
예제기반의 학습을 이용한 한국어 표준 산업/직업 자동 코딩 시스템

An Automatic Coding System of Korean Standard Industry/Occupation Code Using Example-based Learning

임희석*

한신대학교 소프트웨어학과*

Heui-Seok Lim(limhs@hs.ac.kr)*

요약

통계청에서 실시하는 사업체 기초 조사와 인구주택총조사 과정에 업체와 개인에 대한 정보를 기술한 자연어를 표준 산업/직업 코드를 할당하는 수동 코딩 작업이 필요하다. 수동 코딩 작업은 막대한 인건비와 비용을 초래하고 수동 코딩 전문가의 능력과 기분에 따른 작업 결과의 비일관성이 매우 큰 문제로 지적되고 있다. 본 논문은 수작업으로 구축한 규칙베이스를 사용하는 규칙 기반 방법과 수작업으로 분류한 데이터를 이용하는 자동 학습 방법을 통합한 한국어 산업/직업 표준 코드 자동 생성 시스템을 제안한다. 제안하는 시스템은 인구주택총조사 40만 레코드, 사업체기초조사 40만 레코드를 이용하여 학습되었고, 실험 데이터를 이용하여 평가되었다. 10-best 성능 평가 결과 제안된 시스템은 인구주택총조사 직업분류 데이터에 대해서 76.63%, 인구주택총조사 산업분류 데이터에 대해서 82.249%의 성능을 보였으며, 사업체기초조사 산업분류 데이터에 대해서는 99.68%의 정확도를 보였다.

■ 중심어 : | 수동 코딩 | 자동 코딩 | 기계학습 | 산업(직업)코드분류 |

Abstract

Standard industry and occupation code are usually assigned manually in Korean census. The manual coding is very labor intensive and expensive task. Furthermore, inconsistent coding is resulted from the ability of human experts and their working environments. This paper proposes an automatic code classification system which converts natural language responses on survey questionnaires into corresponding numeric codes by using manually constructed rule base and example-based machine learning. The system was trained with 400,000 records of which standard codes was assigned. It was evaluated with 10-fold cross validation and was tested with three code sets: population occupation set, industry set, and industry survey set. The proposed system showed 76.63%, 82.24 and 99.68% accuracy for each code set.

■ Keyword : | Manual Coding | Automatic Coding | Machine Learning | Industry(Occupation) Code Classification |

* 본 연구는 한신대학교 2005년도 학술연구비 지원에 의하여 연구되었음

I. 서론

통계청에서 실시하는 인구 및 주택 조사는 매 5년(0년, 5년)마다 실시되고 있다. 인구주택 총조사의 방법으로는 전체 가구를 조사하는 전수 조사와 전체의 10%만을 발췌하여 조사하는 표본 조사가 있다. 사업체 기초통계조사는 연단위로 이루어져 통계청에서 주관하는 사업체 중 연속성이 있는 조사이다.

인구 주택 총조사나 사업체 기초통계 조사 방법은 조사원들이 조사 대상을 방문하여 그들에게 직접 문의하는 방법으로 이루어진다. 조사 항목은 크게 거주 지역, 출생 지역, 성씨, 나이, 표준산업분류 코드, 표준 직업분류 코드 등 정형화된 데이터와 각 개인이 근무하고 있는 사업체 명, 사업체의 주된 사업 내용, 자신의 직책, 그리고 직무 등을 나타내는 비정형화된 데이터로 구분된다. 정형화된 데이터 중 표준산업분류코드와 표준 직업분류 코드는 국가의 경제, 산업, 예산 등의 국가 기본 정책을 수립하는데 있어서 기반이 되는 중요한 지식이다. 표준산업분류 코드와 표준 직업분류 코드는 조사원이 가구 조사에서 얻은 사업체명, 사업체의 주된 사업 내용, 직책, 그리고 직무에 대한 자연어로 기술한 설명에 근거하여 작성된다. 현재까지 표준산업분류코드와 직업분류 코드 분류 작업은 한국 표준산업분류 책자와 표준직업분류 책자를 참조하여 수작업으로 이루어지고 있으며, 이로 인하여 아래와 같은 문제점이 발생하고 있다.

- 수작업을 수행하기 위한 작업자 교육 및 활용에 많은 비용이 소요 : 표준산업분류 코드와 표준직업분류 코드를 분류할 처리 레코드는 약 300만개에 해당된다. 따라서 300만개의 레코드를 일일이 사람이 코드를 부여할 경우 막대한 수작업이 필요하며, 이에 따른 비용이 매우 크다.
- 매번 반복 작업에 따른 인력 및 비용 소모 : 일년에 한번 또는 5년에 한번씩 이루어지는 통계 조사에 따른 반복된 인력 동원 및 수작업으로 인하여 비용 소모가 매우 크다.
- 구축된 자료의 부정확성 : 수작업으로 코드를 부여할 경우 작업자의 심리 상태, 작업 환경 등에 따라서 실수로 코드를 잘못 부여할 경우가 종종 발생하

게 된다. 또한 수작업자들이 대부분 일정 기간 교육을 받아 분류 작업에 투입된 비전문가들이기 때문에 코딩 결과의 정확도가 떨어진다.

- 구축된 자료의 일관성 유지의 어려움 : 자료가 매우 방대하기 때문에 여러 사람이 작업을 나누어 수행하게 되므로 같은 산업이나 직업에 대해서도 각자가 정확한 분류를 찾지 못한다면 부여된 코드가 서로 다를 수 있다. 수작업자의 산업과 직업에 대한 선입관과 관점에 따라 코딩 결과가 다양성을 띠게 된다.

위와 같은 수작업에 의한 표준 코드 분류 작업의 문제점을 극복하기 위한 방법은 가구 조사에서 얻은 자연어의 응답을 표준 분류 코드로 분류할 수 있는 자동 코드 분류 시스템을 개발하여 활용하는 것이라 할 수 있다. 미국과 캐나다와 같은 외국에서는 1980년대부터 이미 자동 코드 분류 시스템에 대한 연구를 시작하여 현재까지 꾸준히 수행하고 있으며, 높은 정확도를 보이는 시스템이 표준 산업/직업 코드 자동 분류를 위하여 활용되고 있다[1,2,3,4]. 이에 반하여 국내의 통계 조사를 위한 자동 분류 시스템에 대한 연구는 매우 미흡한 실정이다. 한국어로 쓰인 가구 조사 내용을 표준 산업/직업 코드로 분류 작업은 언어적인 차이뿐만 아니라 한국어만의 특성으로 인하여 외국에서 개발된 시스템을 직접 활용하는 것은 불가능하며 자체적인 개발이 필요한 실정이다.

본 논문은 수작업으로 구축한 규칙베이스를 사용하는 규칙기반 방법과 사례 데이터를 이용하는 자동 학습 방법을 통합한 한국어 산업/직업 표준 코드 자동 생성 시스템을 제안하며 또한 사용자의 피드백을 이용한 코드 분류 시스템의 자동 성능 향상 기법을 제안한다.

II. 시스템 개요

본 논문이 제안하는 시스템은 조사원들로부터 획득된 '근무 사업체명', '사업체의 주된 업무', '직책', 그리고 '직무'에 대한 내용을 자연어로 입력받아 입력된 내용에 해당되는 직업/산업 표준 코드를 생성하며, [그림 1]은 제안하는 시스템의 전체 시스템 구성을 도식화한 것이

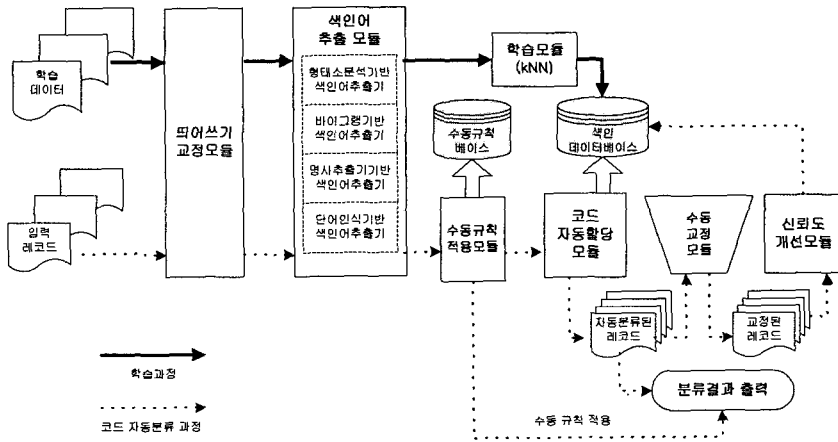


그림 1. 전체 시스템 구성도

다. [그림 1]에서 보인 것과 같이 제안하는 시스템은 크게 학습 모듈과 자동 코드 생성 모듈로 구성된다. 학습 모듈(learner)은 수작업으로 정확한 분류 코드가 할당된 학습 데이터를 입력받아 입력 데이터의 띄어쓰기 오류를 수정하는 띄어쓰기 교정 모듈, 색인어 추출 모듈, 2-포아송 모델에 의하여 색인어의 가중치를 계산하여 역화일 형식의 색인어 DB를 구성하는 kNN(k-nearest neighbors)기반의 학습 모듈로 구성된다. 자동 코드 생성 모듈(automatic code generation module)은 띄어쓰기 모듈, 색인어추출 모듈, 전문가에 의하여 작성된 규칙을 적용하는 수동 규칙 적용 모듈, 수동 규칙에 의해서 분류가 되지 않은 레코드들에 대하여 입력 레코드와 유사한 예제를 검색하고 검색된 결과의 유사도 값을 이용하여 분류 코드의 유사도를 계산하는 코드 자동 할당 모듈, 그리고 시스템이 자동으로 잘못 분류한 코드의 올바른 코드에 대한 피드백을 입력받고, 이를 이용하여 학습데이터의 신뢰도를 재조정하는 신뢰도 개선 모듈(refiner)로 구성된다. 다음 장부터는 학습 모듈과 코드 자동 생성 모듈 내의 각 기능에 대한 상세한 설명과 실험 결과를 보인다.

III. 언어처리엔진

언어처리엔진은 학습 데이터내의 예제 코드 또는 코

드 분류를 위하여 입력된 자연어내의 띄어쓰기 오류를 교정하고, 이들로부터 색인어를 추출하는 기능을 수행하며 언어처리엔진은 아래 [그림 2]와 같이 크게 띄어쓰기 모듈과 음절 바이그람 색인기, 명사 추출기, 형태소 분석기, 단어 인식기 등을 이용한 색인어 추출 모듈로 구성된다.

1. 띄어쓰기 모듈

자동 띄어쓰기는 문장 내에서 잘못 띄어쓴 어절들을 올바르게 복원하는 과정이다. 한국어의 경우, 띄어쓰기는 독자에게 글의 가독성을 높이고 문장의 뜻을 정확히 전달하기 위해 매우 중요하다. 예를 들어, “아버지가 방에 들어가셨다(Father entered the room)”를 “아버지가방에 들어가셨다(Father entered the bag)”와 같이 띄어 쓰면 전혀 다른 의미의 문장이 된다.

자동 띄어쓰기와 관련해서 기존의 연구들은 입력 대상 문서에 따라 띄어쓰기가 부분적으로 잘못되어 있는 문서를 대상으로 한 경우와 띄어쓰기가 전혀 되어 있지 않은 문서를 대상으로 한 경우로 나눌 수 있다. 대부분의 기존 연구들은 후자에 해당된다. 후자는 부분적으로 띄어쓰기 오류가 있는 문서가 입력되더라도 모두 붙여 쓰거나 띄어 쓴 후에 일괄적으로 교정을 할 수 있으므로 전자에 비해 처리할 수 있는 대상 문서의 범위가 넓다. 그러나 입력 문장에서 올바르게 띄어 쓴 부분을 잘못 수정할 수도 있다. 전자는 주어진 문장의 띄어쓰기

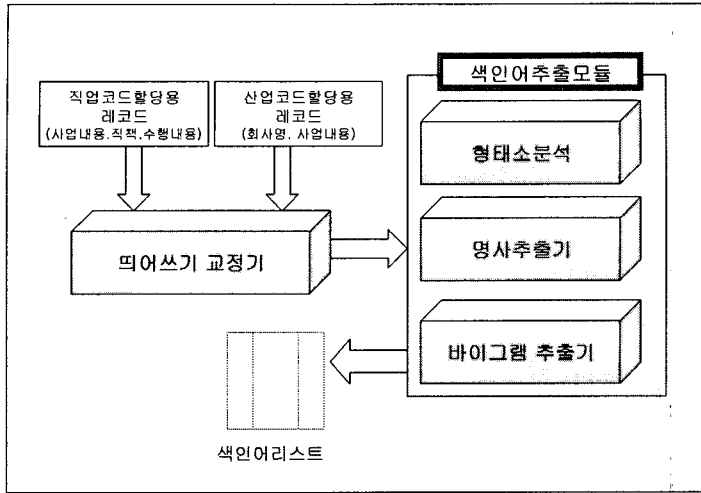


그림 2. 언어처리 엔진

상태가 어느 정도 신뢰할 만한지를 시스템이 스스로 판단할 수 없으므로 동일한 내용의 문서라도 그 문서의 띄어쓰기 상태에 따라 생성되는 결과가 달라질 수 있다. 결국 입력 당시의 띄어쓰기 상태와 그 상태를 고려할 것인 지의 여부에 따라 각각 장단점이 있다. 본 연구에서는 띄어쓰기가 전혀 되어 있지 않은 경우를 대상으로 하고 있다.

자동 띄어쓰기의 방법은 접근 방법에 따라 크게 규칙 기반 방식과 통계 기반 방식으로 나눌 수 있으며 본 논문은 말뭉치로부터 인접한 두 음절간의 띄어 쓸 또는 붙여 쓸 확률을 학습하여 이를 이용하여 띄어쓰기 오류를 교정하는 [5]에서 제안된 통계기반 방식을 사용한다. 이 방식은 대량의 원시 말뭉치로부터 자동으로 음절 정보를 획득할 수 있으므로 어휘 지식이나 규칙을 작성하거나 유지, 보수에 드는 비용이 필요하지 않고, 형태소 분석기를 사용할 때와는 달리 문장에 존재하는 미등록어에 대해서도 견고한 분석이 가능하다.

본 논문이 사용하는 띄어쓰기 모델은 문장 내에 주어진 음절열 $S = (s_1, s_2, \dots, s_n)$ 에 대해 최적의 띄어쓰기 태그열 $T = (t_1, t_2, \dots, t_n)$ 를 찾는 것으로 식 1과 같이 정의된다.

$$\text{argmax}_T P(T | S) \quad (1)$$

[식 1]에서 띄어쓰기 태그는 해당 음절과 다음 음절의 사이를 띄어쓸 것인가 붙일 것인가에 대한 태그로서 이진 값 0 또는 1을 갖으며 0은 해당 음절과 다음 음절을 붙여쓰라는 태그이고, 1은 띄어쓰라는 태그이다. 예를 들어 문장 “학교에서 공부를 참 열심히 합니다.”의 경우 띄어쓰기 태그가 부착된 형태로 표기하면 “학/0+교/0+에/0+서/1+공/0+부/0+를/1+참/1+열/0+심/0+히/1+합/0+니/0+다/0+./1”와 같다. 결국 띄어쓰기 모델은 띄어쓰기 태그열 T 와 음절열 S 의 결합 확률 $P(T, S)$ 를 최대로 하는 띄어쓰기 태그열 T 를 찾는 것이다.

띄어쓰기 모델에서 사용하는 확률값은 대규모의 원시 코퍼스에 나타난 빈도로부터 최우추정(Maximum likelihood estimation)에 의해 계산하며 입력 문장에 대해 최적의 띄어쓰기 태그열은 동적 알고리즘인 Viterbi 알고리즘을 이용하여 계산한다. [그림 3]은 입력 문장 “공부할수있다”의 띄어쓰기태그를 결정할 예를 보인 것이다.

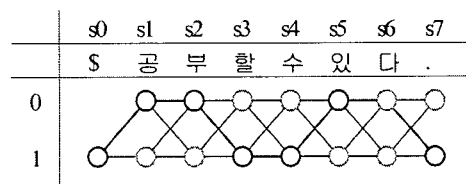


그림 3. 띄어쓰기 수행의 예

2. 색인어 추출 모듈

색인어 추출 모듈은 색인 DB를 구성하기 위하여 학습 데이터로부터 색인어를 추출하거나 코드 검색을 위하여 입력 데이터로부터 검색어를 추출하는 모듈이다. 기존의 한국어 색인어 추출 방식은 크게 형태소 분석기를 이용한 방식, 명사 추출기를 이용한 방식, 단어 인식을 이용한 방식, 그리고 바이그램 추출 방식으로 나눌 수 있다. 각 색인어 추출 방식은 각 방식 나름대로의 장단점을 가지고 있으며 코드 분류를 위하여 입력되는 자연어의 특성상 어느 방법이 적합한지를 미리 결정하기는 매우 어려운 일이다. 따라서 본 논문은 4가지의 색인어 추출 방식을 모두 구현하여 각 방법을 이용하였을 때의 성능을 평가/분석하여 산업/직업 코드 자동 분류 작업을 위하여 가장 좋은 색인어 추출 방식을 찾아내고 이를 최종적으로 이용하고자 한다.

형태소분석을 이용한 색인어 추출 방식은 형태소 분석을 통하여 얻어진 가능한 모든 형태소-품사 결합열에 대해서 수행된 품사 태깅 결과중 명사로 태깅되어 있는 결과를 색인어로 사용하는 방법으로 본 논문은 [5]에서 제안된 형태소 분석기와 품사 태깅 시스템을 사용하였다. 형태소 분석을 이용한 색인어 추출 방식은 미등록어 처리와 복합 명사 인식을 할 수 있다는 장점을 가지고 있지만 품사 태깅의 오류로 인한 잘못된 색인어 추출과 많은 양의 계산 시간을 요구하는 단점이 있다. 명사추출기를 이용한 색인어 추출 방식은 명사가 사용되는 환경과 조사와 같이 명사와 함께 출현할 수 있는 문자열 정보를 이용하여 색인어를 추출하는 방식으로 대용량의 자료에서 고속으로 색인어를 추출할 수 있다는 장점을 가지며 본 논문은 [6]의 명사 추출기를 이용하여 색인어 추출 실험을 하였다. 바이그램을 이용한 색인어 추출 방식은 언어적인 처리에 의해 명사를 추출하는 것이 아니라 인접한 두 음절을 이용하여 색인어를 추출하는 방식으로 입력 내용의 첫음절부터 마지막 음절까지 연속된 두 음절을 색인어로 추출한다. 예를 들어, “우리밭에서 콩 심고 나물도 심어 경작함”이라는 입력이 들어오면 “우리”, “리밭”, “밭에”, “에서”, “서”, “콩”, “콩”, “심”, “심고”, “고”, “나”, “나물”, “물도”, “심”, “심어”, “어”, “경”, “경작”, “작함”을 색인어로 추출한다.

이 방법은 언어적인 처리로도 추출하기 어려운 색인어를 추출할 수 있으며 띄어쓰기 오류와 철자 오류에 대해서 견고하며 복합 명사 인식과 신조어에 대한 색인어 처리가 가능하다는 장점을 갖는다.

IV. 학습 및 자동 코드 할당

본 논문이 제안하는 시스템은 사람에 의해 수동으로 분류된 모든 기본류 레코드들을 미리 색인해 놓고 새로 입력된 레코드를 질의로 색인된 기본류 레코드들을 검색하여, 검색된 레코드의 순위와 새로 입력된 레코드와의 유사성을 참고하여 적절한 산업분류코드와 직업분류코드를 할당하게 된다. 산업분류의 코드할당을 위한 색인시에는 레코드의 필드들 중 사업체명과 사업내용, 직업분류의 코드할당을 위한 색인시에는 레코드의 필드들 중 사업내용과, 직책, 수행내용에 관한 필드들만을 색인대상으로 삼게 된다. 본 절에서는 색인과 검색, 그리고 코드할당 시스템에 대하여 기술한다.

1. 자동 학습

본 논문은 기계 학습 방법 중 kNN 방식의 학습 방법을 채택하여 사용하였는데 이 방법을 채택한 이유는 다음과 같다. 첫째, 코드 자동 분류 작업을 위한 입력 레코드의 길이는 30어절에서 50어절 크기로 매우 짧은 레코드의 검색이 매우 용이하다. 둘째, 이전의 통계 조사 때 수작업으로 분류된 대량의 예제 데이터를 활용할 수 있다. 셋째, 분류하여야 할 범주의 수가 매우 많다.

kNN 방식의 학습은 인공지능분야에서 연구되어 온 기계학습알고리즘의 일종으로 메모리 기반 학습 또는 예제 기반 학습이라고도 한다[7]. 이 방법은 대표적인 비모수(nonparametric) 기계학습방법으로서 각 개체를 이루고 있는 확률분포를 가정하지 않아 실제로 데이터의 확률분포가 널리 알려진 정규분포나 이항분포, 혹은 포이송분포를 따르지 않는 많은 영역에서 매우 우수한 성능을 보여주는 학습방법이다. kNN 방식의 학습은 다른 기계 학습방법과는 달리 학습예제집합에 대한 일반화과정을 수행하지 않고 예제들을 특징을 추출하여

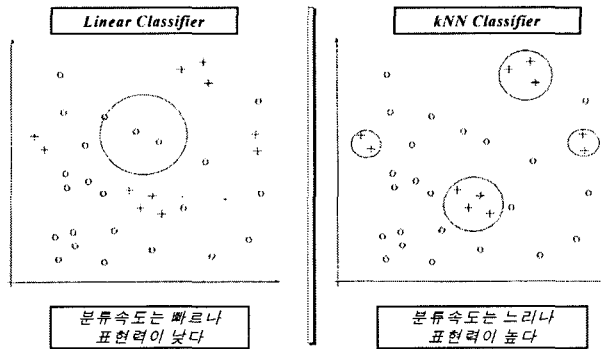


그림 4. kNN방법과 선형분류기의 비교

이를 저장하는 것으로 학습 과정이 끝나게 된다. 즉, 가설공간에서 분류를 위한 최적의 가설을 미리 결정해 두는 것이 아니라, 분류하는 시점에, 분류할 개체와 유사한 학습예제들을 선별하고, 선별된 학습예제들만이 가설 결정에 참여하는 지연학습(lazy learning)이다. 제안된 시스템의 경우 코드 자동 할당을 위하여 입력된 데이터와 유사한 코드 검색을 위하여 사용될 단어의 출현 빈도만을 색인하여 저장하는 것으로 학습 과정이 끝나게 된다.

kNN과 같은 지연학습은, 개체를 분류할 시점에서 일반화를 수행하기 때문에 분류속도가 느다는 단점이 있으나, 일반화를 수행함에 있어서 새로 분류할 데이터와 유사한 데이터만을 사용함으로써 학습과정에서의 노이즈를 줄여줄 수 있다는 장점을 갖고 있다. [그림 4]는 k-NN방법과 선형 분류기에 대한 차이를 극명하게 보여주고 있다.

2. 수동 규칙의 적용 모듈

본 시스템에서는 자동 코드할당 이외에도 사용자가 정의한 규칙에 맞는 데이터의 경우에는 수동 규칙에 따라 코드를 할당할 수 있는 매커니즘을 제공한다. 수동 규칙은 산업/직업 코드 분류를 수행하는 통계청의 전문가들에 의해서 수작업으로 구축된 규칙으로서 '조건-행위'의 형식으로 구축되어 있다. 제안하는 시스템은 코드 분류의 정확도를 높이기 위하여 입력 레코드를 수동 규칙에 적용을 하여 수동 규칙에 적용되면 규칙에 의해서 코드를 할당하며 규칙이 적용되지 않는 레코드의 경우 자동 코드 할당 모듈에 의해서 분류 코드를 할당하도록

한다.

본 논문의 실험 결과에는 수동 규칙을 적용한 후 수동 규칙으로 코드 분류가 되지 않은 레코드에 대해서만 자동 분류 코드를 할당하는 경우의 성능 평가는 수행하지 않는다. 그 이유는 현재 구축되어 있는 수동 규칙이 [표 1]과 같이 분류하고자 하는 코드의 일부분에 해당하는 코드 분류만을 위한 규칙만이 구축되어 있어 규칙의 적용률이 매우 낮아 대부분의 코드 분류가 자동 분류 결과로 생성되기 때문이다. 현재 구축되어 있는 규칙을 2,882개의 실험 레코드에 적용한 결과 약 98.9%의 정확도를 얻을 수 있었으며, 이 때 정확도는 규칙이 적용된 레코드 수에 대한 정확한 코드가 할당된 레코드의 수의 비율을 의미한다.

표 1. 수동 규칙 구축 현황

	산업분류	직업분류
분류 대상 코드 수	442	447
규칙이 구축된 코드 수	140(31.67%)	49(10.96%)
구축된 규칙 수	2,090	565
코드 1개당 규칙 수	14.93	11.53

3. 코드 자동 할당

코드 자동 할당 과정은 입력 레코드와 유사한 코드를 검색하고 검색된 코드와 입력 레코드와의 유사도를 이용하여 최종적인 출력 코드를 계산하는 방식으로 이루어진다.

제안된 시스템에서 사용하는 검색 모델은 2-포이송 모델로 TREC-8의 Okapi 시스템[8]이 사용하였던

BM25 방법에 의하여 식 1과 같이 코드 j의 단어 i의 가중치 w_{ij} 를 식 2와 같이 계산하며 식 3을 이용하여 입력 레코드 q와 코드 j와의 유사도를 계산한다.

$$w = \frac{(k_1 + 1) \cdot tf_{ij}}{k_1 + tf_{ij}} \log \frac{p(1 - q)}{q(1 - p)} \quad (2)$$

$$\text{sim}(d_j, q) = \sum_{i \in d_j} \left(\frac{(k_1 + 1) \cdot tf_i}{k_1 \cdot \left((1 - b) + b \cdot \frac{dl_j}{\text{avdl}} \right) + tf_i} \times \log \frac{N - df_i + 0.5}{df_i + 0.5} \right) \cdot qtf_i \quad (3)$$

식 2에서 k_1 은 실험을 통해 얻어내야 할 파라미터이며 tf_{ij} 는 코드 j에서 단어 i의 출현 빈도를 나타내며 p는 단어 i가 코드 j에 출현할 확률을 나타내며 q는 단어 i가 코드 j가 아닌 다른 코드에서 출현할 확률값을 의미한다. 식 3에서 avdl은 학습 데이터의 코드들의 평균 길이를, dl_j는 코드 j의 길이를 의미하며 df_i는 단어 i가 출현한 코드의 개수를 qtf_i는 질의에 단어 i가 출현한 빈도를 의미한다.

코드 자동 분류를 위하여 새로 입력된 레코드를 질의로 하여 기분류된 레코드들을 검색한 후 식 3에 의한 유사도 값을 이용하여 랭킹한 후 상위 k번째까지 이웃들의 유사도 값을 참고하여 식 4에 의해서 최종적으로 레코드 r_i가 코드 c와 갖는 스코어값을 계산한다.

$$CSV_c(r_i) = \sum_{t_j \in kNN} \text{sim}(r_i, t_j) y(t_j, c) \text{conf}(t_j) \quad (4)$$

즉, 2-Poisson 모델에 의해 계산된 스코어가 상위 k개에 속하는 기분류 레코드들 중에 코드 c가 할당된 레코드들의 스코어를 합산하여 새로 입력된 레코드와 코드 c와의 결합점수를 구해내는 방식이다. 식 4에서 위에서 $y(n_j, c)$ 는 이웃 j의 코드 값과 c와 일치하는 경우 1 그렇지 않은 경우 0값을 출력하는 함수이며 $\text{conf}(n_j)$ 는 상위 k개의 이웃 중 j번째 이웃의 신뢰도를 의미하는

것으로, 사용자 피드백을 이용한 성능 개선을 위해 사용되는 값이며 초기에 이 값은 1.0으로 초기화되어 있다.

V. 사용자 피드백을 이용한 성능개선

본 논문은 코드 자동 할당 모듈의 성능을 점진적으로 개선시킬 수 있는 방법으로 정보 검색에서 사용하는 사용자 연관 피드백 개념을 도입하였다[9, 10]. 본 논문이 제안한 사용자 피드백을 이용한 방법은 많은 양의 레코드에 대해 오류를 생성하는 학습데이터의 신뢰도를 낮게 변경하여 추후에는 그 학습 데이터가 입력 레코드와 유사한 레코드로 검색되어 자동 코드로 분류될 수 있는 가능성을 낮추도록 하는 것이다. 제안하는 사용자 피드백을 이용한 성능 개선 방법은 다음과 같다. 학습을 위해 주어진 모든 레코드의 신뢰값은 디폴트로 1.0으로 초기화된다. 이렇게 초기화된 상태에서 새로운 레코드들에 대해 코드할당을 수행하는데, 사람이 수작업으로 시스템의 잘못된 코드할당사례를 찾아낸 경우 이러한 사례들을 모아 로그파일을 구성한다. 이렇게 구성된 로그파일은 코드할당 성능개선기의 입력으로 주어지는데, 시스템은 어떠한 학습데이터가 잘못 코드할당을 하도록 유도했는지 내부적으로 판단하고 그 데이터의 신뢰도를 현재 신뢰도의 99%로 낮춘다. 반대로 제대로 된 코드를 할당할 수 있게끔 역할을 하는 데이터들의 경우는 현재 신뢰도의 101%로 높여주게 된다. 개별 오류사례에 대해 이러한 업데이트를 수행하게 되면, 만일 특정 학습레코드가 여러 오분류 사례를 유발시킨 경우 해당 학습레코드의 신뢰도는 매우 낮아지게 된다. 시스템은 내부적으로 신뢰도가 특정 값 이하인 학습레코드에 대해서는 코드 할당시 사용하지 않는 방법으로 성능의 개선을 꾀한다.

VI. 실험결과

1. 자동 분류 결과

실험은 인구주택총조사와 사업체기초조사로 나뉘며,

인구주택총조사의 경우 주어진 레코드를 직업분류코드와 산업분류코드를 할당해야 하며, 사업체기초조사의 경우 산업분류코드를 할당해야 한다. 한국 표준 산업 분류와 직업 분류 코드는 1수준에서부터 5수준까지 계층적으로 분류되어 있으며 각 수준의 분류 코드의 수는 [표 2]와 같다.

표 2. 한국표준산업(직업) 분류 코드 분류 체계

코드	수준				
	1	2	3	4	5
산업분류	20	63	194	442	1,121
직업분류	11	46	162	447	1,404

본 논문은 인구주택총조사를 위한 분류에서는 4수준 코드로의 분류를 실험하였고, 사업체기초조사를 위한 분류로 5수준의 분류 코드를 사용하였다. 즉 인구주택총조사를 위한 분류 대상은 산업 코드를 위한 442개, 직업 코드를 위한 447개였으며 사업체기초조사를 위한 분류 대상은 1,404개였다.

자동 코드 분류 시스템의 학습과 실험은 [표 3]과 [표 4]에서 보인바와 같이 무작위로 추출한 레코드를 이용하였다.

표 3. 학습 데이터

항목	레코드수
인구주택총조사에 대한 직업분류코드	400,000
인구주택총조사에 대한 산업분류코드	400,000
사업체기초조사에 대한 산업분류코드	427,767

표 4. 실험 데이터

항목	레코드수
인구주택총조사에 대한 직업분류코드	10,000
인구주택총조사에 대한 산업분류코드	7,671
사업체기초조사에 대한 산업분류코드	35,697

표 5. 인구주택총조사에 대한 직업분류코드 실험 결과

	바이그림	명사추출	명사추출+ 자동띄어쓰기	형태소분석	형태소 분석+ 자동띄어쓰기
1위	42.80	41.86	41.84	40.88	41.23
2위	56.82	55.74	56.27	55.34	55.42
3위	63.85	62.76	63.45	62.31	62.71
4위	68.22	66.94	67.96	66.89	67.20
5위	71.07	69.70	70.98	69.70	70.16
6위	72.98	71.99	72.94	71.55	71.86
7위	74.38	73.45	74.20	72.95	73.20
8위	75.42	74.46	75.22	74.08	74.11
9위	76.10	75.28	75.96	74.72	74.89
10위	76.63	75.89	76.69	75.49	75.73

시스템의 자동 분류의 성능을 평가하기 위한 평가 척도로는 N-best 정확도를 이용하였다. N-best 정확도란 실험에 사용된 전체 레코드의 수에 대해, 시스템이 출력한 상위 N개의 레코드 중에 정답이 포함된 비율로서 계산하였다.

아래 [표 5], [표 6], 그리고 [표 7]은 각각 인구주택총조사에 대한 직업분류코드, 인구주택총조사에 대한 산업분류코드, 사업체기초조사에 대한 산업분류코드에 대한 실험 결과를 보인 것이다. 표에서 각 행은 상위 1개부터 10개까지의 결과를 출력했을 때의 N-best 정확도를 나타내며, 각 열은 색인어 추출방식에 따른 실험 결과를 나타낸다.

표 6. 인구주택총조사에 대한 산업분류코드 실험 결과

	바이그림	명사추출	명사추출+ 자동띄어쓰기	형태소 분석	형태소 분석+ 자동띄어쓰기
1위	66.90	64.87	64.63	65.26	63.60
2위	76.13	73.88	74.20	74.10	72.66
3위	78.88	76.95	77.56	76.94	76.17
4위	80.11	78.59	79.06	78.44	78.37
5위	80.98	79.75	79.77	79.46	79.06
6위	81.44	80.35	80.43	79.91	79.82
7위	81.79	80.67	80.90	80.30	80.28
8위	82.00	80.89	81.18	80.67	80.52
9위	82.09	81.14	81.33	80.80	80.72
10위	82.24	81.27	81.50	80.92	80.80

표 7. 사업체기초조사에 대한 산업분류코드 실험 결과

	바이그림	명사추출	명사추출+ 자동띄어쓰기	형태소분석	형태소 분석+ 자동띄어쓰기
1위	95.84	94.79	94.94	95.63	95.10
2위	98.49	97.95	98.09	98.40	98.08
3위	99.14	98.69	98.76	98.97	98.74
4위	99.38	99.05	99.08	99.21	99.08
5위	99.51	99.25	99.20	99.34	99.20
6위	99.60	99.34	99.30	99.42	99.27
7위	99.63	99.37	99.33	99.47	99.35
8위	99.65	99.42	99.40	99.50	99.38
9위	99.66	99.43	99.42	99.51	99.39
10위	99.68	99.44	99.44	99.51	99.43

실험 결과에 따르면, 색인어 추출 방식 중에서는 세가지 실험 모두에서 바이그림 색인 방법이 가장 높은 성능을 보이며, 명사 추출과 형태소 분석을 이용한 방법이 그 뒤를, 단어 인식 방법을 이용한 경우가 가장 낮은 성능을 보인다. 가장 단순한 바이그림 색인 방법의 성능이 가장 우수했다는 것은 띄어쓰기 문제와 미등록어 처리에 가장 견고했음을 의미하는 것으로 추측된다.

실험 결과 중 직업분류코드를 할당한 실험 결과가 전체적으로 가장 낮은 정확도를 보였으며 [표 7]의 결과인 사업체기초조사에 대한 산업분류코드 실험은 5자리의 코드 1,404개의 코드 중 올바른 코드를 할당하는 작업임에도 불구하고 매우 높은 정확도를 보이고 있다. 이러한 실험 결과는 인구주택총조사에 대한 코드 분류가 사업체기초조사에 대한 코드 분류와 비교하여 문제 자체의 난이도가 높다는 것을 의미한다.

조사원들이 조사한 자료를 입력하는 과정에서 발생하는 자동 띄어쓰기 문제를 극복하기 위하여 시도된 자동 띄어쓰기의 적용은 성능 향상에 약간의 영향을 미친 것으로 확인되었으나 바이그림 색인어 추출 방식을 사용하는 경우보다 높지 않은 성능을 보였다. 이는 띄어쓰기 오류나 철자 오류를 포함할 수 있는 데이터를 처리하고자 할 때 띄어쓰기 오류 교정기를 적용하는 것보다 바이그림 색인어 추출 방식을 사용하는 것이 비용이나 성능 면에서 우수한 결과를 보일 수 있음을 시사하는 것이다.

2. 사용자 피드백을 이용한 성능 개선 실험 결과

이 프로그램은 기본적으로 학습 및 자동분류를 수행한 후, 자동분류된 결과를 사람이 수정하였을 때, 이 수정된 레코드파일을 입력으로 주어 사용하게 된다. 시스템이 이 레코드 파일을 읽으면, 자동분류를 내부적으로 수행하는데, 그 결과가 사람이 할당해놓은 코드가 다르게 되면, 틀린 코드로 자동분류를 수행하는데 기여한 학습레코드들의 신뢰도를 떨어뜨려 점진적으로 신뢰도가 낮은 레코드는 높은 스코어로 새로운 데이터와 유사할 지라도 분류코드 할당과정에서 신뢰도만큼 배제되도록 하는 것이다. 모든 학습 레코드는 기본적으로 1.0의 신뢰도로 시작되게 하였고, 잘못된 코드를 할당하는데 기여할 때마다 기존신뢰도의 0.99 배의 값으로 떨어뜨리도록 하였다. [표 8]과 [표 9]는 각각 인구주택총조사를 위한 산업/직업 분류 코드와 사업체기초조사를 위한 산업분류 코드를 20,00개의 사용자 피드백 자료를 이용하여 성능 개선 실험을 한 결과를 보이고 있다.

[표 8]과 [표 9]에서 보인 바와 같이 본 논문이 제안한 사용자 피드백 정보를 이용한 자동 코드 할당기의 성능이 직업코드는 5% 가까이 성능이 향상되었고 다른 경우들도 조금씩 향상되었음을 알 수 있다.

표 8. 사용자 피드백 성능 개선 실험 결과 (인구주택총조사를 위한 분류)

	산업코드 (개선전)	산업코드 (개선후)	직업코드 (개선전)	직업코드 (개선후)
정분류	13807	14218	11392	12214
오분류	6193	5782	8608	7786
정확도	69.03%	71.09%	56.96%	61.07%

표 9. 사용자 피드백 성능 개선 실험 결과 (사업체기초조사를 위한 분류)

	사업체코드 (개선전)	사업체코드 (개선후)
정분류	17271	17347
오분류	2729	2653
정확도	86.36%	86.74%

VII. 결론

본 논문은 인구통계조사를 통하여 수집된 산업/직업 분류에 관한 자연어로 기술된 내용을 입력받아 해당 표준 코드로 분류하는 산업/직업 코드 자동 분류 시스템을 제안하였다. 수작업에 의한 산업/직업 코드 분류 작업은 막대한 시간과 노동력을 필요로 하는 작업뿐만 아니라 수동 분류자의 작업 환경 및 기분에 따라서 분류 결과가 달라지는 비밀관성 문제가 심각하게 발생되나 자동 분류 시스템을 사용할 경우 이와 같은 문제점을 감소시킬 수 있다는 장점을 갖는다. 제안된 시스템은 코드 분류에 관한 전문가가 이전에 수작업으로 분류한 데이터를 활용할 수 있도록 메모리 기반 학습의 일종인 kNN 학습 기법을 이용하여 학습 및 자동 분류하도록 하였으며, 자동 코드 분류 결과에 대한 전문가의 피드백을 활용하여 자동 코드 할당기의 성능을 점진적으로 개선시킬 수 있는 신뢰도 개선 방법을 제안하였다. 또한 통계 조사로부터 수집된 산업/직업에 관한 입력 자료의 특성을 반영할 수 있고 높은 성능을 보일 수 있는 색인어 추출 방법을 찾기 위하여 다양한 한국어 색인어 추출 방법을 모두 구현하고 비교 평가하였다.

제안된 시스템은 인구주택총조사 40만 레코드, 사업체기초조사 40만 레코드를 이용하여 학습되었고 학습에 참여하지 않은 레코드를 실험 데이터로 이용하여 성능을 평가되었다. 실험 결과 제안된 시스템은 바이그램을 이용한 색인어 추출 기법을 사용하였을 때 가장 높은 성능을 보였다. 10-best 성능 평가 결과 인구주택총조사 직업분류 데이터에 대해서 76.69%, 인구주택총조사 산업분류 데이터에 대해서 82.249%의 성능을 보였으며 사업체기초조사 산업분류 데이터에 대해서는 99.68%의 정확도를 보임을 확인할 수 있었다. 사용자의 피드백을 이용한 신뢰도 개선기의 사용은 인구주택총조사 직업분류 데이터에 대해서 약 5%의 성능 향상을 보였으나 제안된 방법의 성능 향상 여부가 아직 증명된 상태는 아니다.

제안된 시스템은 코드 분류를 위하여 수작업을 전혀 사용할 필요가 없는 완전 자동화 시스템으로 사용하기 위해서는 아직 성능이 떨어지는 상황이지만 10-best의

결과 중 올바른 코드를 할당하도록 함으로써 코드 분류를 하는 전문가의 작업을 경감시킬 수 있는 반자동 도구나 수동 작업자의 코드 분류 결과를 검증할 수 있는 도구로서는 충분히 활용될 수 있다고 판단된다. 향후에는 코드 분류에 대한 전문가의 도메인 지식과 수동 규칙의 활용, 그리고 신뢰도 개선기를 통한 지속적인 성능 향상에 대한 연구를 계속하여 진행할 계획이다.

참고문헌

- [1] M. V. Apeel and E. Hellerman, "Census Bureau Experiments with Automated Industry and Occupation Coding," Proceedings of the American Statistical Association, pp.32-40, 1983.
- [2] E. Rowe and C. Wong, An Introduction to the ACRT Coding System, Bureau of the Census Statistical Research Report Series No. RR94/02 1994.
- [3] <http://www.census.gov/srd/papers/pdf/rr94-4.pdf>
- [4] B. Chen, R. H. Creecy, and M. Appel, "On Error Control of Automated Industry and Occupation Coding," Journal of Official Statistics, Vol.9, No.4, pp.729-745, 1993.
- [5] 임희석, 언어 지식과 통계 정보를 이용한 한국어 품사 태깅 모델, 고려대학교 컴퓨터학과 박사학위논문, 1997.
- [6] 이도길, 이상주, 임채강, "명사 출현 특성을 이용한 효율적인 한국어 명사 추출 방법," 제30권, 제2호, pp.173~183, 2003.
- [7] T. M. Mitchell, Machine Learning, Mc Graw Hill, 1997.
- [8] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford, "Okapi at TREC-3," in the Proceedings of Text REtrieval Conference (TREC-3), 1995.

- [9] B. Yates and R. Neto, Modern Information Retrieval, Addison-Wesley, 1999.
- [10] G. Salton, and M. J, McGill, Introduction to Modern Information Retrieval, McGraw-Hill, New York, 1983.

저 자 소 개

임 희 석(Heui-Seok Lim)

정회원



- 1992년 2월 : 고려대학교 컴퓨터학과(이학사)
 - 1994년 2월 : 고려대학교 컴퓨터학과(이학석사)
 - 1997년 8월 : 고려대학교 컴퓨터학과(이학박사)
 - 1997년 9월~1999년 2월 : 삼성종합기술원 전문연구원
 - 1999년 3월~2004년 2월 : 천안대학교 정보통신학부 교수
 - 2004년 3월~현재 : 한신대학교 소프트웨어학과 교수
- <관심분야> : 인공지능, 정보검색, 생물정보학