딥러닝-규칙기반 병행 모델을 이용한 특허문서의 자동 IPC 분류 방법

김용일, 오유리, 심우철, 고봉수, 이봉건
한국특허정보원
{yikim, yroh0203, sim0915, kbs3579, bglee}@kipi.or.kr

Hybrid Approach Combining Deep Learning and Rule-Based Model for Automatic IPC Classification of Patent Documents

Korea Institude of Portal
Yongil Kim, Yuri Oh, Woochul Sim, Bongsoo Ko, Bonggun Lee

요 약

인공지능 관련 기술의 발달로 다양한 분야에서 인공지능 활용에 대한 관심이 고조되고 있으며 전문영역에서도 기계학습 기법을 활용한 연구들이 활발하게 이루어지고 있다. 특허청에서는 분야별 전문지식을 가진 분류담당자가 출원되는 모든 특허에 국제특허분류코드(이하 IPC) 부여 작업을 수행하고 있다. IPC 분류와 같은 전문적인 업무영역에서 딥러닝을 활용한 자동 IPC 분류 서비스를 제공하기 위해서는 기계학습을 이용하는 분류 모델에 분야별 전문지식을 직관적으로 반영하는 것이 필요하다. 이를 위해 본 연구에서는 딥러닝 기반의 IPC 분류 모델과 전문지식이 반영된 분류별 어휘사전을 활용한 규칙기반 분류 모델을 병행하여 특허문서의 IPC분류를 자동으로 추천하는 방법을 제안한다.

주제어: IPC, 특허문서 분류, 어휘사전, FastText, CNN

1. 서론

자연어처리 기술과 인공신경망 기술을 전문분야의 업무환경에 활용하고자 하는 움직임이 커지고 있다. 특허 분야에서도 특허민원상담, 특허검색, 특허분류 등 반복적인 업무를 보조하는 용도로써 인공지능을 활용한 연구가 진행되고 있다.

특허가 출원되면 특허분류 담당자는 특허문서의 기술적 내용을 파악하고 포함된 기술 내용에 해당하는 IPC를 부여한다. IPC는 기술별로 8개의 섹션, 130개의 클래스, 640개 이상의 서브클래스, 최종적으로 70,000개 이상의 서브그룹으로 이루어진 계층적 구조로 되어 있다. 예를 들어 IPC H01L 21/00에서 섹션 H는 전기, 클래스 H01은 기본적 전기소자, 서브클래스 H01L은 반도체 장치를 의미하며, 하위 레벨로 갈수록 세분화된 기술 구조를 갖는다. 경우에 따라 하나의 특허는 두 가지 이상의 기술분야가 포함되기도 하며 (예: 3D 감상용 안경(G:물리학/H:전기)), 이런 경우 주분류와 부분류로 구분하여 복수개의 IPC를 부여된다.

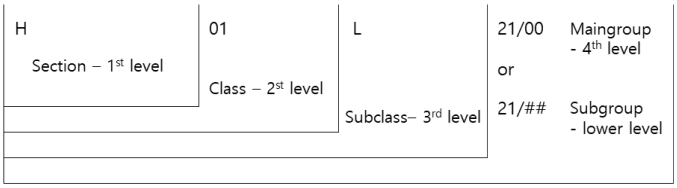

그림 1. 국제특허분류(IPC) 구성 (예시 H01L 21/00)

본 연구에서는 이처럼 특허분류 담당자가 수행하는 특허문서의 기술내용 이해와 코드부여를 자동화하는 것을 목적으로 한다. 자연어처리, 인공신경망 기술을 이용하여 문서에 등장하는 단어와 문맥을 이해하고, 그에 따라 IPC분류를 부여하는 작업을 자동화하는 시스템을 구현하고, 주분류가 H섹션인 특허문서 22,820 건으로 학습된 모델에 서브클래스(47개 분류), 메인그룹(431개 분류)를 예측하는 실험을 수행한다.

최근의 자연어처리 분야에서 사용되는 비지도학습 기반의 토크나이저[1]와 기존의 형태소 분석기의 장단점에 대해 검토하고, 형태소 분석기 사용에서 발생하는 미등록어 문제를 인공신경망 기술을 활용해 보완하는 것을 실험한다.

인공신경망은 원천데이터를 자동적으로 학습하고 이해하기 때문에 예측 결과의 도출 과정을 논리적으로 설명하기 어렵고, 전문영역에서의 직관적 지식을 개입시키기 어렵다는 한계가 있다. 이를 보완하기 위해 전문 도메인 지식을 반영할 수 있는 방법을 병행하는 방안에 대해서 제안한다.

2. 관련 연구

기존에 특허의 IPC 분류를 자동으로 부여하고자하는 연구는 국내외적으로 이루어지고 있다. 연구[6]에서는 딥러닝 기반의 특허 문서 분류시스템의 통계적 기반에 대해 설명하였고, 연구[3]는 특허문서의 특징과 구조적 필드의 역할을 기반으로 TF-IDF와 나이브베이즈 분류기를 기반으로 한 IPC 다중 분류기를 제안하고 검증하였다.

그리고 국제지식재산권 기구인 WIPO에서는 영문, 불문 특허를 대상으로 인공신경망 기반으로 자동 IPC 분류를 수행하는 IPCCAT-NMT 서비스를 개시하였다.[7]

자연어 형태의 텍스트를 의미 기반으로 처리하기 위해서는 단어를 의미 단위로 나누는 토크나이징 과정이 필

요하다. 토크나이징의 방법은 대표적으로 문장을 문법적으로 분석하여 형태소 단위로 분할하고 품사를 부여해주는 형태소 분석기가 있다. 그러나 형태소 분석기는 사전에 등록되지 않은 단어의 경우 처리가 불가능하다는 단점이 있다. 최근 자연어처리 연구에서는 이러한 미등록어 문제를 해결하기 위한 다양한 방법들이 제시되고 있다. 비지도학습 기반 토크나이징은 문법적 요소를 고려하지 않고, 통계적으로 가장 유리한 방법으로 단어를 분할한다[1]. 또한 단어의 n-gram을 skip-gram방식으로 학습하여 언어모델을 생성하고 미등록어라할 지라도 벡터표현 단계에서 의미를 부여할 수 있는 방안도 연구되고 있다[2].

인공신경망에 데이터를 입력 · 학습하기 위해서는 데이터의 특징이 압축된 벡터 형식으로 표현해야한다. 문서의 특징을 추출하여 벡터 형태로 표현하기 위해 전통적으로는 TF-IDF를 통해 단어주머니를 생성하는 방법[3], LSA, LDA 등 토픽모델링 및 차원축소 기법이 사용되었다[4]. 최근에는 인공신경망을 통해 자동으로 특징을 추출하는 Autoencoder 또한 차원축소와 특징추출을 대체하는 알고리즘으로서 사용되고 있다[5].

3. 자동 IPC 분류 구현 방법

본 절에서는 특허문서의 자동 IPC 분류를 위한 특징 추출 방법과 시스템 구현 방법을 단계별로 설명 한다.

한국어분석기의 선택과 문서임베딩 과정에서 FastText를 이용한 미등록어 문제의 보완에 대해 설명한다. 또한 분류를 특정할 수 있는 용어 선별을 통해 전문가의 지식을 반영할 수 있는 시스템을 제안한다.

그림2는 제안하는 자동 IPC 분류 시스템의 구조이다. 먼저 특허문서를 문장 단위로 분할하여 FastText로 임베딩된 데이터를 학습한 모델과 TF-IDF를 이용한 주제벡터를 기반으로 학습한 모델, 두 모델 각각에서의 예측결과를 재순위화하여 최종 예측결과를 도출하게 된다.

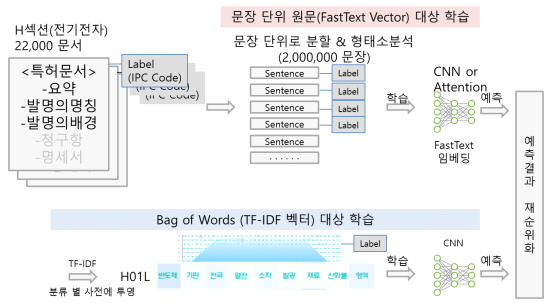

그림 2. 제안하는 자동 IPC 분류 시스템의 구조

3.1 문서 전처리

형태소 분석기는 파이썬 기반의 한글 형태소 분석 패키지로서 konlpy를 사용하고, 분석기는 mecab을 활용하였다.

원천 텍스트데이터를 두 가지 형태로 준비하였는데 첫 번째는 특허문서의 원문 문장을 형태소 별로 토큰화한 데이터, 두 번째는 명사(NNG), 고유명사(NNP)에 대해서만 추출한 데이터로 준비하였다. 첫 번째 데이터는 원문 학습과정에, 두 번째 데이터는 TF-IDF와 분류별 어휘사전 생성 과정에 활용하였다.

3.2 분류별 어휘사전-TFIDF를 이용한 주제벡터 생성

단어의 TF-IDF 계산 및 벡터화를 위해 gensim 주제분석 툴킷을 사용하였다. 문서의 명사, 고유명사를 추출하고, 각 문서를 TF-IDF 벡터의 집합으로 변환하였다.

각각의 분류에서 중요도가 높은 단어를 선별하여 분류별 어휘 사전을 생성하였다. 만약 전문적인 지식에 의해 미리 구축된 분류 별 어휘사전이 있다면 이러한 과정을 통해 생성한 사전 대신 사용할 수 있다.

각 분류 별 문서 집합에서 TF-IDF의 평균점수가 높은 단어 순으로 상위 300 단어를 추출하였다. (그림 3)

H01J	H01K	H01L	H01M	H01P	H01Q	H01S
전극	램프	반도체	전지	공진기	안테나	레이저
방출	필라멘트	기판	리튬	도파관	방사	파장
방전	할로겐	열전	전해질	선로	대역	광학
램프	베이스	전극	배터리	필터	급전	다이오드
이온	벌브	소자	이온	유전체	방사체	광섬유
패널	전구	발광	전극	스트립	패턴	펄스

H01K : 백열램프(방전장치와 백열램프의 양쪽에 사용되는...)
H01M : 화학에너지를 전기에너지로 변환하기 위한 방법

그림 3. 분류별 어휘사전 및 해당 IPC분류 설명

마지막으로 각 문서의 TFIDF 행렬과 분류 별 어휘사전을 2만개의 전체 어휘사전에 투영한 마스크 행렬를 곱하여, 문서를 일종의 주제 벡터 형태로 변환하였다. (그림 4)

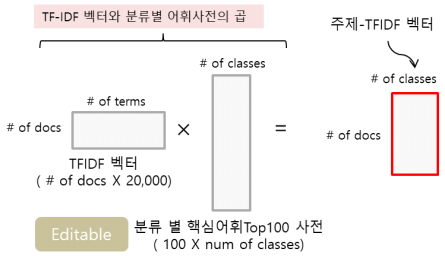

그림 4. 분류별 어휘사전을 이용한 주제벡터 생성

3.3 원문 문장 학습데이터

특허문서를 콜론(:), 구두점(.)을 기준으로 문장 형태로 분리하고, 형태소 분석을 통해 품사와 관계없이 공백을 구분자로 토큰화된 문장을 생성하였다.

토큰화에 형태소 분석기를 사용하였기 때문에, 약 20만 건의 문서를 분석한 결과 10만개를 상회하는 단어가 발생하였다. 이후 과정에서 1~2만개 수준의 단어사전을 생성하거나 임베딩 과정에서 미등록어가 문제가 될 수 있다. 이러한 문제를 해결하기 위해 n-gram을 결합한 skip-gram모델인 FastText를 문장 임베딩에 사용하였으며, 형태소 단위로 토큰화된 문장을 FastText 모델을 통해 10,000 차원의 부동소수점 벡터로 변환하였다.

3.4 모델 생성 및 추론

본 논문에서 문장 분류기는 CNN 구조를 사용하였고, 어텐션 메커니즘에 의한 Seq2Seq 구조를 추가로 실험하였다. 실험에 활용한 CNN 구조는 필터 사이즈 128과 윈도우 사이즈를 갖는 2개의 1차원 Convolution - Maxpooling 레이어에 최종적으로 Dense 전연결 레이어와 Softmax 활성화함수

를 추가한 모델로 학습하였다. 손실함수는 categorical cross entropy, 최적화함수는 Adam optimizer를 사용하였다. 추가적으로 실험한 Attention-LSTM 네트워크는 output dimension 96, hidden dimension 128을 갖는 2개 깊이의 네트워크를 설계하였고, 이외의 부분은 앞에서 설명한 CNN 구조와 동일하게 설정하였다.

4. 실험 및 결과

4.1 학습데이터 구축

공개된 모든 특허데이터에는 IPC분류가 부여되어 있다. 그러나 IPC분류 별로 데이터의 양에 있어 큰 차이가 있으므로 데이터 수량에 따른 bias가 없는 학습을 위해서는 언더샘플링이나 Augmentation이 필요하였다. 본 실험에서는 H섹션(전기) 분류 내에서 데이터가 너무 적은 분류를 제외한 47개의 서브클래스와 431개의 메인그룹을 기준으로 분류를 수행하였다. 그리고 총 100만 여개의 문서 내에서 언더샘플링을 하여 특허문서 22,820건를 사용하였다. 특허의 주/부분류 중 주분류에 대해서만 학습 및 예측하도록 하였다.

실험은 1)문장 단위로 분할된 데이터를 활용하는 것이 성능을 개선하는가에 대한 검증 2)주제벡터를 이용한 모델을 같이 활용하는 경우의 성능 개선에 대한 검증 두가지로 진행하였다. 문서 단위 데이터는 특허문서의 제목, 요약, 배경기술, 기술분야 4가지 요소를 합하여 최대 5,000개 단어 토큰으로 구성하였고, 요소 단위데이터는 4가지 요소를 각각의 문서로 간주하고, 문장 단위 데이터는 3.3에서 설명한 바와 같이 하였다.

표 1. 사용한 학습 데이터

데이터 단위	원본문서 (개)	학습데이터 (개)	최대 길이 (word)
문서	22,820	22,820	5,000
요소		91,280	2,000
문장		711,464	100

또한 최종 데모시스템 구축에서는 3.2절에서 설명된 바와 같은 22,820개의 주제벡터 파일과 3.3절에서 설명된 바와 같은 문서를 문장 단위로 분할한 5,294,097개의 문장을 사용하였다.

4.2 실험 결과

4.2.1 데이터 단위 별 성능 비교

문서를 문장 단위로 분할하는 경우의 성능향상을 검증하기 위해 총 2,844건의 검증데이터로 서브클래스에 대해 문서, 요소, 문장 별로 입력데이터를 달리하여 실험을 진행하였다. CNN으로의 입력은 keras 패키지의 Embedding 레이어를 사용하였다.

표 2. 데이터 단위별 서브클래스 예측 정확도

데이터 단위	서브클래스 예측 정확도(%)	
	top1	top5
문서	68	92
요소	75.1	92.5
문장	**75.4**	**95.5**

실험 결과, 문서 단위, 요소단위로 학습한 모델보다 성능이 근소하게 우위에 있고, 기계학습 과정에서 노이즈가 되는 문장을 자동적으로 제외하고, 분류의 단서가 되는 문장의 특징을 적절하게 선택하는 것을 확인할 수 있었다.

4.2.2 FastText와 Attention-LSTM 적용

FastText 임베딩 시의 성능향상, Attention-LSTM 모델의 CNN 대비 성능향상을 확인하기 위해 추가적으로 실험을 진행하였다.

실험 4.2.1에서 사용한 CNN 모델에서 임베딩 레이어를 사전에 학습된 FastText 가중치 벡터로 대체하여 실험한 결과, 성능이 향상되었다. 그리고 CNN을 Attention-lstm 모델로 변경하여 실험한 결과에서도 성능향상이 이루어졌다.

표 3. fasttext, Attention 적용 이후 예측 정확도

Embedding	Model	서브클래스 예측 정확도(%)	
		top1	top5
fasttext	CNN	76.6	95.3
fasttext	Attention-lstm	77.2	95.6

4.2.3 주제벡터 모델 병합 시 성능 검증

그림 2과 같이 주제벡터 데이터를 학습한 모델에서의 예측결과와 문장 단위 학습모델에 의한 예측 결과를 병합하여 개별 모델의 예측결과에 비해 성능 향상이 있는지 측정하였다. 총 2,844건의 검증데이터로 서브클래스와 서브클래스보다 한 단계 세분화된 분류 레벨인 메인그룹에 대해 예측하였다. 그리고 각 모델에서 예측된 결과의 Confidence값을 서로 비교하여 최종 예측결과를 결정하였다.

표 4. 모델별 예측 정확도

Model	서브클래스 예측 정확도(%)	
	top1	top5
주제벡터 학습모델	67.1	93.5
문장 학습모델	75.4	95.5
병합 모델	75.3	96.0

Model	메인그룹 예측 정확도(%)	
	top1	top5
주제벡터 학습모델	41.9	74.1
문장 학습모델	44	73.2
병합 모델	**47**	**76.8**

그림 5. 특허문서 자동 IPC 분류 데모시스템

실험 결과, 각각의 모델에서의 예측한 결과보다 두가지 모델을 병합한 결과가 우위에 있는 것을 확인할 수 있었다. 이는 주제벡터 학습모델에서는 전역의 어휘특성에 대한 특징이 주로 학습되어 있고, 문장 학습모델에서는 지역적인 문맥에 대한 특징이 주로 학습되어 있기 때문에 상호보완 효과가 나타나는 것으로 추측된다. 그리고 그 효과는 세분화된 분류에서 더 크게 나타나는 것으로 보인다.

4.2.4 특허문서 자동 IPC 분류 데모시스템

제안한 모델을 활용하여 입력된 텍스트나 XML특허문서에 대해 자동IPC 분류를 제공하는 데모시스템을 제작하였다.

5. 결론

딥러닝 알고리즘만으로 시스템을 구현하는 것은 상당한 정확도를 가지지 않고서는 사용자의 신뢰를 얻기가 어렵다. 또한 전문영역에서의 시스템은 사용자가 지식을 반영시킬 수 있는 것을 원하기 때문에 기존 규칙기반으로 딥러닝 기반의 시스템을 보완하는 것이 필요할 것으로 생각된다.

본 연구를 통해 딥러닝 모델과 규칙(분류별 어휘사전) 기반의 모델을 병합 시, 그 결과가 성능향상에도 도움이 되며 딥러닝 모델에 사용자의 지식을 개입시킬 여지가 있는 것을 확인하였다.

본 실험에서는 기존에 정교하게 정의된 분류 별 어휘사전이 없어 임의로 사전을 생성하였지만, 전문적 지식을 반영한 정교한 사전이 있다면 딥러닝 기반 모델을 보완하는 효과가 커질 것으로 보인다.

향후 연구로는 높은 분별력을 갖는 분류별 어휘사전의 구축과 의존구문분석 결과를 활용하여 특허문서의 특징을 대변할 수 있는 영역 또는 문단을 선택하는 방법에 대한 연구가 수행될 필요가 있다.

참고문헌

[1] Wu, Yonghui, et al. "Google's neural machine translation system: Bridging the gap between human and machine translation." arXiv preprint arXiv:1609.08144 (2016).

[2] Li, Bofang, et al. "Subword-level composition functions for learning word embeddings." Proceedings of the second workshop on subword/character level models. 2018.

[3] 임소라, and 권용진. "특허문서 필드의 기능적 특성을 활용한 IPC 다중 레이블 분류." 인터넷정보학회논문지 18.1 (2017): 77-88.

[4] Drummond, Anna, Zografoula Vagena, and Chris Jermaine. "Topic models for feature selection in document clustering." Proceedings of the 2013 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, 2013.

[5] Liang, Hong, et al. "Text feature extraction based on deep learning: a review." EURASIP journal on wireless communications and networking 2017.1 (2017): 211.

[6] Xia, Bing, L. I. Baoan, and Xueqiang Lv. "Research on patent document classification based on deep learning." 2016 2nd International Conference on Artificial Intelligence and Industrial Engineering (AIIE 2016). Atlantis Press, 2016.

[7] WIPO IPCCAT,https://wipo.int/ipccat, 2019.