토 론

사회 : 감사합니다. 최근에 들어서 수년동안 언어학 분야와 전산학 분야의 공동연구 내지 작업이 이루어지는 가운데, 1986년도 한국과학재단의 지원을 받아 '목적기초연구'라고 하는 정책연구 중에 '자연언어처리의 기초연구'라는 3년계획의 연구과제가 주어진 바 있습니다. 그 과제를 마침 이 자리에 계시는 서울대 언어학과의 장석진 선생님께서 맡아 오셨기 때문에, 연구의 목적이나, 진행과정에 대해서 듣는 것으로 이 토론회를 시작하도록 하겠습니다.

장석진 : 지금 이정민 사회께서 이야기하신 대로, 과학재단에서 후원하는 목적기초 연구과제로서, 1986년에 시작했고 이제 3차년도에 들어가고 있습니다. '목적기초과제'라는 것은 '특정과제'와 대조되는 말입니다. 즉 특정과제가 가깝게 산업과 직결된다고 하면, 목적기초는 그러한 의미에서는 basic하고 학구적인 것이지요.

이 계획에는 인문분야와 서울대 공대, 과학원 등이 관련되어 있는데, 이처럼, 전산쪽과 인문쪽이 같이 합칠 수 있었다는 점에서 학제간의 연구였고, 바람직한 것이 아니었나 하고 생각됩니다. 또한 학문분야에서뿐만 아니라, 학교로 보아서도, 오늘 의미관계를 말씀해 주셨던 고려대의 이기용 교수를 비롯하여 과학원의 최기선 교수, 부산대의 권혁철 교수 등 여러분이 참석하셔서 매우 다채로웠었습니다.

세 분야로 나누어서 시작했는데, 제 1세부는 언어이론에 중점을 두고 있으며, 저와 이기용 교수, 유수선 선생이 참여하고 있고, 제 2세부는 국어문법의 구체적 실천이랄까…자료를 다루는 분야로서, 지금 사회이신 이정민 교수가 맡았었으나, UCLA에 가시는 바람에 신수송 교수, 임홍빈 교수가 참여하고 있습니다. 제 3세부는 전산과에서 김영택 교수와 그 팀이 맡고 있는데, 이론과 구체적 데이타를 마지막 구현시키는 시스템을 만듭니다. 그래서 결국 최종적으로 기계화에 접근시키는 과정이라고 할 수 있겠습니다.

각 세부에서 연관성있게 할 수 있도록 하기 위해서 lexicon이나 통사·의미, 특히 의미는 상황의미론뿐만 아니라, 배경지식, 지식의 전달에 이르기까지, 폭넓게 다루고 있습니다. 결국 언어이론에서 다룰 수 있는, 그러면서 자연언어처리에 필요한 사전, 문법, 그리고 의미까지를 폭넓게 생각해 보았다고 하겠습니다. 국어의 구체적 자료에 관하여는 우선 용언의 동사를 중심으로 한 하위범주화에 의해 세부적으로 나누어 보았고, 의미역에 대한 문제 등을 포함하여 국어문법의 모형 비슷한 것을 만들고 있습니다. 거기에 지금껏 반영된 이론으로는, 오늘 계속 이야기되고 있지만, unification-based grammar, 즉 통합기반문법을 추구하였는데, GPSG, LFG, HPSG, Categorial Grammar, 상황의미론 등이 이론적 바탕이 되었습니다. 그러한 입장에서 제 3세부의 전산과 윤덕호군이 LFG에 입각하여 일종의 parser라 할 수 있는 Korean

Syntax Analyser(KSA)라는 모형을 그 연구의 일부로서 발표하였고, 앞으로는 이것을 정리하여 깊이있게 해 볼 생각입니다.

사전부분에 있어서는, 우리가 이론중립적(theory-neutral)이라는 말로서 내세우고 있는데, 어떤 이론에라도 적용될 수 있는 그러한 모형을 추구하고 있습니다. 아까 최기선 교수가 이야기하셨지만, 현재로서는 자연언어처리나 기계번역에서 가장 중요한 과정 중의 하나인 machine-readable한 사전, 즉 computer를 쓰는 사람들이 이용할 수 있는 그러한 database가 없는 실정입니다. 그래서 Hornby나 Longman Dictionary에 이미 되어 있는 문법의 코드, 즉, 하위범주화의 frame들을 옮겨보는 그러한 과정이라 할 수 있는데, 우리 국어에는 아직 그러한 사전이 없으니까 그러한 입장에서 하위범주화 해나가는 작업이 필요하다고 할 수 있겠습니다. 결국 LFG, HPSG, KPSG와 같은 모형을 가지고 사전에 들어가는 작업, 그리고 상황이론적인 배경지식에 대한 연구 등이 마지막 남은 문제라 할 수 있습니다. 현재로서는 작년도에 이미 학위논문으로 나와 있는 권혁철 선생의 추론의 체계가 있고, 이기용 선생을 중심으로 하여 상황의미론이 정립되지 않을까 하는 이러한 단계에 있습니다.

남은 기간 동안에는, 목적기초이니까, Stanford의 CSLI에서 진행되고 있는 것을 가장 빨리 이 안에서 흡수하여, 한국어 자연언어처리에 도움이 될 수 있는 방향으로 나아가야 하지 않을까 생각하고 있습니다.

사회 : 감사합니다. 그러면 여기서 floor에 계신 분의 질문을 받도록 하겠습니다. 앞에 주제발표하신 세 분의 논지나 방금 장식진 선생님의 소개에 대해 질문이 있으시면 이야기해 주십시요.

고영근 : 김한곤 선생님께 말씀드리겠습니다. 아까 발표하실 때에 written language를 '필사어'라고 하고 spoken data를 '구어'라 하셨는데, 각각 '서사어'와 '구두어'로 하면 어떨까 하는 것이 제 자신의 의견입니다.

김한곤 : 술어는 어떻게 바꾸셔도 좋습니다.

고영근 : 예, 알겠습니다. 제가 질문하고 싶은 것은 아까 '필사어의 표기상 난맥상'이라 하셨는데, 우리 맞춤법의 역사를 보면 두가지가 있습니다. 세종대왕 때는 음소적인 체계를 썼고, 지금은 형태음소적인 표기법을 쓰고 있는데, 이중 어떤 표기법을 지칭하시는 것인지…

김한곤 : 그 어느 것도 아닙니다. 말씀을 듣고 보니, 제가 표현을 잘못 썼구나 하는 느낌이 듭니다. 제가 말한 것은 어느 표기법을 쓰든지 일반인들이 관습적으로 너무 표준을 안 지키는 습관이 있다는 의미입니다. 제가 아까 설명할 때, 시간에 쫓겨 내용을 너무 압축해 버렸는데, 심지어는 저를 포함하여 대학 교수의 글도, 그러니까 글을 쓴다는 모든 사람들의 글도 computer에 input해 보면 단어나, 토씨 쓰는 법, 띄어쓰기 등이 제각기여서 문제라는 것입니다. 영어는 단어의 경계가 분명하지 않습니까? 예컨대 국어는 복합어를 붙여 쓸 것이냐 띄어 쓸 것이냐 하는 문제에 대해서도 자기 멋대로인 경우가 많습니다. 그러니까, 국어선생님들께서 표준을 가지고 이것이 표준이다 라고 가르치시지만, 실질적으로 많이 지켜지지 않는다는 의미에

서이지, 표기라 하여 무슨 표기법이 나쁘다는 의미는 아닙니다. 제가 표현을 잘못 사용하여 선생님께 오해를 드린 것 같습니다. 양해하시겠지요?

사회 : 결국 '사용상의 난맥상'을 뜻하는 것으로 이해하면 되지 않을까 생각됩니다. 다른 질문이 있으시면 말씀해 주십시요.

이익환 : 이기용 선생님께 한 가지 말씀드리겠습니다. 그러니까 기계가 '모른다'는 말은 하지 말아야 한다는 말씀이시지요?

이기용 : 항상 모른다고만 하면 안되지만, 모를 때는 모른다고 해야지요.

이익환 : 그런데 우리가 의미론, 화용론을 이야기하고, 또 거기서 '의문의 논리'를 이야기하는데, 가장 이상적인 context 속에서 의문이 오갈 때에는 저쪽에서 모른다는 대답이 나오지 않도록 되어야 가장 이상적인 의문이 되거든요.

이기용 : 아니요, 그렇지 않습니다. 왜냐하면, 우리가 사람이든 기계이든 지식이라는 것은 finite하고, incomplete합니다. complete하다고 전제를 하면 기계를 만들 수 없습니다. 그래서…

이익환 : 그러니까, 제가 이제 질문을 하겠습니다. 예를 들어서 기계에 모든 background 지식, common sense,… 이런 것을 모두 넣어 준다…

이기용 : 아니죠. 그러면 안됩니다. common sense도 다 넣어주고, … 우리가 전지전능한 기계를 만들려고 하는 것은 아닙니다. 우리가 필요로 하는 기계가 어떤 유의 것인지를 알아야 합니다. 기계를 이용하여 무엇을 하겠느냐 하는 것이죠. 기계가 해야할 minimum task가 무엇이겠느냐를 결정해야 합니다. 가령 이 task가 전문화될 수도 있습니다. 전문화되어서 어떤 분야에 대해서만 알고 다른 분야에 대해서는 모른다고 할 수도 있죠. 그러니까 모를 때에는 모른다고 해야 하죠. 절대로 전지전능한 기계를 만들려고 하는 것이 아닙니다. 그런 기계는 만들 수도 없구요.

이익환 : 제 말씀은 전지전능하다는 의미에서, 그러니까. 예를들면, 어떤 화맥에서 전혀 다른 질문을 해도 꼭 대답해야 한다는 그런 뜻은 아니구요, 대화가 가장 이상적으로 이루어질 때에는 상대방이 내 질문에 대해 적어도 대답할 수 있다고 믿든가 알든가 했을 때, 질문하게 됩니다. 그러한 background나 common sense가 있는 상태에서 질문이 전해질 경우 상대방이 만약에 모르겠다고 대답하면, 우리가 다 알다시피, 질문의 논리에서는 상대방의 질문자체에 대한 denial이지, 질문에 대한 대답이 아니거든요. 그러니까, 기계가 만약에 모르겠다고 하면, 그것은 상황판단을 못하는 기계가 되지 않겠는가 하는 생각입니다. 선생님께서는 아까 상황을 판단하는 기계가 된다고 하셨는데…

이기용 : 네, 그런데 이 기계는 natural language processing을 하는 기계입니다. 이 기계에게 도서관에 이 책이 있느냐고 질문을 던졌다면, 일러주기 전에는 모르지요. database에 그런 information이 들어가 있는 게 아니예요. natural language processing은 다른 기능을 갖는 기계입니다. 아무 information을 주지 않고 질문에 대한 대답을 얻으려면 다른 기계에 물어야 합니다. 그러니까, 이 기계는 주변상황에 대해서는 거의 백지상태이고, 오직 말을 알아듣는 기계라고 생각하시면

됩니다. 언어를 해석할 수 있는 최소의 지식만 있는 것이고, 다른 지식은 discourse가 들어 올 때에만 존재하는 것입니다. 그리고 그 discourse가 사라지고 어느 기간이 되면 memory가 또 지워져야 된다고 생각합니다.

사회 : 그것으로 답변이 되었다고 간주하겠습니다.

권혁철 : 제가 그 분야에 대해서 전산처리의 경험이 조금 있어서 잠깐 말씀드리겠습니다. 실제 indirect speech act 처리는 매우 어렵습니다. computer에서는 어떤 특별한 technique을 사용하는 것이 아니라 Austin이나 Searl의 이론을 도입해서, discourse의 목적이 무엇이냐, 그 사람의 plan이 무엇이냐 하는 것을 이용하여 처리합니다. 결국 이 문제는 인공지능에서 여러 로보트들이 합동하여 어떤 일을 처리하는 문제와 일치합니다. 심리학이나 철학에서 연구된 개념을 도입하여 언어를 처리하면, 어렵기는 하지만 부분적으로는 화용문제의 처리가 가능합니다. 제가 만들어본 system도 근본적으로 화용론 자체를 대화참여자의 plan과 goal을 이용하여 처리하며, 그것에 의해서 indirect speech act도 상당한 수준에서 처리할 수 있습니다. 물론 현실적으로는 어느 정도 장애가 있지요.

이기용 : 제가 한 마디만 더 하겠습니다. 지식의 차이가 어디에 있는가 하면, 가령 Joe가 Koe한테 'Kim loves Jim?'이라고 했다고 합시다. 그랬다면 이 기계는 Kim이 누구인지 Jim이 누구인지 알 필요는 없습니다. 그러나 이건 알아야 합니다. 즉, 그 기계가 그 말을 들었을 때, Joe가 Kim이 누구이고 Jim이 누구인지 아느냐 하는 질문을 받았다면 '네'라고 대답할 수 있어야 합니다. 그런 차이입니다.

이익환 : 제 말씀은 기계한테 질문을 하는 사람이, 적어도 그 기계가 이 질문에 대하여 대답할 수 있다는 지식이 있을 때 질문하는 것이 가장 정상적인 질문이 된다는 이야기입니다. 따라서, 질문자가 가장 정상적인 화맥에서 질문을 했을 때에는, 기계에서 모른다는 대답이 나올 수 없다는 것입니다.

사회 : presupposition failure 같은 것도 있을 수 있으니까 모든 상황을 고려하기는 해야 할 것으로 생각됩니다. 그러면 이 문제는 이 정도로 그치기로 하고, 다른 질문이 없으시면, 경희대 박병수 선생님께서 자연언어처리와 관련된 문법모형의 선택에 있어서 기준을 어떻게 두어야 할 것인지 이론적 측면과 응용적 측면에 관하여 말씀해 주시면 고맙겠습니다.

박병수 : 언어학에서 많은 문법모형들이 개발되어 서로 경쟁하고 있는데, 자연언어처리를 하는 전산학적 입장에서 볼 때, 이 중 어느 것을 택할 것인가 하는 기준을 이야기해 보라는 주문이신데, 제가 이 분야에 관하여 특별히 내세울 만한 그런 연구를 하지는 못하였습니다. 그래서 제가 새운 것은 아니지만, 지금까지 해왔던 것을 정리해서 서너 가지의 기준을 들어 보도록 하겠습니다.

첫째로, 소위 Chomsky hierarchy라고 알려져 있는데, 적어도 어떠한 type이 되어야 할 것이냐 하는 기준을 들 수 있겠습니다. type 0, type 1, type 2, type 3까지 대개 4가지 type을 이야기 하는데, 지금까지는, type 0나 type 3 처럼 극단적이어서는 안되고, type 1이나 type 2

근처에 있는 것일 것이라고 결론이 내려져 있습니다. 예를들어 표준이론의 변형문법은 type 0에 가까운 것으로, 여기에 맞지 않아 버렸으며, 최근의 GB 이론은 rule-based formalism에서 principle-based formalism으로 점차 바뀌어 가고 있는 상태이기 때문에 type 2나 type 1에 가까운 문법 formalism이라고 말하는 사람도 있습니다. GPSG 주창자인 Gazdar는 GB이론을 완전히 낙제라고 혹평하였지만, Pollard 같은 사람은 최근에 전적으로 그렇지는 않다고 이야기하고 있지요.

둘째로는, information-based theory가 좋은 것이라고 이야기하고 있습니다. 그러니까, 어떤 표현이 가지고 있는 information을 잘 기술할 수 있어야 한다는 것이죠. LFG나 HPSG, GPSG, CG와 같은 전산언어학에서 이용하고 있는 model들이 거기에 맞는 정보 중심의 이론들입니다.

셋째로는, 이러한 정보를 잘 나타내는 방법이 무엇인가 하는 것인데, 그 방법으로서 자질 중심이어야 한다고 흔히 말하고 있습니다. 즉 속성-속성가의 matrix로 표현할 수 있어서 어떠한 종류의 information이라도, 즉 phonology, syntax, semantics 심지어는 pragmatics까지도 잘 표현할 수 있는 이론이어야 한다는 것입니다.

대충, 이상 언급한 선택기준을 세워놓고 있다고 생각됩니다. 앞에서 여러 선생님들께서 이미 지적하셨지만, 이러한 기준에 합격하는 이론을 통합기반문법(unification-based grammar)이라고 부르고, 대개 이러한 방향으로 나아가고 있다고 할 수 있습니다.

사회 : 감사합니다. 그러면 이어서 한국 전자통신연구소에 계시는 정희성 박사께서 문법모형 가운데 phonology, morphology, syntax, semantics의 각 level의 언어학적 연구가 자연언어처리에 어떻게 기여하면 좋겠는지 하는 문제에 대하여 말씀해주시면 고맙겠습니다.

정희성 : 저희는 지금 통신연구소에서 한국구문이해 system '나랏말씀'을 이론부터 implementation에 이르기까지 진행중에 있습니다. 순수언어학에 대한 깊은 이해가 없어서 저희들 스스로 소위 unification-based grammar, 예를들어 JPSG나 HPSG, GPSG에서 보이는 보편성있는 이론들을 도입하여 우리 말과 글의 현상을 풀어보고자 정밀화 작업을 하고, 거기에 따라서 각각의 implementation을 시도해 보고 있는 실정입니다.

오늘 주제가 자연언어처리의 이론과 방법인데, 최기선 교수님께서도 말씀하셨지만, 자연언어처리의 원리는 있느냐?… 없을 것이고 실제라는 것은 모두 장난감이다 라고 하여도 옳은 이야기가 될 수 있겠습니다. 따라서 저희들이 추구하는 바는 최소한 이론언어학에 대한 지견은 얻어야겠다는 것입니다. 이론언어학의 어느 범주의 것은 아주 철학적인 것도 많이 있습니다. 이러한 이론들을 과연 computer에 태울 수 있느냐 하는 것이 문제인데, 저희들이 보기에는 computational model이 거의 존재하지 않는 것으로 생각됩니다. 저희들의 입장으로서는, GB라든가, 그밖의 unification-based grammar에 있어서 문법의 우월론을 따지고 싶지는 않고, 우선 computer에 태우고 싶다는 것이지요. 국내에서도 이론언어학이나 국어학을 하시는 분이 굉장히 많이 계시고, 해마다 많은 이론들이 수없이 나오고 있는데, 저희들이 이용할 수 있는,

그러니까, 어떤 algorithm을 찾을 수 있는 이론이 얼마나 있느냐에 대해서는 회의적입니다.

현재의 자연언어처리 기술은 전부 ad hoc하다고 할 수 있습니다. 그래서 하나의 제안이라고 할 수 있지만, 좀더 이론을 세우고, 거기에 따른 계산 model을 세우고, algorithm을 찾아, 거기에 대한 implementation을 하여야 하지 않을까 생각합니다. 단 우리가 노리는 바는 computer에 우리의 말과 글을 이해하는 system을 만들어 통일적 조작성이 가능하도록 하는 것입니다. 예를들어 대학교 4학년까지 나와서도 computer를 제대로 쓰지 못하는데, 그 이유는 program languge, hardware, 그리고 operating system 등 수많은 영역을 배워야 하기 때문입니다. 만일 computer와 인간사이의 interface가 자연언어에 의해 이루어진다면, 남녀노소가 쉽게 쓸 수 있겠지요. 그렇게 함으로써, 한국의 노동생산력이 높아지고, 지적공간도 확대되지 않겠느냐 하는 생각입니다.

조금전 최기선박사께서 '감기는', '차는'이라는 예를 들어서 우리말은 단어의 경계라든가 그밖에 상당히 어려운 부분이 많다고 하셨는데, 저희들로서는 관점을 약간 달리하고 있습니다. '감기는'이 '감기'(명사)+'-는'(topicalizer)이냐 아니면 관형사형이냐 하는 것인데, 다음에 어떤 성분이 오느냐에 따라 (즉, 술부가 오느냐, 명사가 오느냐에 따라) 확률상으로 인식할 수 있다고 봅니다. 따라서, 보다 근본적인 문제는 우리말 데이타 중에서, 예를들어, '물이 마시고 싶다'는 문법적인 문장인데, '물이 집에서 마시고 싶다'는 비문이 되는 이러한 현상을 어떤 규칙으로 형식화하느냐 하는 것입니다. 또, '서울까지 갔으면서 63빌딩을 보지 않고 돌아왔다', '신문을 보면서 밥을 먹는다'의 '-면서'가 형태는 똑 같지만, 앞의 경우는 접속문을 만들고, 뒤의 경우는 병렬문을 만드는데, 이러한 현상을 syntax에서 어떻게 다룰 것이냐 하는 것이 문제입니다. 그리고 요즈음 언어학에서는 divide and conquer형식으로 제일 밑에서부터 phonology, morphology, syntax, semantics, pragmatics로 올라가는데, 과연 그와 같은 model이 바르냐 하는 점도 문제입니다. 앞의 경우와 같은 문장을 인식할 때는 분명히 어떤 형식의 semantics를 거쳐가야 하는데, 그것이 lexicon level에서의 semantics가 될지 아니면 그렇지 않을지는 아직 잘모르겠습니다만, 대단히 어려운 문제입니다.

그밖에 '…의 …의'로 연결된 복합명사 'A사의 종업원의 수'와 '이달의 A사의 매상고'에 있어서 parsing tree가 서로 다른데, 이를 단순히 명사로만 처리할 수는 없을 것으로 생각되고, part of speech도 엄밀히 나눌 필요가 있다고 봅니다.

아까 김한곤 선생님께서도 우리 말의 morphology가 굉장히 힘들다고 지적하셨는데, LFG나, HPSG에 의한 우리말의 분석은 introduction 단계에 있으며, 또, 그러한 문법이론들이 우리말에 사용되리라고는 생각지 않습니다. 그만큼 우리말이 단순하지 않고, 특히 morphology는 어렵습니다. 여태까지의 morphology는 syntax의 아래에 있었는데, aspect와 tense는 sentence 밖으로 나와야 되고, 이것을 처리하지 못하면 speech act가 되

지 않습니다. 예를들어 경어법 표현 같은 것도 전부 speech act에 속하는 것인데, 이것들을 sentence 안에 묶어버리면 처리가 되지 않지요. 최소한 top-down에 언어학, 인지과학에서 말하는 개념을 넣고, 우리말의 현상을 bottom-up으로 풀어감으로써 중간에서 만나는 접점이 분명히 있다고 봅니다. 그래서 저희들 computer하는 사람들과 순수언어학하는 사람들이 협조하여 뭔가 계산언어학 - 양쪽에서 보면 거리감이 있을테니까 이를 적절히 조합시킬 수 있는 새로운 학문영역으로서의 계산언어학 - 같은 것을 시도해 보아야 할 필요가 있다고 말씀드리고 싶습니다.

한가지 오해가 있을 것 같아 말씀드리겠는데, 흔히 computational linguistics를 '전산언어학'이라고 하는데, 이 전산언어학은 전산학을 하는 사람들이 언어학에 흥미가 있어 제시한 것은 아닙니다. computer가 있기 이전에 이미 computability라는 개념이 있습니다. 여러분도 다 잘 아시겠지만, 이론언어학에서 경험적 타당성, 기술적 타당성, 설명적 타당성이 있듯이 computability라는 것도 있지요. 즉 computer 상에서 어떤 문법이론이 좋으냐 하는 것입니다. 박교수님께서도 아까 말씀하셨지만, context-free 문법이 좋다고 할 수 있습니다. context-free 이상이 되면 현재까지는 algorithm이 존재하지 않고 혹시 algorithm이 있다 하여도 최악의 경우 지수함수로서 발산해버려서, halt가 안되기 때문입니다.

사회 : 네, 감사합니다. 결국 언어학 내지 한국어학하는 사람들에 대해 불만이 상당히 많으시다는 결론이군요. 앞으로 언어학하시는 분들께서 분발해야 되겠습니다. 그러면 부산대 전산학과의 권혁철 선생께서 방법론에 있어서의 linguistic tool과 computational tool의 차이문제에 대해 특히 lexicon에서의 어려움과 관련하여 말씀해주시겠습니다.

권혁철 : 먼저, 자연언어처리는 lexical analysis를 하고, syntax, . semantics와 pragmatics를 처리하게 되는데, 전체적으로 GB이론을 비롯하여 언어학에서 제시된 대부분의 이론들이 computability를 고려할 때 computer 처리가 불가능한 formalism들 입니다. 예를들어 GPSG를 보면, GPSG가 context-free이기 때문에 computer처리가 가능할 것이라 생각했는데, 그것마저 오해였습니다. 언어학하시는 분들이 제시하는 이론들이 computer에 의한 처리가 불가능하고, 그러다보니까, 자연언어를 처리하기 위해서는 그 이론에다 추가로 heuristics라든지 아니면 다른 무언가가 보완되어야 했습니다. 이러한 것을 찾는 과정에서 computer하는 사람들은 이론보다는 case by case로 문제를 해결하려고 시도하는 경우도 종종 있었는데, 바로 여기에 또다른 문제점이 있다고 할 수 있습니다. 그래서 대부분 syntax 처리를 위한 이론 자체가 computer처리에 많은 어려움을 제공하고 있는 만큼, syntax 보다는 lexicon에 관점을 두고, syntax에서 다루는 양을 대폭 줄여 context-free grammar에 가깝게 하고, 나머지 문제들은 semantics나 다른 부분에서 다루게 하는 것이 현재 전산언어학이라 불리우는 분야의 전체적 경향이라 할 수 있겠습니다.

그러면, 각 분야에 대해서 살펴보기로 하죠. 앞에서 말한 것처럼, syntax에서 다루던 것들을 다른 부분으로 보내다 보

니까, 결국 lexicon을 어떻게 유지하느냐가 문제가 되었습니다. 실제 우리가 사용하는 사전에 있는 단어가 10만이상이 되기 때문에, 이들을 모두 computer처리에 적합한 형태로 써넣는다는 작업자체가 불가능합니다. 그래서, 예를들면, Longman Dictionary of Contemporary English는 2,000여 단어를 key word로 하고 있다는데, 이와 같은 제한된 중요한 단어를 중심으로 meaning postulate나 semantic decomposition 등을 이용하여, computer가 읽어서 사용할 수 있는 형태로 바꾸는 것에 관한 연구가 이루어지고 있고, 또한 그러한 system을 어떻게 만들 것이냐에 관한 이론이 많이 연구되고 있습니다. 그러나 아직 확실한 방법이 제시된 것은 아닙니다. 이것이 결국 machine-readable lexicon의 문제인데, 조금전에 최기선 선생님께서도 이야기하셨지만, 자연언어처리에 필요한 어휘항목은 방대하여 이것이 기계가 읽을 수 있는 사전으로 되어 있지 않다면, computer가 어떠한 tool을 이용하여 다른 자료로부터 이 항목들을 자신이 이용할 수 있는 format으로 바꾸느냐 하는 문제가 morphological analysis 부분에서 연구될 필요가 있습니다.

syntax의 분석에 있어서도, 크게 두가지의 관점이 있습니다. 먼저 언어학을 위한 tool로서 computer를 이용하는 관점이 있고, 또 전산학을 하는 사람들이 자연언어처리를 하기 위하여 언어학이론을 도입하는 관점이 있으며, 관점에 따라 tool도 완전히 달라집니다. 언어학하는 사람들이 이용할 수 있도록 개발된 tool로는 PATR Ⅱ, Marcus Parser 등이 있는데, 이들은 언어학자들이 자기의 이론을 검증해 보기 위한 것입니다. 예를 들어, Marcus Parser는 Chomsky의 이론중 하나인 universal grammar에 관한 인간의 competence를 check하기 위해 만든 system이며, 이 때문에 한계를 가지고 있고 저희들이 이용하기에는 문제가 있습니다. 자연언어처리를 하는 사람들은 보다 heuristic하고 자기들의 처리에 맞는 program을 개발하였는데, 그런 것으로는 ATN, chart parsing방법 등이 있습니다. 전산학자가 이용하는 tool은 어떤 이론을 넣기 보다는 case by case나, heuristics를 표현하기에 편리하게 되어 있습니다.

semantics에 관해 이야기하기로 하지요. 저도 이익환선생님의 「현대의미론」부터 출발하였는데, 거기에 나오는 Montague semantics를 computer로 처리하기는 불가능합니다. modal logic이나 possible world semantics 때문인데, 이런 것들을 없애려다 보니까, Situation Semantics도 나왔고, first-ordered predicate logic으로 처리해보려고 시도하고 있는 것입니다. first-ordered predicate logic이 computer로 처리할 수 있는 한계라 할 수 있으며, 이것을 넘어가면, 처리 algorithm이 존재하지 않습니다. 결국 first-ordered predicate logic에 머물러니까 거기에 따른 의미론이 개발되고, 그래서 언어학적인 명확성보다는 다른 방향에서의 접근이 이루어지고 있다고 할 수 있겠습니다. 한편 어떤 사람들은 상당히 heuristic한 semantics를 개발하기도 합니다. 극단적인 예로는 procedural semantics를 들 수 있는데, program 자체가 semantics가 되는 것이죠.

pragmatics는 요즈음 많이 연구되고

있기는 하지만 실제 처리되는 내용이 매우 단순합니다. indirect speech act를 처리하기 위해서는 상대방의 목적의식, 즉 상대방이 무엇을 하려고 하고, 나한테 무엇을 요구하는가의 분석을 통하여 attitude로 표현하고 이것을 분석할 필요가 있습니다. 이 과정에서, 이론자체보다는 상당히 많은 가정을 하게 되는데, 예를 들면, mutual belief가 있다든가…하는 것이죠. 그러다보니까 한계를 갖게 마련입니다.

전체적으로 결론을 내리면, 자연언어 처리가 생각보다는 쉽게 안된다는 것입니다. 작은 domain에서는 잘됩니다. 국어의 경우도 제한된 범위의 문장을 처리해 보면 매우 잘 되지만, 그것을 넘어서면 critical한 problem이 하나만 나와도 computational complexity가 엄청나게 증가하여 더이상 처리할 수 없는 dilemma에 빠지게 됩니다. 그러니까, 이러한 문제를 어떻게 조화시킬 것인가 하는 것이 현재부터의 연구방향이 아닌가 생각됩니다.

사회 : 전산학하고 언어학은 끊임없이 공동작업을 해야할 입장이라 할 수 있겠는데 언어학쪽에서 computational linguistics를 통하여 'Natural Language Parsing'이라는 책을 낸 적이 있었습니다. computer science하는 사람이 이 책을 review 한 것을 보니까, 제목은 parsing인데, 전부 generation에 관해서만 이야기하고 있다고 하였더군요. 사실 parsing이라하면, 주어진 string이 주어진 언어의 expression이나 문장이 되는지 recognize하는 방법, 즉 분석하는 방법인데 자꾸 generative rule을 쓰던 습관때문에 방향이 잘못되는 것이라 할 수 있습니다. 김한곤 선생님께서 기계번역과 관련하여 parser와 generator 사이의 관계를 언급하셨는데, 이에 대해 최기선 선생님께서 잠깐 이야기해 주시면 고맙겠습니다.

최기선 : 기계번역에서 parser와 generator의 과정을 간단히 말씀드리겠습니다. 꼭 parser라 하기보다는 analyser라 하는 것이 좋겠는데요, 기계번역도 engineering problem이니까 되든 안되든 morphological analyser, syntactic analyser, semantic analyser로 나누고 거꾸로 생성하는 것이죠.

morphological analyser의 첫번째 단계는 단어의 분할입니다. 영어는 쉽게 되지만, 일본어나 한국어는 좀 시간이 들겠지요. 하여튼 모든 가능성있는 분할을 넘겨줍니다. 그리고 syntactic analyser에 들어가기 전에 하나의 simple sentence에서 가장 중요한 것은 용언이니까, 그 용언에 대한, 아까 말한 Hornby pattern 같은 필수격 frame을 template에 올려놓습니다. 그러니까 지금까지는 morphological analyser이니까 단어가 분할되어 있고, 용언이 가질 수 있는 pattern 즉, 빈칸만 올라오는 것입니다.

syntactic analyser에서 첫번째 하는 일은—물론 synactic analyser를 들여다보면 대부분의 program에서 and나 or와 같은 coordination을 해결하는 것이 가장 커다란 부분이지만, 이를 제외한다면—template에다 거기에 맞는 필수격 조사나 필수격의 NP가 있느냐 없느냐를 찾아 메꾸는 것입니다. 메꿀 때에는 semantic feature나 여러가지 금지사항들을 사용하여 집어 넣습니다. 단, 조동사에 의해 case shifting이 일어나는 경우에, 그 전에 원상태로 돌려놓아야 하겠지요. 그러니까 사역이나 피동이 오면, Hornby

pattern에는 원동사에 대한 pattern들밖에 없기 때문에, 조동사에 관한 case shift pattern이 있어서 이것에 의해 거꾸로 원동사로 되돌려 필수격을 찾는 것입니다. 필수격 table을 보면, 표층격이 밝혀지면, 이미 심층격이 대응된 상태가 다 pattern table에 들어가 있으니까, 예를들어 tree를 만든다면, tree에 써주는 것이지요.

morphological analyser는 모든 가능성있는 것을 찾기 때문에 대부분 chart parser를 씁니다. 단어분할이 끝났다는 이야기는 결국 사전에 있는 모든 내용을 전부, chart parser에 loading 해놓은 것이라 할 수 있지요. syntactic analyser에 들어가서는 여러가지 방법이 있겠지만, 제가 경험한 system에서는 parse tree를 만들었습니다. 그리고 pattern matching에 의해 가능한 심층격을 모두 매달아서 syntactic analyser를 다시 한번 거칩니다. 두번째 phase에서는 optional case에 대해 처리합니다. optional case는 적당한 rule이 없기 때문에 대개 heuristics로 앞뒤로 보아서 하게 되는데, 상당히 긴 flow chart로 만들어져 있고 flow chart를 compile해서 빠르게 만들어, free case라든지, 필수격이 안 정해진 것, ambiguity가 있는 것들을 처리합니다. 마지막에는 번역을 위한 중간구조로 변환을 시켜줍니다. 그러니까 만일 pivot approach나 conceptual primitive를 쓴다면, 예를들어 한국어와 영어의 경우, 한국어와 영어의 중간에 normal form을 정의하여 그쪽으로 옮겨다 주지요. 물론 기계번역에서 이 과정이 가장 복잡한 부분이라 할 수 있겠는데, knowledge-based하고 그러면 복잡해지겠지만 현재의 상업적인 것들은 거의 그런 것은 하지 않고 빈칸으로 남겨둡니다.

중간구조인 conceptual primitive가 담겨져 있는 internal representation으로 가면, 생성을 시작합니다. 생성에서 맨처음에 하는 것은 style을 선택하는 것인데, sentence style에서 가장 primitive한 것으로는, 그러니까 그것이 즉, parse tree가 sentence인 경우가 있겠고, 그냥 NP로 된 것이 있겠는데, 이것을 결정하고, 또 앞으로 수동으로 번역될 것이냐, 사역으로 번역될 것이냐를 결정해야겠지요. 이것은 각 나라 말마다 특유한 것이니까, 적당한 것을 못 찾으면 source language의 문체를 그대로 옮기면 됩니다. 그 다음에 모든 단어들이 conceptual primitive로 쪼개져 있으니까. conceptual primitive를 보고 거기에 연결된 단어들을 가져옵니다—사전에는 surface word로부터 시작하여 conceptual primitive로 가는 쪽으로도 link가 되어 있고, conceptual primitive에서 여러개의 surface word로도 연결이 되어 있습니다. 그러니까, 생성할 때는 conceputual primitive를 보고서 거기에 연결된 모든 낱말을 가져오는 것이지요. 그래서 만약 가져온 낱말이 10개이라면, 어느 것이 더 적절한지를 점수로 매깁니다. 점수를 매기는 어떤 이론이 있는 것은 아니고, heuristic하게 경험적으로 해보는 것입니다. 여러가지 방법에 의해 점수의 합계를 내어 가장 높은 것을 찾게 되는데, conceptual primitve를 써서 기계번역에 실패할 경우, 이 부분이 가장 큰 원인이 되기도 합니다. 이 때문에 여러 회사에서 rule writer들이 자기가 필요로 하는 feature를 넣어서 사전을 다시 만드는 이중작업을 하고 있습니다.

sentence style을 처리한 다음 syntactic information을 올립니다. syntactic information은 analysis보다는 훨씬 간단합니다. analysis에서 optional case recognition은 매우 어렵고 복잡한 heuristics를 사용하였는데, 여기서는 대표적인 것에만 연결시켜주면 되기 때문이지요. 예를들어, '자격'이라 하면 '-으로'도 필요없고 '-으로서'의 한 형태만 연결시켜 주는 것입니다. 이런 것이 다 결정되면 어순을 결정합니다. 영어와 우리말의 경우는 어순이 거의 반대이지요. 맨마지막에는 phonological problem을 해결합니다. 그러니까, 우리말의 경우 활용이라든가 하는 문제를 해결함으로써 끝나게 되는 것이지요.

사회 : 감사합니다. 시간이 많이 부족하므로 '예, 아니오'로 대답할 수 있는 짧은 질문을 한두 개만 더 받도록 하겠습니다. floor에서도 좋고, discussants 중에서도 좋으니 긴요한 것이 있으면 질문해 주십시요.

이상억 : 저는 '예, 아니오'라고 대답할 필요도 없는 짧은 comment를 하겠는데, 아까 토론자 여러분이 여기저기에서 말씀하신 것들에 해당합니다. 특히 정희성 선생님께서 bottom-up과 top-down 두가지를 다 병행해야 한다고 하셨는데, 결과적으로 여러가지를 갖추면 보다 나은 system이 될 것은 분명하다고 생각합니다. 그래서 저도 거기에 동의하고, 오늘 최기선 선생님께서 드신 예로 이것을 예증해 볼까 합니다.

'감기는'은 '이번 감기는 독하다'에서의 명사와, '칭칭 감기는 낙지발'에서의 passive form과 '실을 감기는 편하다'에서의 명사형등 여러가지로 나타납니다. 대부분 phonology가 밑에 있다고 하고 top-down으로 내려오시는데, 그럴때 명사형 '감끼'의 '끼'를 어떻게 잡아내실지 궁금하군요. 물론 input에서 nominalizer를 '끼'로 잡아주면 되겠지만, '실을 풀기는 쉽다'에서 '풀기'의 '기'는 그렇지가 않습니다. 같은 '-기'에 대해 어떻게 조정할 수 있는 방법이 있겠지만, 이런 경우 bottom-up으로 들어가면 phonological 또는 phonetic information에서 즉각적으로 좋은 hint를 얻을 수 있지 않을까 해서 정희성 선생님의 approach가 당연한 것이라 생각합니다.

그리고 김한곤 선생님께서 '표기상 또는 사용상의 난맥상'이라고 표현하신 문제는 음성인식에 있어서도 마찬가지로 봉착하는 것인데, 매우 variation이 많기 때문에, 처음에는 speaker-dependent한 방향으로 시작해서 적절한 filter장치들이 되면 speaker-independent하게 data를 처리할 수 있도록 나아가는 그러한 과정을 경험하신 것이 아닌가 생각합니다.

사회 : 대답이 필요없다고 하셨기 때문에, 대답은 없겠습니다. 그러면 그것과 관련된 문제인데, Karttunen이 tool level phonological rules의 system을 개발하였습니다. 사실 정희성 선생님의 그것도 관련이 있어 보이고 morpheme의 base form인 underlying phonological representation과 phonetic manifestation의 관계를 잇는 좋은 system으로 각광을 받고 있다고 할 수 있습니다. 아까 tree-diagram과의 관계가 어떻겠는지 간단하게 말씀해주십시요.

정희성 : 음성인식 system에 있어서 phonology에 대한 어느 이론을 짜서 하는 것으로는 Hearsay Ⅱ를 알고 있습니

다. 그것은 knowledge representation의 형식으로 보면 blackboard 형식이라고 하는데, syntax까지 reference를 해서 하나의 단어를 찾는다는 것입니다. 음성인식쪽에서는 rule, 즉 knowledge를 어떤 형식으로 표현할 것이냐 하는 것이 중요하지요. unification-based formalism으로 phonology를 기술한 것은 저희들이 처음이 아닌가 생각합니다.

사회 : 감사합니다. 긴급한 질문이 있으시면 하나만 더 받겠습니다.

박병수 : 제가 지금 제기된 문제에 대해서 생각을 달리하기 때문에 한가지 말씀드리겠습니다. 우리말의 형태론의 복잡성이 대단하다는 것은 잘 알고 있는 사실인데, 이것 때문에 앞에 주제발표하신 두 분이 지적하셨고, 단어의 기저형이 어떻게 정해져야 하느냐 하는 문제도 나왔습니다. 조금전에 정희성 선생님께서는 morphology의 복잡성 때문에 HPSG의 이론을 갖다 쓸 수조차 없어서 우리말에 맞는 새로운 이론이 세워져야 한다는 상당히 극단적인 비판까지 하셨는데, 제 생각으로는 그럴 필요까지는 없다고 봅니다. morphology-syntax interface 이론도 있고, Karttunen의 이론, 또 phonology와 syntax가 서로 interaction하는 이론 등 언어학에서 많은 이론이 나오고 있습니다. 간단히 말해서 lexicon속에 형태규칙을 잉여규칙으로 넣어서, 마치 공장처럼 많은 단어를 만들어 내도록 하는 것이지요. 이러한 공장을 통해서, '감기는'도 나오고, '차는'도 나오고, '내가'도 나오고… 이것들을 단어로 생각하면 될 것 같습니다. 결국 syntax를 복잡하게 하지 않아도 되고, 그 대신 lexicon쪽이 조금 복잡해지겠지요. 그렇게 하면 전혀 새로운 이론을 생각해낼 필요는 없다고 봅니다.

정희성 : 잠깐 오해가 있으셨던 것 같습니다. HPSG를 쓰지 말자는 것이 아니라, 그러한 paradigm을 받아들이되, 그것을 확충한다든가, 뺄 것을 빼고 우리말에 맞도록 수정을 하자는 말씀을 드렸었습니다.

사회 : 네, 그럼 조금 아쉬운 점이 있기는 하지만 이것으로 '자연언어처리의 이론과 방법'이라는 커다란 주제의 토론회를 마치도록 하겠습니다. 감사합니다.

〈정리 : 최동주〉