문서의 계층화를 이용한 문서비교 방법

[1]황명권[O] [1]공현장 [2]황광수 [1]김판구
[1]조선대학교 컴퓨터공학과
{mghwang[O], kisofire, pkkim}@chosun.ac.kr
[2]조선대학교 컴퓨터공학과
hwang00ks@gmail.com

The Method of Document Comparison using Document Hierarchy

Myunggwon Hwang[O] Hyunjang Kong Kwangsu Hwang Pankoo Kim
Dept. of Computer Engineering Chosun University

요 약

오늘날 웹의 비약적인 성장으로 텍스트, 이미지, 비디오, 그리고 사운드 등의 다양한 데이터 형식의 많은 정보가 축적되었으며 날마다 늘어나고 있다. 이들 정보의 효율적 검색을 위해 많은 연구가 이루어졌으며, 특히 텍스트 문서의 효율적인 검색을 위해 확률을 이용한 방법, 통계적인 기법을 이용한 방법, 벡터 유사도를 이용한 방법, 베이지안 자동문서 분류 방법 등이 제안되었다. 그러나 이러한 기존의 방법들은 문서의 특징을 정확하게 반영할 수 없고, 의미적 검색이 이루어지지 않는 단점을 가지고 있다. 이에 본 논문은 문서를 미리 분류하는 기존의 방법을 개선하기 위해, 사용자가 원하는 문서와 비슷한 문서를 의미적으로 찾아내기 위한 방법을 제안한다. 본 방법론은 문서의 내용을 의미적인 계층으로 표현하고 중요 도메인에 가중치를 두어 각 문서들의 계층들의 도메인 비중과 도메인 내의 개념 일치도를 이용하여 문서들 간에 유사도를 구한다.

1. 서 론

웹의 비약적인 성장으로, 현재의 웹은 인간이 필요로 하는 모든 정보를 텍스트, 이미지, 비디오, 사운드 등의 다양한 데이터 형식으로 담고 있으며, 매일 새로운 정보들이 생성되고 추가된다. 이들 정보의 검색을 위해 각종 포털 사이트들이 등장했으며, 대부분의 검색이 검색 엔진을 통하여 이루어지고 있다. 특히, 텍스트 기반의 웹 문서는 검색을 위해 일반적으로 키워드 매칭 방법을 이용하고 있다. 하지만, 이러한 방법을 통해 나타나는 검색 결과는 단순히 특정 키워드의 출현빈도와 무의미한 단어 매칭을 통한 것이기 때문에 사용자는 원하는 정보를 찾기 위해 다시 검색을 수행해야 하는 문제점이 있다.

웹 문서의 효율적인 검색을 위해, 문서를 자동으로 분류하는 방법으로 확률을 이용한 방법[1,2], 통계적인 기법을 이용한 방법[3,4], 벡터 유사도를 이용하는 방법[2], 베이지안 확률을 사용한 방법[5] 등이 연구되었다. 이 연구들은 사용자가 검색을 수행할 때 문서안에 출현한 단어들 또는 미리 학습된 규칙을 이용해서, 문서들을 분류하며, 질의어에 해당하는 문서그룹을 보여주는 방식이다. 하지만 이들 분류방법은 문서에 대한 정확한 특징을 반영할 수 없고 자동분류 이후에 사람의 수동적 재분류를 필요로 하며, 문서의 의미를 구체적이고 정확하게 반영할 수 없어 이를 보완하기 위해 국내 연구에서 [6]이 제안되었다.

본 논문에서는 문서를 미리 분류하는 기존의 검색방법을 개선하기 위해, 사용자가 웹 에서 특정의 문서를 선택하였을 때 선택된 문서와 유사한 문서를 검색하기 위한 문서를 의미적으로 비교하는 방법을 제안한다. 본 방법은 사용자가 원하는 문서와 비슷한 문서를 의미적으로 찾아주기 위한 방법으로, 문서를 구성하고 있는 단어들 중에서 명사만을 추출하여, 미리 구축된 대형의 온톨로지를 지식베이스로 이용하여 명사들 사이의 전체적인 계층 구조를 파악한 후, 형성된 트리를 비교하여 문서의 유사도를 측정하는 방법으로 문서의 계층화 방법 및 계층 비교 방법으로 구성되어져 있다. 이를 위해, 형성된 트리를 비교하기 위한 수식을 제안하고 있다.

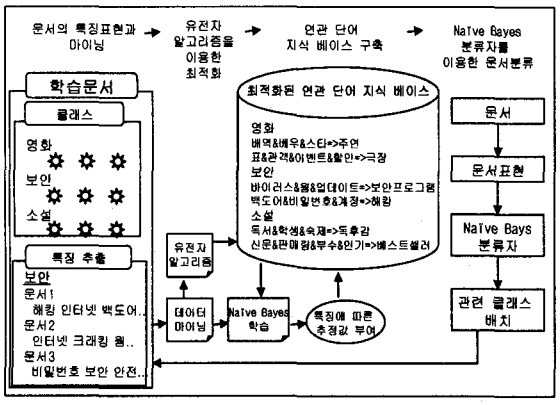
본 논문은 2장에서 본 연구에 필요한 요소들을 상세히 설명하고, 3장에서 본 논문의 핵심인 문서의 계층화 방법과 계층 비교 방법을 기술한 후, 4장에서 결론과 향후 연구 방향을 제시하며 마무리한다.

2. 관련연구

본 장에서 기존의 의미적 문서 자동 분류 방법인 베이지안 자동 문서 분류 방법을 응용한 연구를 소개하고, 문서의 의미적 유사도 비교에 필요한 지식베이스로서 본 연구에서 사용하는 워드넷의 전반적 내용과 내부 구성에 대해 상세히 설명한다.

2.1 베이지안 자동 문서 분류 방법

문서의 자동 분류에 대한 기존 연구방법들의 잡음으로 인한 오분류, 단어 의미 중의성 문제점들을 보완하기 위해 Apriori-Genetic 알고리즘을 이용한 베이지안 분류 방법이 제안되었다. Apriori 알고리즘은 단어간의 의미를 반영하여 연관단어 형태로 추출하고 잡음으로 인한 오류를 줄이기 위해 추출된 연관단어를 이용하여 연관 단어 지식베이스를 구축한 후, 구축된 지식베이스를 유전자 알고리즘을 이용하여 최적화한다. 그리고 베이지안 확률을 이용하여 최적화된 연관 단어 지식베이스를 기반으로 새로운 입력 문서를 클래스별로 분류한다. [그림 1]은 Apriori-Genetic 알고리즘의 전체 과정을 나타내고 있다.

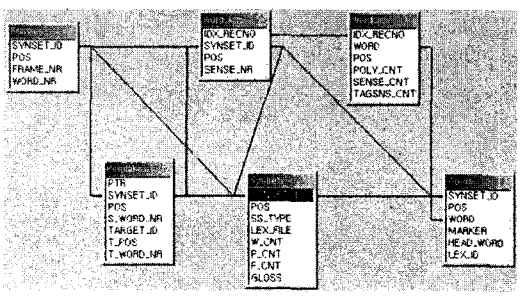


[그림 1] Apriori-Genetic 알고리즘 적용 분류 방법

본 방법은 최적화된 연관단어 지식베이스 구축으로 Naive Bayes 분류자가 정확하며 빠른 문서 분류가 가능하며, 실험문서의 특징을 연관 단어의 형태로 표현함으로써 단어 의미의 중의성 문제를 해결할 수 있다는 장점이 있다.

2.2 워드넷(WordNet)

워드넷은 현재까지 가장 널리 사용되는 범용의 대형 온톨로지로써 실제 그 내용은 6개의 데이터베이스 테이블들로 구성되어져 실세계에 존재하는 어휘에 대해서 체계적으로 정의하고 있다. 워드넷에서 중요한 내용은 바로 개념간의 관계를 정의하고 있는 부분이며, 사전과 가장 큰 차이점이 또한 바로 이러한 부분이다.[9]



[그림 2] WordNet 데이터베이스 관계도

[그림 2]는 워드넷 내의 6개의 테이블의 논리적 관계를 잘 나타내고 있다.

워드넷은 크게 4개의 카테고리(명사, 형용사, 부사, 동사)로 분류되고, 그 안에는 다시 45개의 소카테고리로 분류되어져 있다. 그리고 워드넷 내의 모든 개념들은 특정의 심볼들을 사용하여 각 개념들간의 관계를 표현하고 있다. [표 1]은 워드넷에서 사용되는 개념들 간의 관계를 정의해 놓은 심볼들과 그 의미를 설명하고 있다.

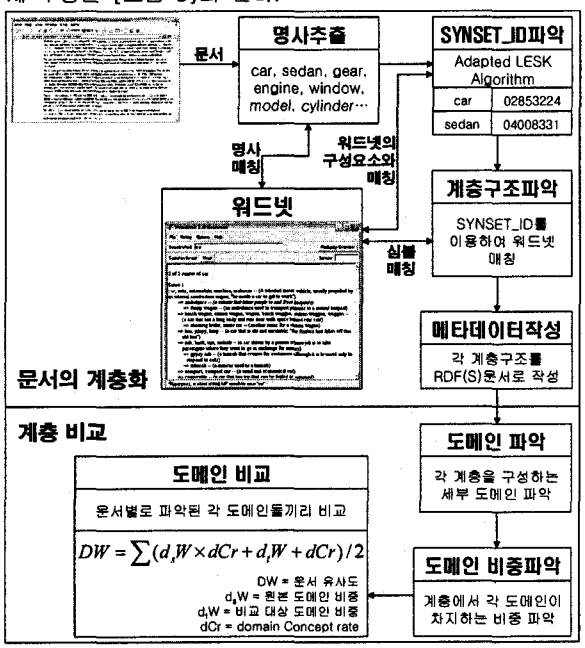
[표 1] 심볼과 의미

워드넷	의미
!	반의어
@	상위 개념
~	하위 개념
동일 synset ID	동일 의미(유의어)

이러한 워드넷은, 영어 단어 개념들 간의 의미와 관계를 상세하게 정의해 놓은 범용의 대형 온톨로지로서 미국의 프린스턴 대학(Princeton University)에서 10년 이상 개발하고 있다. 또한 자바 워드넷 라이브러리(Java Wordnet Library)가 제공되어 이를 응용한 많은 연구가 진행되고 있다.

3. 문서의 계층화 방법 및 계층 비교 방법

사용자가 원하는 문서와 비슷한 문서를 의미적으로 검색하기 위해, 본 연구에서는 각각의 문서를 계층화한 후, 생성된 계층을 서로 비교하는 방법을 제안한다. 본 연구의 전체 구성은 [그림 3]과 같다.



[그림 3] 전체 구성

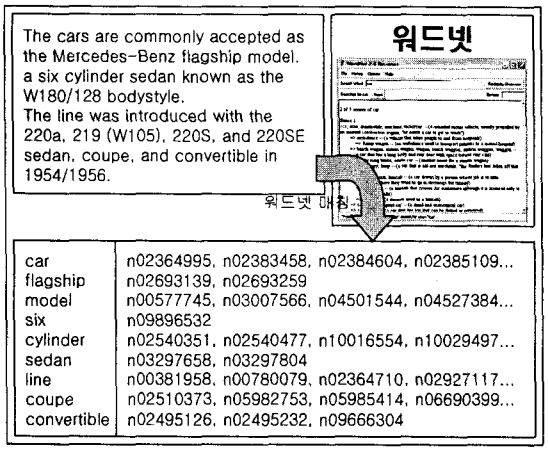
3.1 문서의 계층화 방법

문서를 의미적으로 분류하기 위해 우선 문서가 포함하는 내용을 계층화 한다. 문서를 계층화하기 위해서는 문서에서 명사를 추출하고, 각 명사의 정확한 의미를 파악한 후, 파악된 의미를 이용하여 계층구조를 형성하는 단계가 요구된다.

3.1.1 문서내에서 명사 추출

어떤 특정한 주제를 위해 기술된 문서는 단어들 사이에 의미적인 관계가 존재한다. 이러한 문서는 명사, 동사, 형용사 등의 여러 품사들의 단어들로 구성이 될 수 있는데, 특히 명사들은 문서를 구성하는 핵심 기술자라 할 수 있다. 이에 문서를 계층화하기 위해 첫 번째로 문서 내에 포함된 명사들을 추출하는 단계를 수행한다.

앞에서 설명한 워드넷에는 명사, 형용사, 동사, 부사 등을 기술하고 있다. 이에 본 연구에서는 특정한 문서를 입력으로 받고, 문서 내의 각 단어를 워드넷의 명사 데이터베이스와 매칭을 시켜 일치하는 명사의 SYNSET_ID(의미)들을 얻는다. 하나의 명사는 여러개의 의미를 가질 수 있는데, 정확한 의미는 이후 과정에서 파악하고 현 과정에서는 명사가 갖고 있는 모든 의미를 찾는다. 본 내용은 [그림 4]와 같다.



[그림 4] 명사 추출

3.1.2 명사의 의미파악

여러 가지 의미를 갖고 있는 단어의 정확한 의미를 파악하는 것은 쉽지 않다. 이를 위해 WSD(Word Sense Disambiguation)라는 한 연구 분야가 생성되어 연구가 진행되고 있다. 특히, 워드넷에 기반한 것으로는 Adapted LESK Algorithm이 있다.[7,10] Adapted LESK Algorithm은 워드넷내에 기술되어 있는 모든 관계와 개념들의 정의를 최대한 활용하여 단어의 의미를 파악하는 것으로써, 현재까지 가장 신뢰성 있는 연구라 할 수 있다. 본 논문에서 추출된 명사들의 정확한 의미를 파악하기 위해, 이러한 Adapted LESK Algorithm을 문서 내의 명사들의 정확한 SYNSET_ID를 얻는데 적용하였다.

3.1.3 SYNSET_ID 기반 문서 계층구조 파악

워드넷에는 단어들의 관계를 정의하기 위해 각종 심볼(Symbol)을 이용하고 있다. 이러한 심볼들은 [표 2]와 같이 SYNSET_ID, TARGET_ID을 이용하여 단어들의 관계를 정의하고 있다. [표 2]는 워드넷 데이터베이스 내에 작성된 Pointers 테이블에 작성된 내용의 일부를 보여주고 있다.

[표 2] 워드넷 데이터베이스의 일부

PTR	SYNSET_ID			TARGET_ID		
~	n02392911	n	0	n02311742	n	0
~	n02392911	n	0	n02946569	n	0
~	n02392911	n	0	n02946676	n	0
~	n02392911	n	0	n03346295	n	0
@	n02393107	n	0	n03569523	n	0
@	n02393264	n	0	n03055972	n	0
@	n02393349	n	0	n02236345	n	0
@	n02393457	n	0	n02230661	n	0
%p	n02393457	n	0	n02792702	n	0
@	n02393577	n	0	n03293673	n	0
...	...	...	...	...	...	...

앞의 과정에서 얻어진 모든 SYNSET_ID들의 의미적인 계층구조를 파악하기 위하여, 워드넷 데이터베이스에 접근을 통해, 워드넷 내에 정의된 SYNSET_ID들의 관계와 매칭을 이용하여 [그림 5]와 같이 SYNSET_ID들의 계층구조를 생성할 수 있다.

3.1.4 계층 구조를 RDF(S) 문서로 작성

작성된 계층 구조는 추출된 SYNSET_ID의 보존과 문서비교의 용이성을 위해 W3C에 의해 2004년 Semantic Web 표준으로 작성된 RDF(Resource Description Framework)문서로 작성한다. 이렇게 작성된 RDF(S) 문서는 웹 문서의 메타데이터 역할을 하고, 실제 사용자가 선택한 문서와 유사한 문서를 검색하고자 할 때 유사도 측정을 위한 데이터로 사용된다. [표 3]은 위의 과정에서 추출된 명사들의 SYNSET_ID들을 RDF(S) 문서로 작성한 것이다.

[표 3] SYNSET_ID들의 RDF(S) 문서

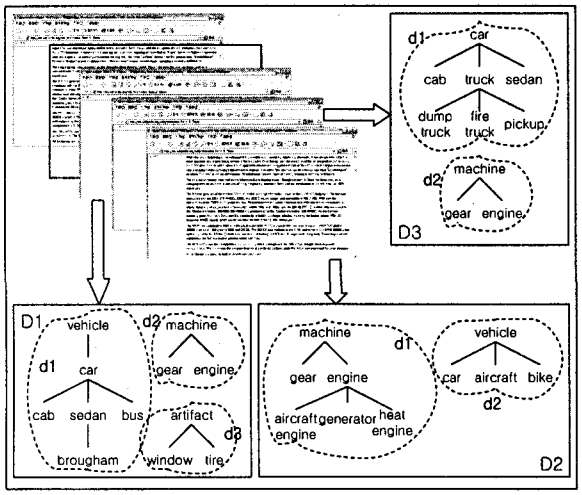
```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xml:base="http://vector.chosun.ac.kr/doc/vehicles">

  <rdfs:Class rdf:ID="n04348422"/>
  <rdfs:Class rdf:ID="n02853224">
    <rdfs:subClassOf rdf:resource="#n04348422"/>
  </rdfs:Class>
  <rdfs:Class rdf:ID="n04008331">
    <rdfs:subClassOf rdf:resource="#n02853224"/>
  </rdfs:Class>
  <rdfs:Class rdf:ID="n03006338">
    <rdfs:subClassOf rdf:resource="#n02853224"/>
  </rdfs:Class>
...
```

n04348422 : vehicle	n02853224 : car
n04008331 : sedan	n03006338 : coupe

3.2 계층 비교 방법

문서의 의미적인 유사도 측정을 위해, 3.1의 과정에서 작성한 계층구조를 비교하기 위한 알고리즘을 작성하였다. 구축된 각 문서의 계층구조는 [그림 5]와 같이 몇 개의 도메인으로 구성된다.



[그림 5] 계층 표현

특정한 주제를 가지고 기술된 문서는 주제를 포함하는 도메인에 개념들이 집중되어 있고, 주변의 도메인들은 핵심이 되는 도메인의 설명을 돕는 속성요소로써의 역할을 하고 있다. 이에 본 계층 비교 방법에서는 문서에서 기술하고 있는 각 도메인별 비중을 구하고, 각 도메인에서 측정한 개념들을 Jaccard-Similarity를 이용해서 일치도를 구한 후, 문서들의 유사도를 계산한다. [그림 5]에서 몇 개의 도메인들이 각 문서에 포함되는데, 각 도메인이 문서에서 차지하는 비중을 구하기 위해 [수식 1]을 적용하였다.

$$Domain-Weight=\frac{Concepts\,in\,Domain}{Concepts\,in\,Document} \quad \text{.....(1)}$$

[수식 1]의 도메인 비중(Domain-Weight)은 전체 계층에 포함된 개념들 중 해당 도메인에 포함된 개념들의 수를 구하는 것이다. [수식 1]을 이용하여 각 문서에 포함된 도메인들의 비중을 구한 후, Jaccard-Similarity를 이용하여 각 문서의 동일 도메인에 포함된 개념들의 일치도를 구한다. [수식 2]는 Jaccard-Similarity를 수식이다.

$$Jaccard-similarity(c_1,c_2)=\frac{P(c_1\cap c_2)}{P(c_1\cup c_2)} \quad \text{.....(2)}$$

Jaccard 수식은 최소 0과 최대 1 사이의 값을 갖으며, 0은 두 도메인이 서로 전혀 연관이 없음을 의미하고, 1은 두 도메인이 서로 일치함을 나타낸다.[8]

위의 두 수식을 적용하여 구해진 값을 이용하여, 문서의 유사도를 측정하는 식은 동일 개념을 기술하는 도메인의 비중과 일치하는 개념 비율의 곱으로 구할 수 있다. [수식 3]은 문서의 유사도를 측정하는 수식이다.

$$Document\ Weight=\sum(D_sW\times dCr+D_tW\times dCr)/2$$

.....(3)

위 수식에서 D_sW는 원본 문서의 도메인 비중을 나타내고, D_tW는 비교하고자 하는 문서의 도메인 비중이며, dCr은 Jaccard-Similarity로 측정한 도메인 내의 개념 일치도를 의미한다. [수식 3]으로 얻어진 값에서 1은 두 문서가 일치함을 나타내고, 0은 전혀 다른 문서임을 나타낸다.

위의 수식들을 이용하여 [그림 5]에 표현된 각 문서의 계층들 중 D1을 중심으로 유사도를 측정한 결과를 [표 3]에서 보이고 있다.

[표 3] [그림 5]의 유사도 구하는 예

문서	D1			D2		D3	
도메인	d1	d2	d3	d1	d2	d1	d2
도메인 비중	0.5	0.25	0.25	0.6	0.4	0.7	0.3
Jaccard-Similarity							
D1과 D2				D1과 D3			
(D1-d1):(D2-d2)		(D1-d2):(D2-d1)		(D1-d1):(D3-d1)		(D1-d2:D3-d2)	
2/8=0.25		3/6=0.5		3/10=0.3		3/3=1	
문서 유사도							
D1과 D2				D1과 D3			
(0.5*0.25+0.4*0.25+0.25*0.5+0.6*0.5)/2=0.325				(0.5*0.3+0.7*0.3+0.25*1+0.3*1)/2=0.475			

[표 3]에서 문서 D1, D2, D3의 각 도메인의 비중을 [수식 1]을 이용하여 구하고, Jaccard-Similarity를 적용하여 개념 일치도를 구한다. 문서 D1을 중심으로 D2와 D3의 유사한 정도를 파악한 결과 D1은 D3과의 유사도 값 0.475에 기반하여 D2보다 유사한 것을 알 수 있다.

4. 결론 및 향후 연구방향

본 논문은 사용자가 원하는 문서와 유사한 문서를 찾아주기 위해 문서들을 의미적으로 비교하여 유사도를 구하기 위한 연구이다. 문서들의 의미적인 비교를 위해 각 문서들을 워드넷과 매칭하여 명사를 추출, Adapted-LESK 알고리즘을 적용한 SYNSET_ID를 파악, 파악된 SYNSET_ID를 통해 계층으로 표현하였다. 이러한 과정으로 추출된 각 문서들의 계층들은 도메인 비중 구하는 [수식 1], 동일 도메인 내의 개념 일치도를 구하는 Jaccard-Similarity [수식 2], 그리고 도메인 비중과 개념 일치도를 이용하여 문서의 유사도를 구하는 [수식 3]을 통해 최종적으로 문서들 사이의 유사도를 측정할 수 있었다. 본 연구는 문서의 특징을 정확하게 반영할 수 없고 의미적이지 않은 단점을 가진 기존연구에 비해, 문서의 내용을 의미적인 계층으로 표현하고 중요 도메인에 가중치를 부여함으로써 좀더 의미적이라고 할 수 있다. 본 연구의 실험과 평가는 아직 진행되지 않았으며, 이는 향후 방향 및 목표이다.

감사의 글

본 연구는 문화관광부 및 한국문화콘텐츠진흥원의 문화콘텐츠기술연구소(CT)육성사업의 연구결과로 수행되었음.

참고 문헌

[1] D.D.Lewis, "Naive(Bayes) at forty : The Independence Assumption in Information Retrieval," In European Conference on Machine Learning, 1998

[2] J. McMahon and F. Smith, "Improving statistical language model performance with automatically generated word hierarchies," Computational Linguistics, Vol.22, No.2, 1995.

[3] T.Joachims, "A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization," ICML-97, 1997.

[4] 한광록, 선복근, 한상태, 임기욱, "인터넷 문서 자동 분류 시스템 개발에 관한 연구", 제9회 한국정보처리학회 논문집, 제7권 제9호, pp.2867-2875, 2000

[5] A. McCallum and K. Nigram, "A Comparsion of Event Models for Naive Bayes Text Classification," AAAI-98 Workshop on Learning for Text Categorization, 1998

[6] 고수정, 이정현, "Apriori-Genetic 알고리즘을 이용한 베이지안 자동 문서 분류", 정보처리학회 논문지 B, Vol.01, No.01, p.001~012, 2001년 6월

[7] Satanjeev Banerjee, Ted Pedersen, "An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet", Computational Linguistics and Intelligent Text Processing: Third International Conference, p.136-147, Vol.2276, February 17-23, 2002.

[8] Hyunjang Kong, M.G. Hwang, P.K. Kim, "A New Methodology for Merging the Heterogeneous Domain Ontologies based on the WordNet", International Conference on Next Generation Web Services Practices, 2005. 08.

[9] http://wordnet.princeton.edu/

[10] S. Banerjee, T. Pedersen, "An adapted Lesk algorithm for word sense disambiguation using WordNet," In Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics, Mexico City, pp. 136-145, 2002