실시예에 따라 인용 관계에 대한 정보만을 별도로(또는 추가적으로) 수집할 수도 있다.

전처리부(120)는 수집된 데이터에 대한 전처리 동작을 수행할 수 있다(S120). 실시예에 따라, 전처리부(120)는 수집된 데이터에 기초하여 복수의 특허들 각각에 대응하는 (임베딩) 벡터를 생성할 수 있다. 또는, 전처리부(120)는 각 특허별 단어의 빈도수 정보를 추출하여 TF 행렬을 생성하거나 TF-IDF 행렬을 생성할 수도 있다. 또한, 실시예에 따라, 전처리부(120)는 임의의 사전학습 언어 모델(Pre-Trained Language Model, PTL)을 이용하여 임베딩을 수행함으로써 입력 데이터(예컨대, 각 특허의 제목, 요약, 인용특허에 대한 정보 중 적어도 하나)에 대한 임베딩(벡터)을 생성할 수도 있다.

유사도 산출부(130)는 복수의 특허들 간의 유사도를 산출할 수 있다(S130). 유사도는 소정의 유사도 산출 함수를 이용하여 산출될 수 있다. 예시적인 유사도 산출 함수는 코사인 유사도 함수이다. 또한, 유사도 산출부(130)는 산출된 유사도가 미리 정해진 임계값을 초과하는 경우(또는 이상인 경우)에 해당 특허 간에는 유사성이 있는 것으로 판단할 수 있다. 결과적으로, 유사도 산출부(130)는 복수의 특허들에 대한 유사도 행렬을 생성할 수 있다. 유사도 행렬에서, 유사성이 있는 것으로 판단된 특허 사이에는 요소 값으로 1(또는 0)이 할당되고, 유사성이 없는 것으로 판단된 특허 사이에는 요소 값으로 0(또는 1)이 할당될 수 있다.

인용관계 도출부(140)는 복수의 특허들 간의 인용관계를 도출할 수 있다(S140). 일 예로, 인용관계 도출부(140)는 복수의 특허들 간의 인용관계를 나타내는 행렬인 인용관계 행렬을 생성할 수 있다. 여기서 인용관계는 특허 간의 직접적인 인용관계를 의미할 수 있다.

선행기술 도출부(150)는 복수의 특허들 중 두 개의 특허에 대하여 선행기술 여부를 판단할 수 있다(S150). 실시예에 따라, 선행기술 도출부(150)는 복수의 특허들 중 유사성이 인정되지만 직접적인 인용관계가 존재하지 않는 특허 쌍을 도출할 수 있다.

- [0075] 구체적으로, 선행기술 도출부(150)는 복수의 특허들 중 두 개의 특허를 선택한다. 이때, 선택되는 두 개의 특허는 유사성이 인정되는 특허를 의미할 수 있다. 또한, 선택되는 두 개의 특허는 직접적인 인용 관계가 존재하지 않는 특허를 의미할 수 있다. 선택된 특허는 등록 일자(또는 출원 일자)를 기준으로 선행특허와 후행특허로 결정된다. 즉, 두 개의 특허 중에서 등록 일자(또는 출원 일자)가 앞서는 특허를 선행특허로 결정하고, 두 개의 특허 중에서 등록 일자(또는 출원 일자)가 늦는 특허를 후행특허로 결정할 수 있다.
- [0076] 다음으로, 선행기술 도출부(150)는 선행특허와 후행특허 사이의 인용관계 거리를 산출할 수 있다. 인용관계 거리는 복수의 인용관계로 연결되는 선행특허와 후행특허 사이에서 연결되는 인용의 횟수를 의미할 수 있다. 즉, 인용관계 거리는 앞서 기재한 인용관계 네트워크에서 두 특허를 연결하는 간선의 개수를 의미할 수 있다. 여기서 간선의 개수는 선행특허와 후행특허 간의 홉(hop)수에 대한 정보로 사용자에게 제공될 수 있다. 또한, 간선의 개수에서 1을 뺀 값은 k-PIC의 k값이 될 수 있다.
- [0077] 저장부(160)에는 선행기술 탐색 장치(100)의 동작에 필요한 OS(Operating System), 프로그램, 소스 코드 등이 저장되어 있을 수 있다. 또한, 저장부(160)에는 데이터 수집부(110)에 의해 수집된 데이터, 전처리부(120)에 의해 전처리된 결과, 유사도 산출부(130)의 유사도 산출의 결과, 인용관계 도출부(140)에 의해 도출된 인용관계, 선행기술 탐색부(150)에 의해 탐색된(또는 검색된) 결과 등이 저장될 수 있다.
- [0078] 이상에서 설명된 장치는 하드웨어 구성 요소, 소프트웨어 구성 요소, 및/또는 하드웨어 구성 요소 및 소프트웨어 구성 요소의 집합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성 요소는, 예를 들어, 프로세서, 컨트롤러, ALU(Arithmetic Logic Unit), 디지털 신호 프로세서(Digital Signal Processor), 마이크로컴퓨터, FPA(Field Programmable array), PLU(Programmable Logic Unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(Operation System, OS) 및 상기 운영 체제 상에서 수행되는 하나 이상의 소프트웨어 애플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술 분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(Processing Element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 컨트롤러를 포함할 수 있다. 또한, 병렬 프로세서(Parallel Processor)와 같은, 다른 처리 구성(Processing Configuration)도 가능하다.
- [0079] 소프트웨어는 컴퓨터 프로그램(Computer Program), 코드(Code), 명령(Instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(Collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성 요소(Component), 물리적 장치, 가상 장치(Virtual Equipment), 컴퓨터 저장 매체 또는 장치, 또는 전송되는 신호 파(Signal Wave)에 영구적으로, 또는 일시적으로 구체화(Embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 하나 이상의 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.
- [0080] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(Magnetic Media), CD-ROM, DVD와 같은 광기록 매체(Optical Media), 플롭티컬 디스크(Floptical Disk)와 같은 자기-광 매체(Magneto-optical Media), 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 상기된 하드웨어 장치는 실시예의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.
- [0081] 본 발명은 도면에 도시된 실시예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 가능하다는 점을 이해할 것이다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성

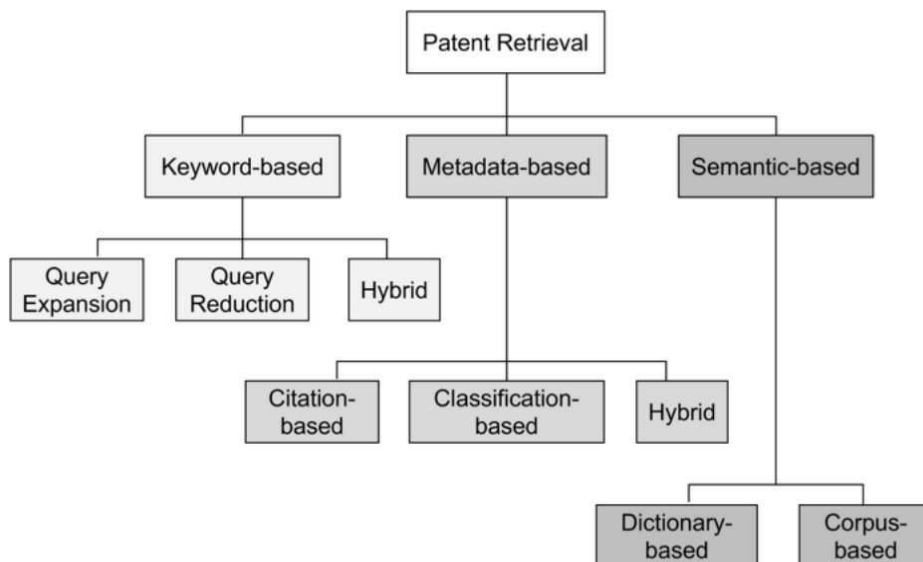
요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성 요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다. 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 등록청구범위의 기술적 사상에 의해 정해져야 할 것이다.

부호의 설명

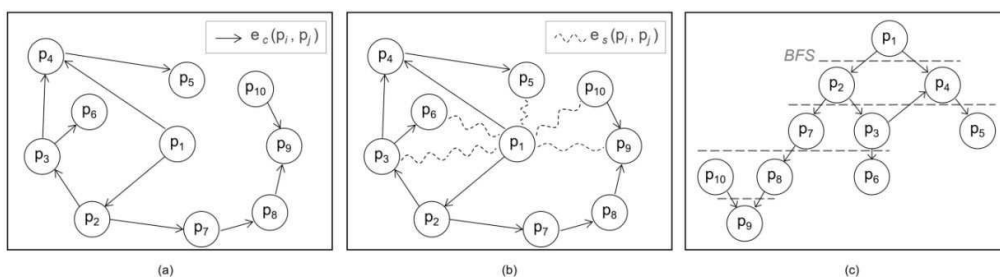
- [0082]
- 100 : 선행기술 탐색 장치
 - 110 : 데이터 수집부
 - 120 : 전처리부
 - 130 : 유사도 산출부
 - 140 : 인용관계 도출부
 - 150 : 선행기술 탐색부
 - 160 : 저장부

도면

도면1

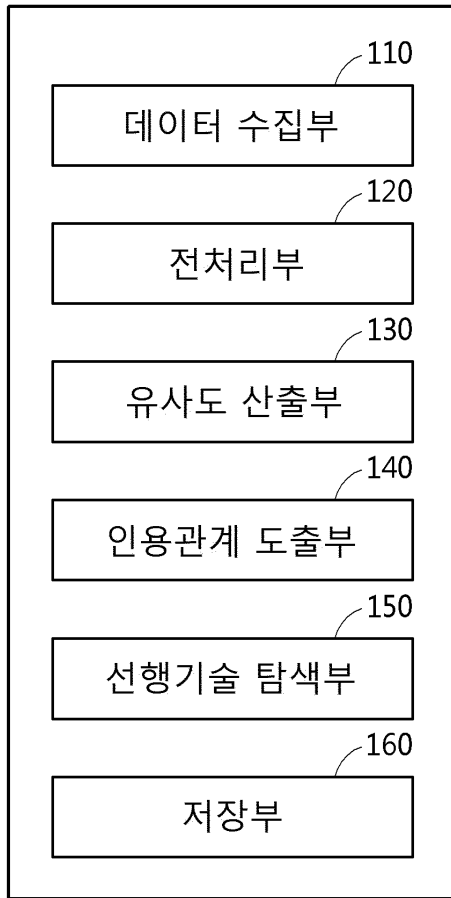


도면2



도면3

100



도면4

