한국어 문서의 유형 정보를 이용한 EMFA의 구현

장 정 효*, 이 상 열°, 이 상 곤*, 조 현 준°
전주대학교 정보기술공학부°, 전주대학교 일반대학원 컴퓨터공학과 언어과학실*
{fsfsharp, pcguys7, samuel, hycho}@jj.ac.kr

Implementation of E-Mail Filtering Agent by Using Document Type Information

Jeong-Hyo Jang, *Sang Yeol Lee°, Samuel Sangkon Lee, Hyun-joon Cho
Language Science Lab., Dept. of Comp. Sci. & Eng., Jeonju University

요 약

전자메일은 일상의 연락수단 일 뿐만 아니라 여러 목적의 업무처리에 있어서도 매우 중요한 통신수단이지만 사용자는 전자메일을 처리하는데 상당히 많은 시간이 걸리고 있다. 본 논문은 메일 서버에 도착한 메일의 중요도를 자동적으로 판정하여 빠른 업무 처리에 도움을 주는 메일 클라이언트를 개발하였다. 본 프로그램은 수신된 메일 문서에서 송신처, 제목, 문서 유형, 시간제한 어구의 출현 유무 등의 여러 가지 속성값을 추출하여 이를 조합하여 저장한 후, 새로운 전자메일이 도착하였을 때 이미 파악된 사용자의 유형을 파악한 구조화된 지식을 이용하여 전자메일을 자동으로 필터링하는 새로운 개념의 메일 클라이언트를 구현하였다.

1. 서 론

인터넷이 우리 생활에 보급됨에 따라 전자메일이 일상의 연락수단으로 사용될 뿐만 아니라, 여러 형태의 업무처리에도 중요한 통신수단으로 이용되고 있다. 전자메일은 편리한 작성과 신속한 전달 속도, 통신비용의 절감 등 여러 가지 이점을 가지고 있기 때문에 현대사회에서 대부분의 서류는 전자메일을 통해 전달되고 있다. 이와 더불어 개인의 메일 서버에는 대량의 메일문서가 수신되며 이 중에는 중요한 업무메일과 함께 광고메일, 스팸메일 등도 점차 증가되어 개인의 업무처리에 큰 지장이 되고 있는 실정이다. 이에 따라 중요도가 높은 메일을 먼저 처리할 수 있는 내용기반(contents-base) 정보 필터링(information filtering) 기술의 필요성이 대단히 높아지고 있다. 따라서 본 논문의 목적은 위에서 언급한 연구 배경을 목적으로 전자메일 자동 필터링 프로그램(EMFA; E-Mail Filtering System)을 개발하고자 한다.

본 논문은 메일의 송신처, 전송 시간, 본문의 원본 텍스트에서 추출한 키워드 등을 이용하여 다양한 형태의 메일 필터링 시스템을 구현할 때 필요한 내용을 제공한다. 본 논문의 개발 범위는 전자 메일의 내용과 첨부 파일이 텍스트 파일인 경우만을 대상으로 처리하고, 텍스트 형태의 파일 이외의 파일이 첨부된 메일 문서는 필터링 대상에서 제외하였다.

2. 메일의 데이터 형식과 구조

MIME(Multipurpose Internet Mail Extensions)은 텍스트 메시지에서 기본적인 텍스트 지향 인터넷 메일 시스템을 확장하여 메시지에 임의의 바이너리 파일을 포함할 수 있게 해준다. 이것은 사용자가 비-텍스트 문서(워드프로세스 파일이나 스프레드시트 등)를 다른 사용자에게 보내고자 할 때에도 유용하게 사용된다. MIME의 구조에 대하여 설명하면 다음과 같다.

▼ MIME 메시지 헤더

- MIME Version : 메시지에 사용된 MIME의 버전 정보를 제공한다.
- Content-Type : 메시지를 파싱할 수 있도록 데이터의 유형을 식별한다. 데이터는 정의된 매체 유형(Media Type) 중 하나로 식별된다.
  - text/plain : 평문 ASCII 텍스트 메시지
  - text/html : 하이퍼텍스트 마크업 언어
- Content-Transfer-Encoding : 메시지에서 가장 중요한 정보이다. 이 헤더는 메시지에서 수행되는 인코딩 유형을 보여주고, 그에 따른 디코딩 방법에 관한 정보를 메일 클라이언트에게 제공한다.
  - 7 bit : 단순한 US-ASCII 텍스트로 0에서 127까지의 아스키 십진수 값을 가진 옥텟만 허용한다.
  - 8 bit : 아스키 문자의 127 이상의 십진수 값이 허용되는 점을 제외하고는 위의 7 bit의 경우와 동일하다.
  - binary : 문자 집합으로 구성될 수 있다.
  - Quated-printable : 보통의 텍스트이며 사람이 읽을 수 있지만, 7 bit가 아닌 데이터를 위해 사용한다.
  - Base 64 : 7 bit와 quoted-printable 인코딩 유형에 적합하지 않은 파일을 네트워크를 통해 송신할 때 사용한다.

▪ Content-ID : 메시지에 국제적인 식별 번호(고유 번호)를 제공한다.

▪ Content-Description : 메시지 부분의 콘텐츠에 대한 텍스트 설명이다. 예를 들어, 첨부 파일에 대한 설명을 들 수 있다.

▪ Boundary : 메시지에 있는 메시지 부분들 사이의 경계를 정의하는 7 bit와 US-ASCII 텍스트로 된 스트링(대/소문자 구분)이다. 이 정보는 메시지의 본문과 첨부 파일을 구별할 때 사용한다.

3. 인코딩의 유형

3.1 Quoted-printable

이 방식의 인코딩은 보통의 텍스트이므로 사람이 읽을 수는 있지만, "7 bit"가 아닌 데이터의 처리를 목적으로 사용한다. 이것은 8 비트 혹은 순수한 텍스트인 데이터를 SMTP를 통해 전송하기 위해 7 비트 형식으로 변환하는 방법이다. quoted-printable 인코딩 방식으로 작성된 일부 바이너리 정보를 인코딩하는 것이 가능하지만 대부분의 경우 바이너리 데이터는 base 64 방식으로 인코딩한다.

3.2 Base 64

Base 64 인코딩은 7 비트와 quoted-printable 인코딩 유형이 적합하지 않은 파일을 전송할 때 사용된다. 모든 데이터 형식은 Base 64로 인코딩 될 수 있지만, 7 비트 텍스트는 그대로 두고, 다른 텍스트 데이터는 quoted-printable로 인코딩 하는 것이 전송 시에 효율적이다.

<표 1> Value Encoding의 예

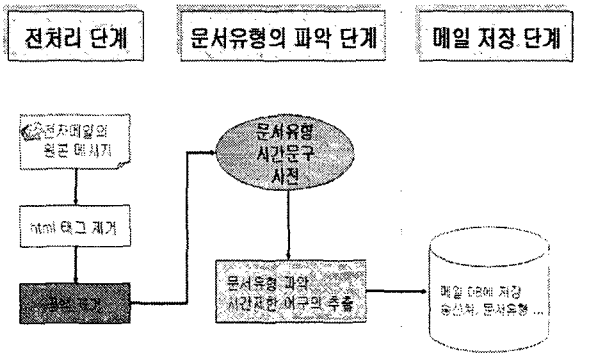
0	A	11	L	22	W	33	h	44	s	55	3
1	B	12	M	23	X	34	i	45	t	56	4
2	C	13	N	24	Y	35	j	46	u	57	5
3	D	14	O	25	Z	36	k	47	v	58	6
4	E	15	P	26	a	37	l	48	w	59	7
5	F	16	Q	27	b	38	m	49	x	60	8
6	G	17	R	28	c	39	n	50	y	61	9
7	H	18	S	29	d	40	o	51	z	62	+
8	I	19	T	30	e	41	p	52	0	63	/
9	J	20	U	31	f	42	q	53	1	pad	=
10	K	21	V	32	g	43	r	54	2		

Base 64 인코딩은 직접 프린팅이 가능한 제한된 텍스트로 아스키 제어 문자를 포함하여 모든 비트열을 전송하는데 사용된다. Base 64로 인코딩된 문자는 pad 문자('=')를 제외하고는 64개(0-63)이다. 이것은 아스키 셋트(ASCII Set)의 절반 밖에 안 되는 사이즈로 제어 문자, +, / 등의 특수 문자가 제외된 것이다. 다시 말하면, A~Z, a~z, 0~9, +, / 만 존재하는 문자 세트(Character Set)이다. 인코딩의 원리는 24 비트(3 바이트)를 6 비트씩 네 등분으로 나누어 각각의 6 비트를 위의 <표 1>에서 보는 바와 같이 각각의 대응 문자(8 비트)로 변환한다. 만약 뒷부분의 비트가 없을 경우는 0(2진수)으로 채우고 인코딩한다. 인코딩을 수행할 대상이 전혀 존재하지 않는 경우에는 pad 문자로 변환한다. 따라서 인코딩 결과 6 비트가 8 비트로 늘어나므로 메일 본문의 처음 크기에서 약 1.33배 정도로 항상 증가한다.

4. 필터링 방법

4.1 EMFA의 시스템 구조

문서 유형과 시간 어구를 효율적으로 추출하기 위해 먼저 전처리 과정을 거친다. 전처리 단계에서는 두 가지의 간단한 처리 과정을 거치는데, 하나는 HTML 태그와 공백(space)을 제거한다. 문서 유형을 파악할 때 특정 단어를 스트링 매칭으로 검사하게 되는데, 태그를 제거함으로써 더 빠른 처리가 가능하게 된다. 공백을 제거하는 이유는 사용자 개인별로 띄어쓰기를 다르게 하기 때문이다. 예를 들어, "제 출"로 입력하거나, "제출"로 입력하는 등 한국어의 자유로운 띄어쓰기로 인해 모든 공백을 제거한다. 문서 유형 사전과 시간 표현 어구를 정리한 사전을 이용하여 메일의 제목과 본문에서 문서 유형과 시간 제한 어구를 추출한다. 전처리 과정을 거치고 문서 유형과 시간 제한 어구를 추출 과정이 끝났으면 최종적으로 데이터베이스에 메일의 내용을 저장한다.



(그림 1) EMFA의 전체 시스템 구조도

4.2 HTML 태그 제거

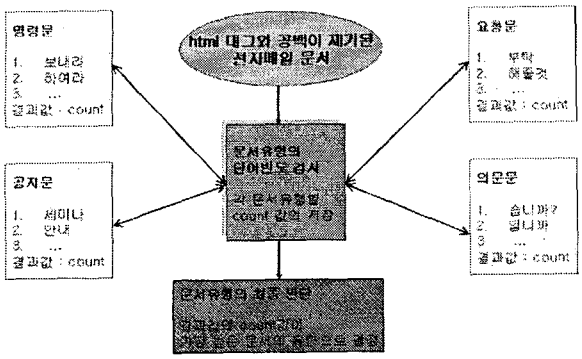
Html 태그의 제거는 다음과 같이 C#에서 지원하는 .NET FRAMEWORK의 정규식 클래스를 이용하여 제거한다.

```
System.Text.RegularExpressions.Regex.Replace(htmlString,
"<(.*?)>", "")
```

위에서 첫 번째 인자는 html 문서가 포함된 문자열이고, 두 번째 인자는 정규식 표현을 작성하는데, "<(.*?)>"는 html 태그를 의미한다. 세 번째 인자는 변환할 값을 삽입한다. 전체의 태그를 제거하고자 하기 때문에 "" 빈 문자열을 사용하였다. 그러면 모든 html 태그는 공백("") 으로 채워지게 되기 때문에 최종적으로 html 태그 제거 기능의 역할을 하게 된다. 위의 메서드를 실행하면 태그는 모두 제거되고, 문자열만을 리턴 한다. 공백 제거는 C#에서 제공하는 Replace() 문자열 메서드를 이용하여 제거하였다.

4.3 문서 유형 파악

문서 유형 파악은 문서 유형별로 구축된 사전을 이용한다. 메일의 제목이나 본문에 문서의 유형에 해당하는 단어를 차례로 매칭한다. 해당하는 문서 유형의 단어가 제



(그림 2) 문서 유형의 결정 방법

목에 출현 시 본문보다 좀 더 높은 가중치를 부여하였다. 왜냐하면 보통의 경우 제목에 본문을 요약하면 문구가 많이 포함되기 때문이다. 단어가 출현하면 해당하는 문서의 유형을 지시하는 카운트 값을 하나 증가시킨다. 위의 (그림 2)에서 보는 바와 같이 결과 값의 최종 산출 시, 이 카운트 값이 가장 높은 문서유형을 결정한다.

4.4 시간제한 어구의 처리

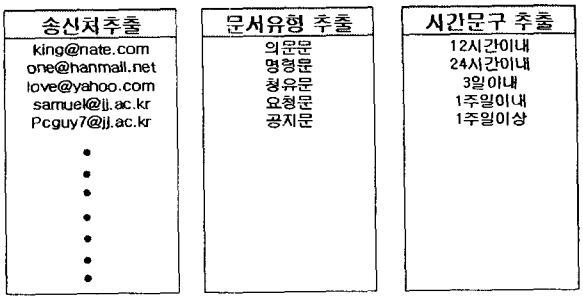
한국어에서 시간의 제한을 나타내는 시간제한 표현 어구의 처리는 문서 유형의 파악과 동일한 알고리즘을 사용하였다. 여러 개의 어구가 출현하면 마지막에 제시된 어구를 중심으로 결정하였다. 예를 들어, "가능한 한 빨리 보내주세요. 혹은 "될 수 있으면 1시간 이내에 보내주세요"란 어구가 출현하는 메일은 가장 최후에 발견된 "1시간 이내" 어구로 결정한다.

4.5 필터링 모드

필터링 모드는 송신처 추출과 문서 유형, 시간 어구 추출 등 크게 세 가지 모드로 구별하였다. 특정 송신처 우선순위를 높게 설정할 것인지, 혹은 특정한 문서 유형의 우선순위를 높일 것인지, 시간문구의 우선순위를 높게 정의할 것인지를 결정하게 된다.

사용자마다 자신이 중요하게 생각하는 속성에 우선순위를 다르게 설정할 수 있으며, 조직 구성원의 변화에 따라 우선순위의 변경도 가능하도록 하였다. 이러한 필터링 기능은 POP3 서버에서 이미 메일을 받아온 상태에서도 무리 없이 동작하도록 구현하였다.

지금까지 POP3 프로토콜과 이메일 필터링 클라이언트 시스템 구조에 대해서 살펴보았다. 다음에는 본 논문에서 제시하는 실제 구현 프로그램을 설명한다. 프로그램이 처음 실행되면 사용자 DB 생성 및 POP3 서버 설정을 한다. 각 사용자 마다 DB를 생성하게 된다. POP3 서버의 올바른 설정을 테스트하기 위해 POP3 서버 설정 테스트 기능을 제공하며, 사용자 설정을 마친 후 POP3 서버에서 메일을 받아온다. 메일 받기가 완료되면 즉시 문서 유형 파악 및 시간 어구를 추출하고 유형별 메일을 순위화하여 (그림 3)과 같이 사용자에게 제시한다. 주소록에서 그룹 생성을 하면 그룹 메일 폴더함이 생기고 그룹 멤버의 메일 주소를 확인하여 해당하는 그룹 편지함으로 메일이 전송되게 된다. 메일 계정 설정 기능은 사용자가 여러 개의 POP3 서버를 사용할 때 관리하는 기



(그림 3) 사용자 우선순위의 설정 방법

능으로 POP3 서버의 추가, 삭제 수정이 가능하다. 사용자가 필터링 우선순위를 설정할 수 있는 필터링 설정 기능은 메일의 각 송신처별로 문서유형의 우선순위 설정과 원하는 문서 유형을 추가 그리고 추가된 문서의 우선순위를 자동으로 설정할 수 있다.

5. 결 론

본 논문에서는 수신된 메일 문서에서 다중속성 항목으로 구성된 프로파일과 입력되는 메일 문서에서의 중요도를 계산하여 지능적이며 사용자 편의성 위주로 설계되었다. 본 논문의 EMFA 시스템은 사용자가 중요성을 느끼는 송신처와 문서 유형을 포함한 문서를 우선적으로 처리할 수 있어서 신속하고 효율적인 업무처리를 기대할 수 있다. 향후에는 시간제한 표현의 포함여부를 자동으로 판정하여 긴급하게 처리할 업무는 사용자에게 충고하는 지적인 능력을 갖는 메일클라이언트의 구축과 이를 모바일 기계나 유비쿼터스 컴퓨팅 환경에 적용시킬 수 있는 시스템으로 확장할 계획이다.

감사의 글 : 본 과제(결과물)는 교육인적자원부, 산업자원부, 노동부의 출연금으로 수행한 산학협력중심대학 육성사업의 연구결과입니다.

참고 문헌

[1] Pawel Lesnikowski, http://lesnikowski.fm.interia.pl

[2] David Wood, 채규혁 역, Programming Internet E-mail, 한빛미디어, 2000.

[3] Charless Petzold, 김태현, 박한돌 역, Programming Microsoft Windows with C#, 정보문화사, 2002.

[4] Richard Blum, 김형규, 최낙준 역, 사이텍 미디어, 2004, C# Network Programming, 2004.

[5] 김 보 미, "프로파일 정보에 기반한 전자메일 클라이언트 시스템의 설계 및 구현", 전주대학교 교육대학원 석사학위 논문, pp. 01-48, 2005.

[6] 김 보 미, 이 상 열, 이 상 곤, "이메일문서의 속성값에 기반한 필터링 시스템의 설계 및 구현", 컴퓨터종합학술대회 2005 논문집, 제 32권, 제 1(B)호, pp. 142-144, 2005.

[7] 김 보 미, 이 원 휘, 이 상 곤, "전자메일의 중요도에 기반한 이메일문서 필터링 방법", 정보처리학회 제 22회 추계 학술발표 논문집, 제 11권, 제 2호, pp. 811-814, 2004.