creative commons

COMMONS DEED

저작자표시-비영리-동일조건변경허락 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.
- 이차적 저작물을 작성할 수 있습니다.

다음과 같은 조건을 따라야 합니다:

저작자표시. 귀하는 원저작자를 표시하여야 합니다.

비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.

동일조건변경허락. 귀하가 이 저작물을 개작, 변형 또는 가공했을 경우에는, 이 저작물과 동일한 이용허락조건하에서만 배포할 수 있습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 이용허락규약(Legal Code)을 이해하기 쉽게 요약한 것입니다.

Disclaimer

dCollection

공학석사 학위논문

DietAdvisor: 스마트폰 상에서의 개인 건강 관리 어플리케이션

DietAdvisor: A Personalized eHealth Agent in a Mobile Computing Enviroment

2012년 8월

서울대학교 대학원

컴퓨터공학부

임병권

DietAdvisor: 스마트폰 상에서의 개인 건강 관리 어플리케이션

DietAdvisor: A Personalized eHealth Agent in a Mobile Computing Enviroment

지도 교수 장 병 탁

이 논문을 공학석사 학위논문으로 제출함
2012년 8월

서울대학교 대학원
컴퓨터공학부
임병권

임병권의 공학석사 학위논문을 인준함
2012년 8월

위 원 장 _____ 최 양 희 _____ (인)

부위원장 _____ 장 병 탁 _____ (인)

위 원 _____ 김 종 권 _____ (인)

초 록

전세계에서 건강에 관한 사람들의 관심이 높아지고 있다. 건강에 관한 관심이 높아짐에 따라 사람들은 체중관리에 많은 노력을 쏟고 있으며, 체중관리는 정부의 국민 건강을 위한 정책으로도 나타나고 있다. 체중관리를 위해서는 개인의 식단과 운동을 관리해야 한다. 하지만 전문지식이 없는 일반인이 자신 스스로 먹은 식단을 분석하고 어떤 음식을 먹으면 좋을지 추천하는 것은 쉽지 않다. 따라서 본 연구실에서는 하이퍼네트워크를 이용해 개인 맞춤형 식단 추천 알고리즘을 개발하였다. 하이퍼네트워크는 복잡한 요소들의 연관관계를 잘 표현하고 학습할 수 있는 방법론으로 이를 이용한 식단 추천 알고리즘은 사용자의 식단 로그를 이용하여 사용자의 취향을 학습하고 이를 통해 사용자가 체중관리를 위해 적절한 칼로리를 지닌 식단을 추천한다. 개발된 식단 추천 알고리즘을 이용하여 본 연구실에서는 체중관리를 도와주는 DietAdvisor라는 스마트폰 어플리케이션을 개발하였다. DietAdvisor는 사용이 쉽고 개인화 된 서비스를 제공하게 설계되었으며, 체중관리에 필요한 식단과 운동 방면의 서비스를 제공할 수 있어 실제 사용자들에게 체중 조절에 관한 도움을 줄 수 있다.

주요어 : 하이퍼네트워크, 추천 시스템, 행동 인식, 지능형 에이전트, 패턴 인식

학 번 : 2010-23287

목 차

1. 서론 1

1.1 연구의 배경 1

1.2 연구의 내용 1

1.3 논문의 구성 2

2. DietAdvisor 3

2.1 데이터 입력 부분 4

2.2 운동 추천 모듈 4

2.3 식단 추천 모듈 5

3. 하이퍼네트워크를 이용한 식단추천 알고리즘 7

3.1 식단 추천 기술 관련 연구 7

3.2 하이퍼네트워크 9

3.3 식단 데이터 11

3.4 식단추천 알고리즘 12

4. 식단 추천 알고리즘 성능평가 14

5. 결론 및 향후 연구 24

표 목차

표 1 사용자 활동별 MET 수치. 5

표 2 그림 6과 8의 주요 에지 음식명. 20

그림 목차

그림 1 DietAdvisor의 전체구조..... 3

그림 2 하이퍼네트워크 구조..... 9

그림 3 음식 정보 데이터베이스의 구조와 음식 데이터의 형태. 11

그림 4 날짜별 입력 식단과 추천 식단의 분포..... 15

그림 5 실날짜별 입력 식단과 추천 식단의 분포 차이..... 16

그림 6 날짜별 하이퍼네트워크 모델의 주요 에지들의 변화. 17

그림 7 날짜별 실험자 6 & 8와 실험자 8 & 9 간의 추천 식단 거리차이. 18

그림 8 실험자 6,8,9의 하이퍼네트워크 모델 변화. 19

그림 9 사용자 별 추천 식단 분포. 20

그림 10 실험자 별 타 실험자와의 추천 식단 거리 차. 21

그림 11 실험자 8(좌)와 실험자 9(우)의 추천 식단 예제. 22

1. 서론

1.1 연구의 배경

본문내용 시작 전 세계적으로 건강은 큰 관심사이며 많은 사람들이 신경을 쓰는 문제이다. 이러한 상황에서 체중 조절은 대부분의 사람이 가장 흔하게 접하는 중요한 건강 문제이다. OECD 통계에 따르면 2009년 OECD 가입국 과체중 인구 비율은 남성 61.0%, 여성 51.9%로 인구의 절반 이상이 과체중 상태이다[1]. 과체중이나 비만은 고혈압이나 당뇨등의 원인이 되므로 체중 조절은 건강 관리에 매우 중요한 부분이다[2]. 체중 관리를 위해서는 식단 관리가 반드시 필요하다. 하지만 전문지식이 없는 사람은 자신의 식단을 분석하고 새로운 식단을 구성하는데 어려움을 겪는다. 이러한 문제를 해결하기 위해 사용자의 체중관리에 도움을 줄 수 있는 스마트폰 어플리케이션을 개발하고자 하였다.

1.2 연구의 내용

이를 위해서 본 연구실에서는 하이퍼네트워크를 이용한 식단 추천 알고리즘을 개발하였다. 하이퍼네트워크는 다수의 노드를 연결한 하이퍼에지의 가중치를 주어진 데이터를 통해 학습 시키는데, 이를 통해 각 요소들의 관계가 잘 반영되도록 데이터 전체의 분포를 학습할 수 있다. 개발된 식단 추천 알고리즘은 하이퍼네트워크의 특성을 잘 반영하여 사용자의 식단 로그를 학습해 사용자의 식성을 파악할 수 있으며, 파악된 사용자의 식성을

이용해 체중조절 계획에 적절한 칼로리를 가진 식단을 추천할 수 있다. 따라서 전문지식이 없는 일반인이더라도 이 식단 추천 알고리즘을 이용하면 자신의 식성에 맞고 적절한 칼로리를 지닌 식단을 추천 받을 수 있다.

또한 개발된 식단 추천 알고리즘을 이용하여 DietAdvisor라는 체중관리 스마트폰 어플리케이션을 개발하였다[3]. 대부분의 체중관리 어플리케이션은 서비스의 폭이 제한되었지만 DietAdivsor는 식단과 운동 양쪽 방면의 서비스를 제공하여, 사용자의 체중관리에 필요한 모든 부분에서 도움을 줄 수 있다. 또한, DietAdvisor는 사용자마다 자신의 데이터를 이용하여 자신에게 맞는 서비스를 받을 수 있도록 개발되었으며, 사용자의 입력을 최소한으로 제한하여 사용자가 이용하는데 번거로움이 적게 만들었다.

1.3 논문의 구성

1장에서는 연구의 배경과 연구의 내용을 설명하였다. 이후로는 2장에서 DietAdvisor에 관하여 설명하고 3장에서 하이퍼네트워크 모델을 이용해 개발된 식단 추천 알고리즘에 관하여 설명할 것이다. 4장과 5장에서 개발된 알고리즘의 실험 결과를 다양한 방면에서 제시하고 결론을 논의할 것이다.

2. DietAdvisor

본 연구진은 개발된 식단 추천 알고리즘을 이용하여 사용자의 체중 조절에 전반적인 도움을 줄 수 있도록 DietAdvisor를 개발하였다. DietAdvisor는 사용의 정보를 바탕으로 사용자에게 식단 추천과 운동 추천 서비스를 제공한다. 따라서 DietAdvisor의 사용자는 체중관리에 필요한 전반적인 서비스를 제공받을 수 있다. 또한, 사용자의 이용 편의성을 위해 스마트폰 센서를 사용하여 사용자의 활동정보를 수집하고 이를 통해 자동으로 사용자의 소모 칼로리를 계산하여 사용자에게 입력을 요구하는 정보를 줄였다. 그리고 입력된 사용자의 정보는 사용자의 서비스를 위해 다시 이용되므로 사용자의 자신에게 맞는 개인화된 서비스를 제공받는다. DietAdvisor는 그림1과 같은 구조를 가지고 있으며, 이후의 세부 절에서 각 부분을 설명하겠다.

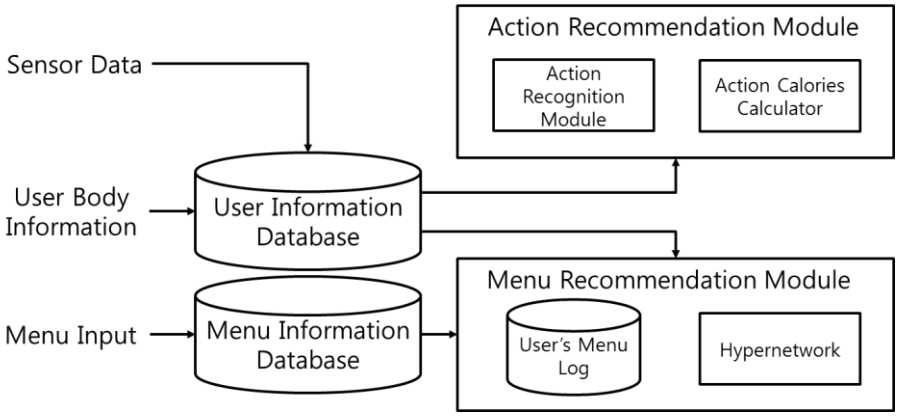

그림 1 DietAdvisor의 전체구조. 스마트폰의 Sensor Data와 사용자가 입력한 user body information 및 menu input을 바탕으로 사용자에게 운동 추천 서비스와 식단 추천 서비스를 제공한다.

2.1 데이터 입력 부분

DietAdvisor의 동작에 필요한 데이터는 사용자 입력과 스마트폰 센서를 이용하며 그림1의 sensors data와 user body information, menu input부분이다. 사용자는 어플리케이션이 처음 동작할 때 나이, 성별, 키, 몸무게의 정보를 입력해야 한다. 이렇게 입력된 정보를 바탕으로 사용자의 신체상태와 수준을 판단하여 추천 서비스에서 이용한다. 또한 사용자는 자신이 먹은 식단을 입력해야 한다. 식단이 입력되면 Menu Information Database에서 식단들의 정보를 찾아 저장한다. Menu Information Database는 8000여 가지의 메뉴 아이템이 기록되어 있으며, 각각의 칼로리와 카테고리 정보를 기록하고 있다. 이렇게 입력된 메뉴를 바탕으로 사용자가 섭취한 칼로리와 식단 추천 알고리즘을 위한 데이터를 수집한다.

Smartphone’s sensor는 가속도와 GPS센서를 이용하는데, 이를 이용하여 사용자의 활동 상태를 판별하는데 이용한다. 스마트폰 센서를 이용함으로써 사용자가 입력할 필요 없이 사용자가 소모한 칼로리를 계산할 수 있다.

2.2 운동 추천 모듈

운동 추천 모듈은 사용자가 체중 관리를 위해 적절한 달리기 시간을 알려주는 부분이다. 스마트폰 센서에서 얻어진 사용자의 센서 정보는 Action recognition module을 통해 사용자의 운동상태를 판별한다. 운동 상태가 판별되면 DietAdvisor는 아래의 식을 사용해 사용자의 운동상태를 통해 사용자가 소모한 칼로리를 계산할 수 있다.

$$\text{Calorie consumption(kcal/day)} = Basal\ metabolism + Active\ metabolism$$

$$\text{Basal metabolism(kcal/min)} = 66.47 + \left(13.75 \times Weight(kg)\right) + \left(5 \times Height(m)\right) - (6.76 \times Age(year))$$

$$\text{Active metabolism(kcal/min)} = 0.0175 \times \text{Duration(min)} \times \text{Weight(kg)} \times \text{MET}$$

식단 로그를 통한 칼로리 섭취량과 위 식을 이용한 칼로리 소모량을 통해 사용자가 체중을 조절하기 위해선 오늘 얼마나 운동을 해야 하는지 알려준다. 사용자가 운동을 함에 따라 남은 필요한 운동시간은 즉각적으로 변동되며, 사용자의 운동 상태에 따라 남은 시간도 변화한다. 운동 추천 모듈을 통해 사용자는 자신의 체중 관리 계획에 적절한 운동 추천 서비스를 제공받을 수 있다.

MET: Metabolism Equivalent of Task			
Task	MET		
Walk	3km/h	2.5	
		4km/h	3
Run		7km/h	7
		8km/h	8

표 1 사용자 활동별 MET 수치

2.3 식단 추천 모듈

식단 추천 모듈 사용자에게 체중관리에 필요한 식단 추천 서비스를 제공한다. 이를 위해 DietAdvisor는 3장에서 설명할 식단

추천 알고리즘을 이용한다. DietAdvisor에 입력된 사용자의 식단 로그가 기준 시간 이상 쌓이면 하이퍼네트워크 모델을 재학습시켜 사용자의 식성이 변하는 것을 그때그때 반영할 수 있게 설계하였다. 이렇게 학습된 식단 추천 알고리즘은 사용자의 식성에 맞춰 식단을 추천하게 되므로 사용자는 자신에게 맞는 개인화된 서비스를 제공받을 수 있다. 또한 DietAdvisor에 기록된 하루 동안의 칼로리 소모량과 섭취량을 통해 사용자가 한끼 식사로 필요한 칼로리를 계산할 수 있으며, 이를 통해 사용자는 적절한 칼로리를 지닌 식단을 추천 받을 수 있다.

3. 하이퍼네트워크를 이용한 식단추천 알고리즘

3.1 식단 추천 기술 관련 연구

사용자들의 취향을 학습하여 추천에 이용하는 기계학습 기술은 Netflix의 영화평점 추정대회나, Amazon의 쇼핑 추천 기술 등에 사용되고 있다. 주로 사용하는 기법은 Collaborative filtering 기법을 사용하며[4], 사용자의 편의성 향상에 많은 도움을 주고 있다.

식단 추천 기술은 60년대 초에 처음으로 개발되었다. Balintfy는 적절한 영양소를 가진 식단을 구성하기 위한 수학적 방법론을 개발하였다[5]. 하지만 개발된 방법은 임의성이 포함되지 않아 같은 조건에서 항상 같은 결과를 내는 문제점이 있었다. 이에 이어 Eckstein은 임의성을 추가하여 음식의 조리시간, 영양소, 비용 등을 따져 하나의 식단을 구성하는 방법을 개발하였다[6]. 하지만 포함되어 있는 음식이 너무 적고 사용자의 취향을 고려하지는 못하였다. 이후로는 expert system을 이용한 식단 추천 방법들이 개발되었다. 조리 시간, 비용, 영양소 등을 고려하여 적절한 식단을 구성하는 rule-based expert system[7]과 사용자의 기록을 통해 식단을 구성하는 case-based expert system이 개발되었다[8]. 또는 유전 알고리즘을 이용하는 방법도 개발되었는데[9], 역시 메뉴가 매우 제한된 상황을 가정하고 있다.

현대에는 인터넷의 발달로 대량의 데이터를 손쉽게 얻을 수 있기 때문에, collaborative filtering 기법이 사용되고 있다[10]. Collaborative filtering 기법은 추천을 하기 위해서 복잡한

규칙들이 필요하지 않기 때문에 현재 대부분의 추천 기술들에 사용되고 있다[11]. Collaborative filtering 을 사용한 식당 추천 방법을 개발한 논문이나[12], 슈퍼마켓 쇼핑 목록을 추천해주는 논문은 사용자 식단 추천에 응용될 수 있을 것이다. 하지만 Collaborative filtering 기법을 식단 추천 알고리즘에 바로 적용하기에는 무리가 있다. 실제로 식단 추천에 Collaborative filtering 기법을 사용한 논문들이 있지만[13][14], 하나의 주메뉴만을 추천할 뿐이다. 일반적인 collaborative filtering의 특성상 unconfiguable한 데이터만을 다룰 수 있기 때문이다[15]. 식단은 주메뉴 뿐만 아니라 같이 섭취하는 다른 메뉴들도 중요하기 때문에 주메뉴 만을 추천하는 식단 추천 방법은 사용자들이 실제로 사용하기에는 무리가 있다.

그리하여 전문가적인 식단 구성 ontology와 대량의 사용자 데이터를 이용해 사용자에게 적절한 식단의 구성요소를 모두 추천하는 방법이 개발되었다[16]. 하지만 대량의 외부 데이터와 전문적인 지식이 꼭 필요하여 새로운 데이터로 확장하거나 응용하는데 문제가 있다. 따라서 본 연구실에서는 식단의 카테고리 정보만을 이용하고 비교적 대량의 외부 데이터를 사용하지 않으며 사용자의 식성에 맞고 적절한 칼로리를 지닌 식단을 추천할 수 있는 방법을 개발하였다.

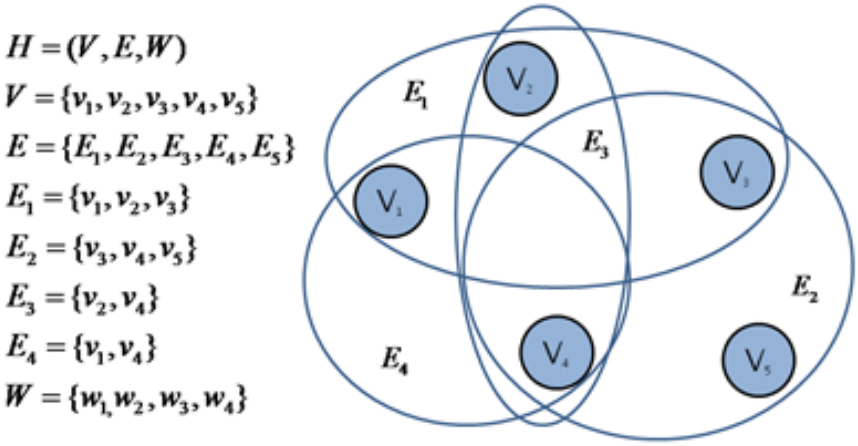

그림 2 하이퍼네트워크 구조. 하이퍼네트워크는 정점과 에지(edge) 그리고 에지의 가중치(weight)로 이루어진다. 에지는 여러개의 정점을 연결하는 하이퍼에지(hyper-edge)로 구성되어 있다. 여러 에지들은 가중치를 통해 데이터의 분포를 표현한다.

3.2 하이퍼네트워크

하이퍼네트워크는 다수 인자들 간의 고차원 연관 관계를 학습 할 수 있는 확률적 그래프 모델이다[17]. 이는 기존의 일반 그래프와는 달리 2개 이상의 정점을 동시에 연결 가능한 하이퍼에지(hyperedge)로 모델로서 하이퍼네트워크에서 정점은 데이터를 구성하는 인자(attribute)를 의미하며 하이퍼에지는 가중치가 할당된 인자들간의 고차원적 조합을 의미한다. 수식적으로 하이퍼네트워크 H 는 $H = (V, E, W)$ 로 각각 V, E, W 는 정점, 하이퍼에지, 가중치들의 집합으로 정의된다. 하이퍼네트워크의 기본적인 모양은 그림2에서 확인할 수 있다. k-차수(cardinality)의 하이퍼에지를 k-하이퍼에지라 하며 모든 하이퍼에지가 k-차수

하이퍼에지들로만 구성된 하이퍼네트워크를 k-하이퍼네트워크라 한다. 하이퍼네트워크는 이론적으로 데이터를 저장하기 위한 확률적 메모리 모델로 간주될 수 있기 때문에 k-하이퍼네트워크의 에너지는 다음과 같은 식으로 정의될 수 있다.

$$\varepsilon\left(x^{(n)};W\right) = -\sum_{i=1}^{|E|} w_i^{(k)} I(x^{(n)}, E_i)$$

위의 식에서는 i번째 k-차수 하이퍼에지 Ei의 가중치이고, x(n)는 하이퍼네트워크에 저장된 n번째 데이터 패턴이며, I(x(n), Ei)는 하이퍼에지 Ei 에서 x(n)의 요소 조합을 나타낸다. 이 때, 하이퍼네트워크로부터 데이터가 생성될 확률은 Gibbs분포 가정 하에 다음과 같이주어진다

$$\mathrm{P}\left(x^{(n)}\middle|W\right) = \frac{1}{Z(W)}\exp\{-\varepsilon\left(X^{(n)};W\right)\}$$

Z(W)는 부분 함수로 Z(W)는 아래와 같은 식으로 정의된다.

$$\mathrm{Z(W)} = \sum_{x^{(n)}\in D} \exp\{\sum_{i=1}^{|E|} W_i^{(k)} I(X^{(m)}, E_i)\}$$

즉, 하이퍼네트워크는 학습을 통해 랜덤 변수 조합의 확률 분포를 추정하는 모델이며. 모델 학습은 진화기법을 기반으로 한 구조학습과 파라미터 학습에 의해 진행된다. 하이퍼네트워크 모델은 텍스트나 이미지 분석뿐만 아니라 바이오인포메틱스나 멀티모달 데이터 분석등 여러 분야로 응용된다[18].

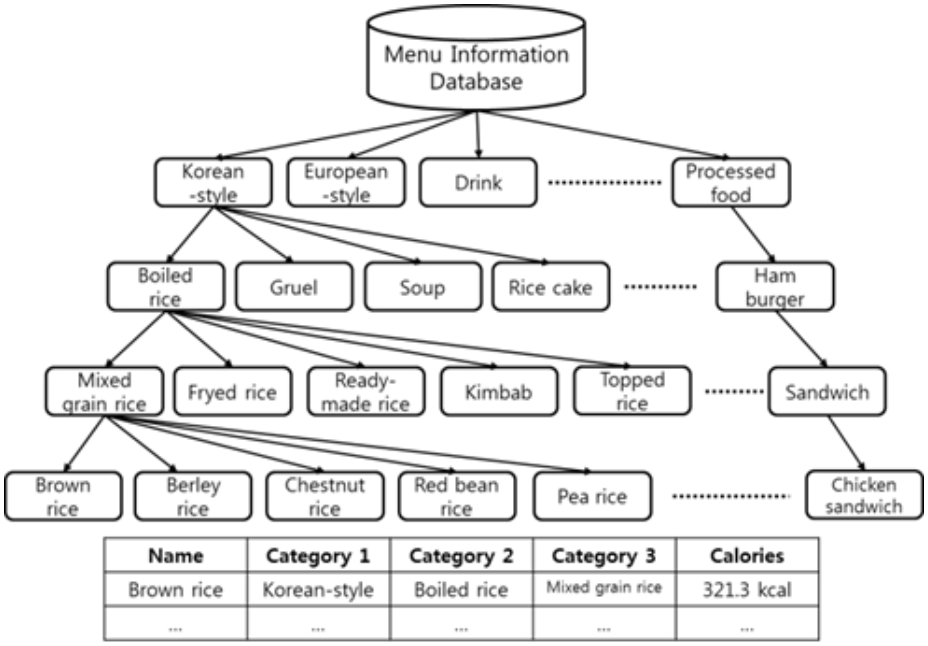

그림 3 음식 정보 데이터베이스의 구조와 음식 데이터의 형태. 음식 데이터베이스는 8천여개의 음식 정보를 가지고 있다. 각각의 음식은 3단계의 카테고리를 통해 분류된다. 음식은 자신이 속한 카테고리 정보와 칼로리 정보를 가지고 있다.

3.3 식단 데이터

학습에 사용되는 식단 데이터는 음식 데이터베이스의 정보를 바탕으로 저장되어 있다. 음식 데이터베이스는 식품영양청에서 조사한 식품영양분석표의 음식 분류와 칼로리 정보를 바탕으로 작성되었다[19]. 음식 데이터베이스와 음식 정보는 그림3과 같이 저장되어 있다. 각 음식 정보들은 음식명과 음식의 카테고리 정보, 칼로리 정보를 담고 있다. 음식의 카테고리 정보는 3단계의 카테고리로 이루어져 있다. 첫 번째 카테고리는 총 10개로 나뉘어져 있으며, 첫 번째 카테고리의 하위 카테고리로 이루어진 두 번째 카테고리는 60가지, 세 번째 카테고리는 176가지로 나뉘어져

있다. 음식의 종류는 총 7987가지 이다.

3.4 식단추천 알고리즘

본 논문에서는 하이퍼네트워크 모델을 식단 추천 알고리즘으로 응용하였다. 하이퍼네트워크는 데이터의 인자간의 연관관계를 통해 전체 데이터의 분포를 학습하므로, 식단 메뉴 아이템간의 연관관계가 중요한 식단 추천 알고리즘에 있어서 매우 적합한 방법이다.

식단 추천 알고리즘은 기존의 식단 로그를 학습한 하이퍼네트워크 모델을 가지고 있으며, 식단 로그가 24시간씩 새로 쌓일 때마다 하이퍼네트워크 모델을 재학습하는 과정을 거친다. 하이퍼네트워크 모델의 재학습 과정은 알고리즘1에 설명되어 있다. 이렇게 하이퍼네트워크 모델은 같이 섭취한 품목들간의 가중치를 학습함으로써 사용자가 어떤 음식들을 함께 먹는지 학습하게 되고, 결국 하이퍼네트워크 모델이 학습한 하이퍼에지와 에지의 가중치를 바탕로써 사용자의 식성을 표현할 수 있다. 학습된 하이퍼네트워크 모델을 이용한 식단 추천 방법은 알고리즘2와 같이 설계되었다.

위와 같은 식단 추천 알고리즘을 통해 얻어진 추천식단은 사용자가 즐겨 같이 먹는 품목으로 이루어져 있어 사용자의 식성을 잘 반영한다. 또한 품목별로 기록된 칼로리를 통해 사용자가 과다한 칼로리를 섭취하지 않도록 식단을 제한할 수 있다.

알고리즘 1. 하이퍼네트워크 모델 학습방법

1. 모아진 식단 로그 L을 하루치 식단 로그 $daily_L$로 분류한다.

2. $daily_L$에서 hyper-edge e를 무작위로 반복 생성한다.

1) 만약 H의 edgeset E가 e를 포함하지 않는다면

$E \leftarrow E \cup e$

2) e의 weight w를 증가시킨다.

3. 2의 과정을 모든 $daily_L$에 대하여 반복 수행한다.

알고리즘 2. 하이퍼네트워크를 이용한 식단 추천 방법

1. 학습된 하이퍼네트워크 H와 사용자의 적정 섭취 칼로리량 c_limit을 정의한다.

2. H의 edgeset E에서 하나의 edge e를 weight W의 비율에 맞춰 무작위로 선택한다.

1) 선택된 e에서 하나의 아이템 i을 무작위로 선정한다.

2) 추천된 메뉴 M에 i를 추가한다.

3. E에서 i를 포함하는 edge를 찾아 E_sub로 정의한다.

4. E_sub에서 W 비율에 맞춰 하나의 e를 무작위로 선택한다.

5. e에서 i이외의 아이템을 j로 정의한다.

1) 만약 j가 M에 이미 있다면 4로 돌아가 새로운 j를 선정한다.

2) j를 M에 추가하고, j를 새로운 i로 정의한다.

6. M의 item 들의 칼로리 합이 c_limit을 넘는지 체크한다.

1) 넘는다면, 추천된 식단으로 M을 제공한다.

2) 넘지 않는다면, 3으로 다시 돌아간다.

4. 식단 추천 알고리즘 성능평가

실험에 사용하기 위하여 10명의 사용자를 통해 식단 로그 데이터를 수집하였다. 실험자 10명을 20일동안 자신이 섭취한 음식을 모두 식단 로그로 기록하였다. 실험 참가자중에 7명은 남자 3명은 여자이고, 1명은 채식주의자 9명은 일반인이다. 수집된 데이터 중에 1명은 초기 하이퍼네트워크 모델을 학습시키는데 사용하고, 나머지 9명을 통해 비교실험을 수행하였다.

개발된 식단 추천 알고리즘이 실험자의 식성을 잘 반영하여 사용자에게 적절한 정보를 추천하는지 평가하기 위해, 사용자가 자신의 식단 정보를 입력함에 따라 다른 사용자와 추천 받는 식단이 얼마나 차이가 나는지 알아보기로 하였다. 개발된 알고리즘이 사용자의 로그를 학습하여 사용자의 식성을 반영한다면, 사용자가 식단 로그를 전혀 입력하지 않은 초기에는 모든 사용자가 비슷한 메뉴를 추천 받을 테지만 사용자의 자신의 식단 로그를 입력함에 따라 하이퍼네트워크 모델이 변화해 사용자의 특징이 드러나는 식단을 추천 받게 될 것이다. 먼저 개발한 알고리즘이 사용자의 스타일을 얼마나 따라하는지 알아보도록 하겠다. 실험자가 입력하는 식단과 그에 따라 추천 받는 식단이 어떻게 변하는지 비교해보았다. 그림 4 은 4번 실험자가 식단을 입력하는 날이 지나감에 따라 자신이 입력한 식단과 추천 받은 식단의 스타일의 변화를 나타낸 것이다. 각 바 형태의 그래프에서 가로축은 입력받거나 추천받은 식단의 카테고리 분포를 나타낸것이고, 세로축은 각 카테고리의 비율이다. 하나의 카테고리의 왼쪽 바는 입력받은 식단의 카테고리 분포, 오른쪽 바는 추천된 식단의 카테고리 분포를 나타낸다. 5개의 그래프중 첫번째는 아무것도 입력하지 않은 0일째, 그 다음으로 순서에 따라 5일째, 10일째,

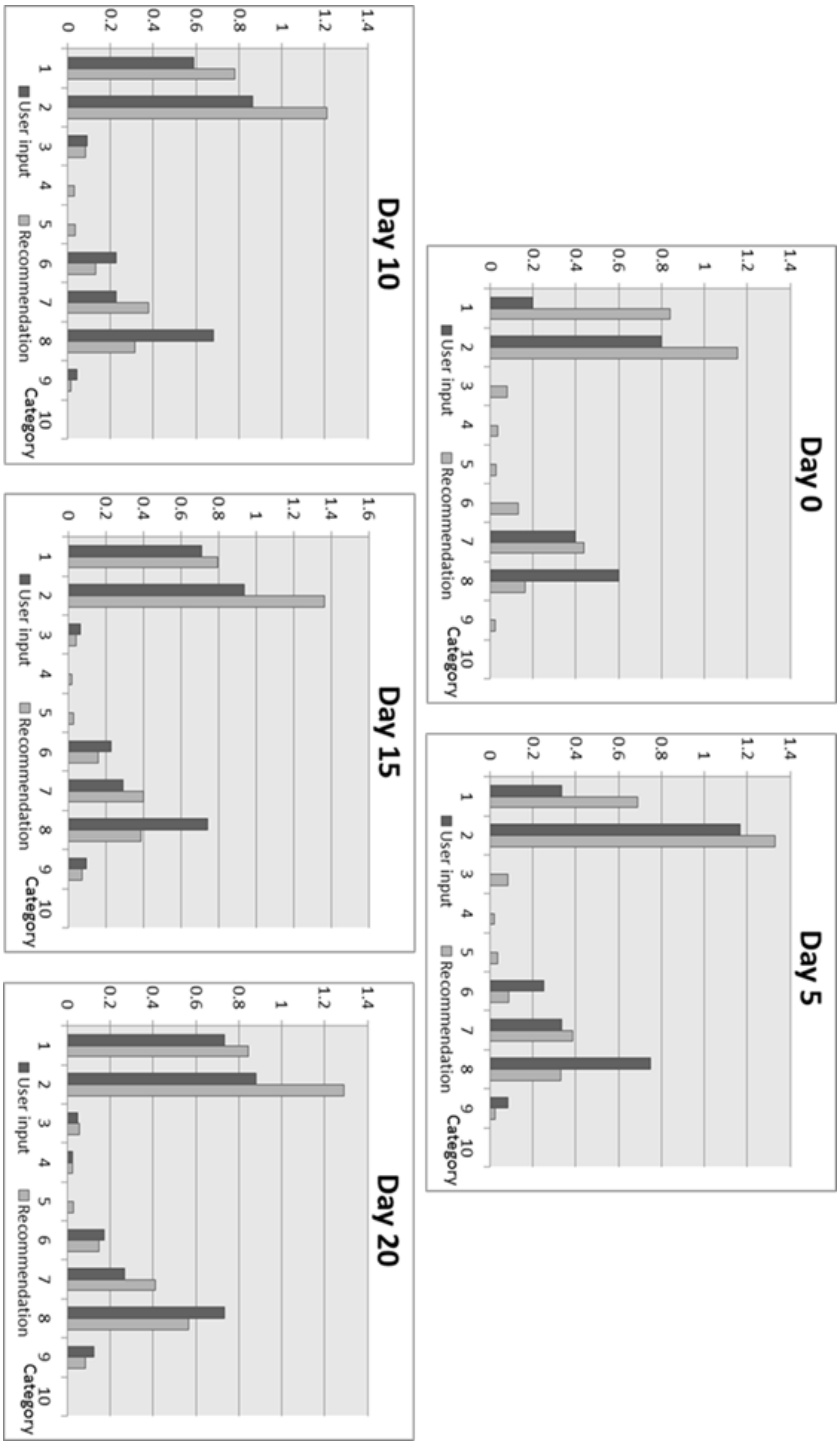

그림 4 날짜별 입력 식단과 추천 식단의 분포. 실험자4의 날짜별로 사용자가 입력한 식단과 추천 받는 식단의 카테고리 분포를 비교한 것이다. 식단 추천 알고리즘의 사용자의 데이터를 학습함에 따라 입력 식단과 추천 식단의 분포 모양이 비슷해지는 것을 확인할 수 있다.

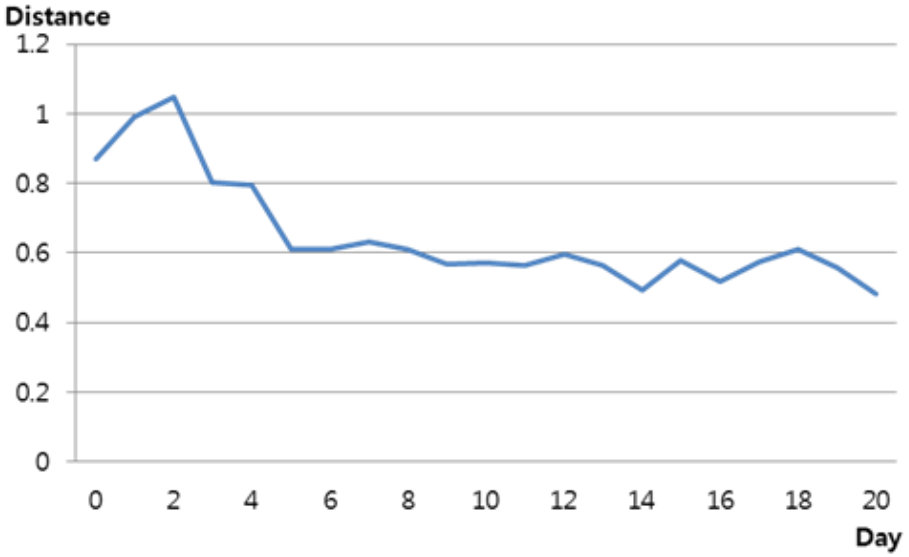

그림 5 실날짜별 입력 식단과 추천 식단의 분포 차이. 그림4에서 보여준 실험자 4의 입력 식단과 추천 식단의 분포 차이를 날짜 별로 나타낸 것이다. 분포 차이가 시간이 지남에 따라 감소하는 것을 확인할 수 있다.

15일째, 20일째의 상태를 나타낸 것이다. 순서에 따라 그래프를 살펴보면 시간이 지남에 따라 좀더 그래프의 모양이 비슷해지는 것을 볼 수 있다. 그림 5 을 보면 더 자세히 볼 수 있는데 각 날짜별로 입력한 식단과 추천 받은 식단의 카테고리 분포의 거리차이를 그래프로 나타낸 것이다. 날짜가 지나감에 따라 점차적으로 입력한 식단과 추천 받은 식단간의 거리차이가 줄어드는 것을 볼 수 있다. 실제로 하이퍼네트워크 모델이 사용자가 입력한 식단을 학습하며 변화하는 것을 그림 6에서 변화하는 하이퍼네트워크 모델을 통해 확인할 수 있다. 그림 6은 실험자 한명에 관하여 시간이 지남에 따라 변화하는 중요 하이퍼에지를 나타낸 것이다. 하이퍼네트워크 모델은 사용자가 입력하는 데이터에 따라 학습하여 시간이 지남에 따라 사용자 식성에 맞춰 학습해 가는 것을 확인할 수 있다. 이렇게 사용자의 입력에 따라 변화하는 하이퍼네트워크 모델을 통해 사용자 취향에 맞는 식단을 추천 할 수 있다.

또한, 사용자 별로 자신의 정보를 입력함에 따라 얼마나 추천

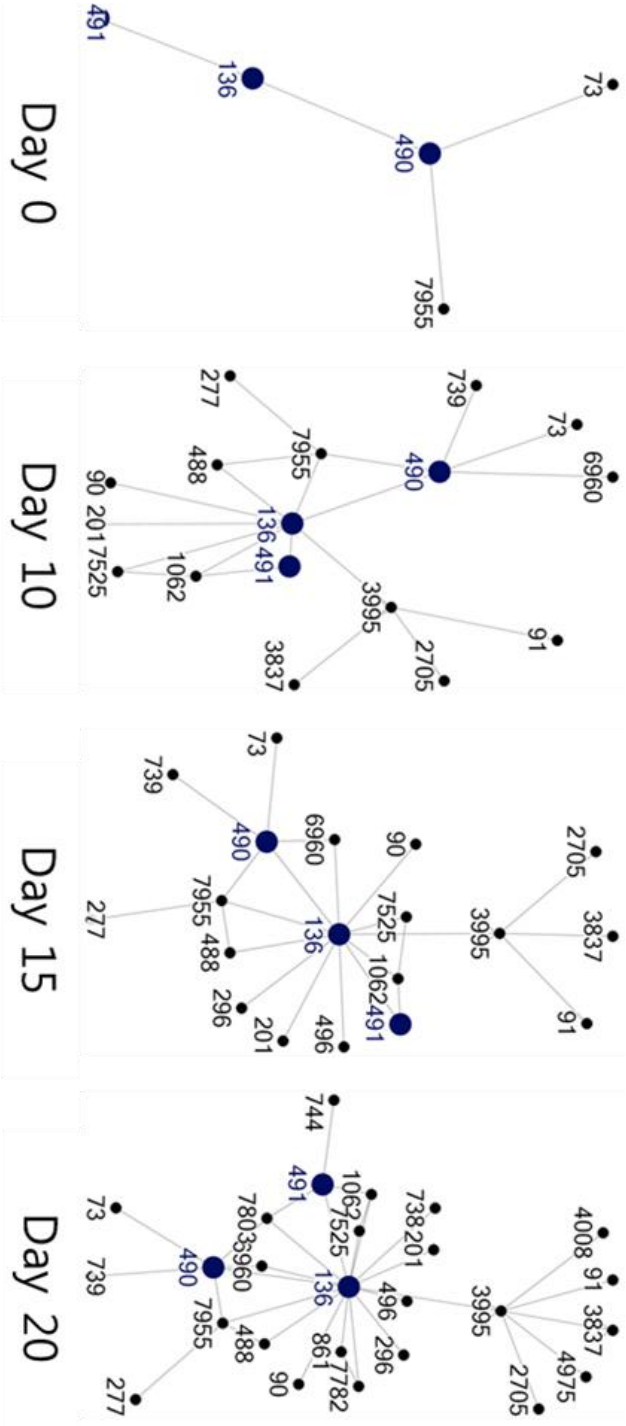

그림 6 날짜별 하이퍼네트워크 모델의 주요 에지들의 변화. 실험자의 데이터가 입력됨에 따라 에지들은 변화하며 사용자의 취향을 학습해간다. 변화된 에지들은 사용자의 이전보다 더 사용자 취향에 맞춘 식단 추천을 가능하게 한다.

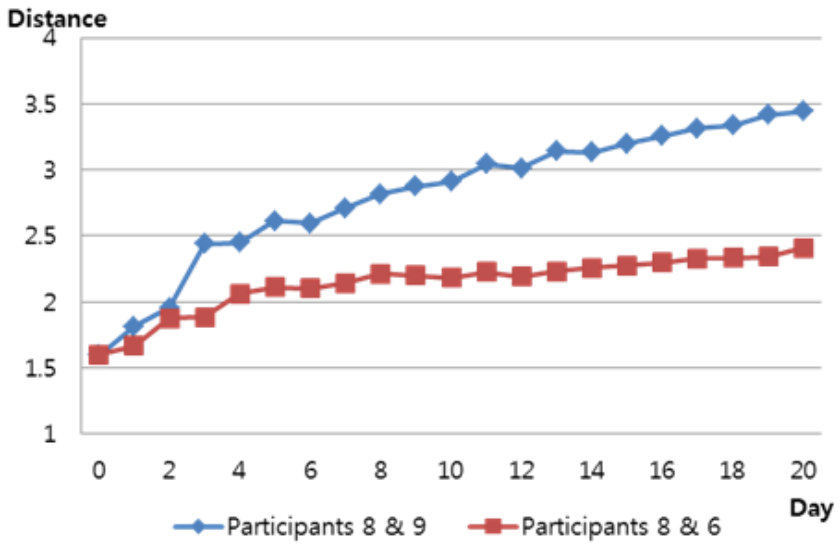

그림 7 날짜별 실험자 6 & 8와 실험자 8 & 9 간의 추천 식단 거리차이. 실험자간의 차이가 별로 없는 실험자 6 & 8의 결과와 달리 실험자 8 & 9는 시간이 지남에 따라 추천 받는 식단의 거리가 점점 벌어지는 것을 확인할 수 있다.

결과가 달라지는 지를 알아보기 위하여 사용자 간의 추천 식단 거리를 계산하였다. 추천된 식단간의 거리는 추천된 메뉴를 세번째 카테고리로 치환하여 치환된 각각의 세번째 카테고리끼리의 L1-distance로 정의하였다. 식단 추천 알고리즘으로 생성되는 추천 식단은 확률적으로 결정되기 때문에, 비교실험을 정확하게 진행하기 위해 반복 실험의 결과를 평균하여 사용하였다. 두명의 식단 추천 결과를 바로 비교하는 것이 아니라, 추천 결과를 100개씩 모아 평균하여 평균된 추천 메뉴끼리의 차이를 비교하였다. 이렇게 평균된 추천 메뉴끼리의 비교 또한 임의성을 줄이기 위해 100번의 반복 비교 결과를 평균내었다.

그림 7에서 실험자 8와 실험자 9간의 비교 결과와 실험자 8와 실험자 6간의 비교 결과를 확인할 수 있다. 실험자 8와 실험자 9 실험결과는 실험자 8와 비교한 모든 사람중에 가장 차이가 큰

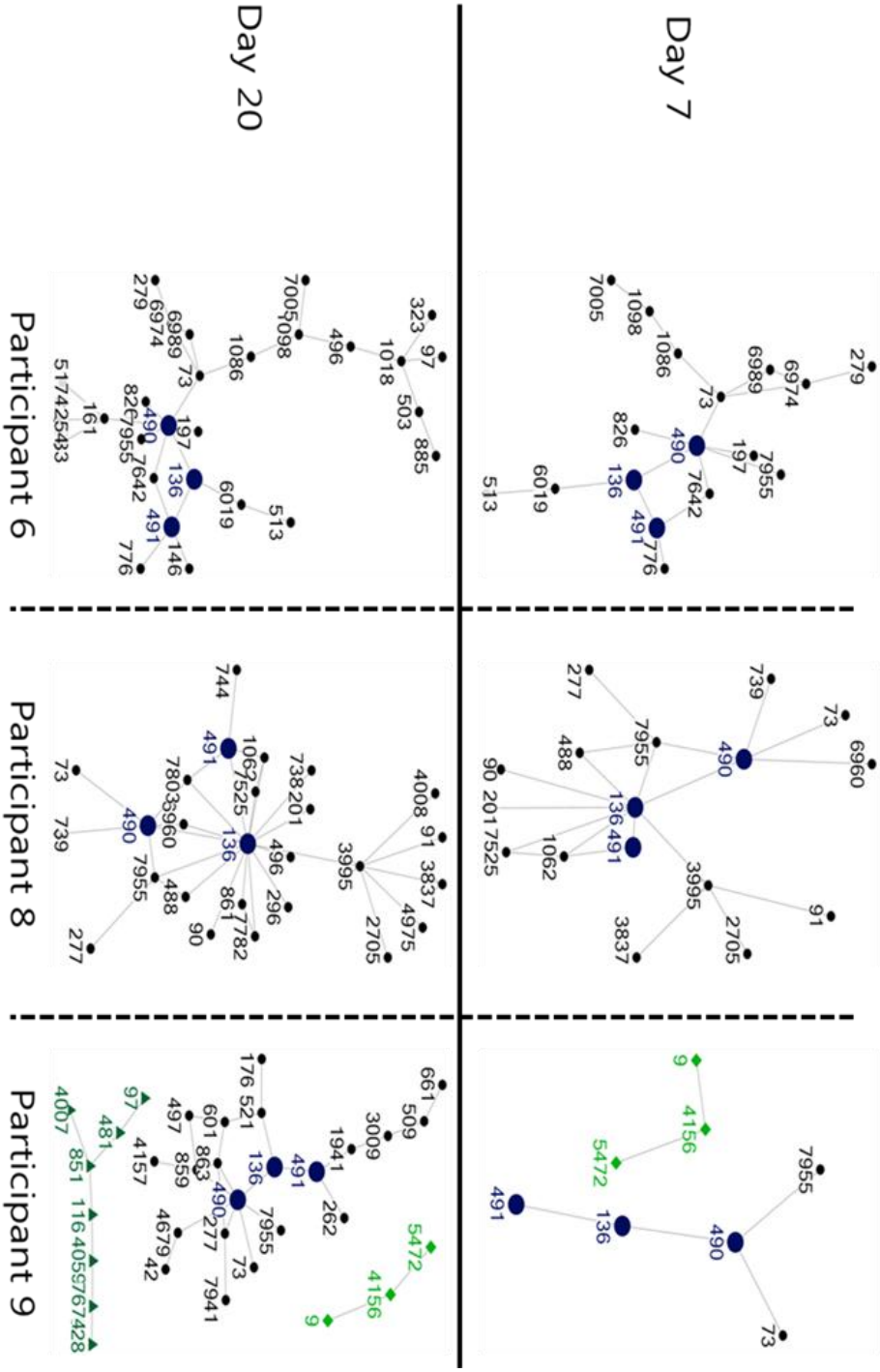

그림 8 실험자 6,8,9의 하이퍼네트워크 모델 변화. 실험자 6&8의 경우 136,490,491 을 중심으로 하는 하이퍼에지 연결을 중심으로 비슷한 음식들이 주요 에지들로 연결되어 있는것을 확인할 수 있다. 반면 실험자 9는 136,490,491을 중심으로 하는 연결 이외에도 다른 음식들로 연결되어있는 하이퍼에지 그룹이 나타나는 것을 확인 할 수 있다.

번호	136	490	491	851	5472
음식명	쌀밥	배추김치	깍두기	김	두유

표 2 그림 6과 8의 주요 에지 음식명

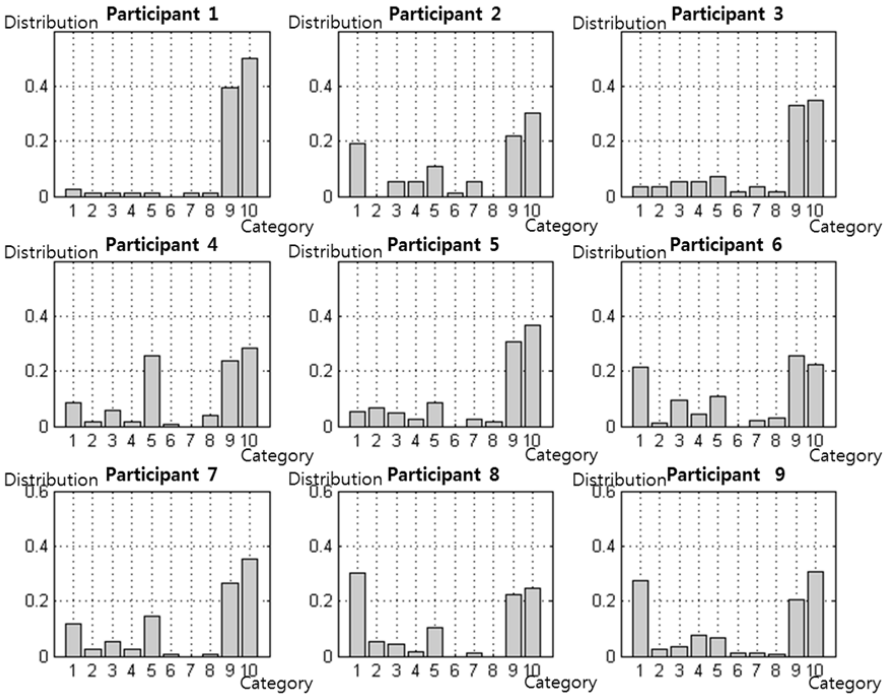

그림 9 사용자 별 추천 식단 분포. 실험에 참여한 모든 실험자들의 각 20일째 추천 받는 식단의 카테고리 분포를 나타낸 것이다. 각각의 사용자들이 서로 다른 분포의 식단을 추천 받음을 알 수 있다.

실험결과이고, 실험자 8와 실험자 6의 실험결과는 가장 차이가 작은 실험결과이다. 실험자 8와 실험자 9의 실험결과는 메뉴간의 거리차이가 1.59에서 3.44로 총 1.84만큼 변화하였고, 실험자 8와 실험자 6의 실험결과는 메뉴간의 거리차이가 1.59에서 2.40로 총 0.81만큼 변화하였다. 두 결과를 통해 고안된 식단 추천 알고리즘의 사용자의 정보가 입력 될수록 사용자의 식성에 따라 추천하는 식단이 비슷한 사람끼리는 비슷하게 다른 사람과는 다르게 변한다는 것을 알 수 있다. 그림 8은 실험자 6, 8, 9의

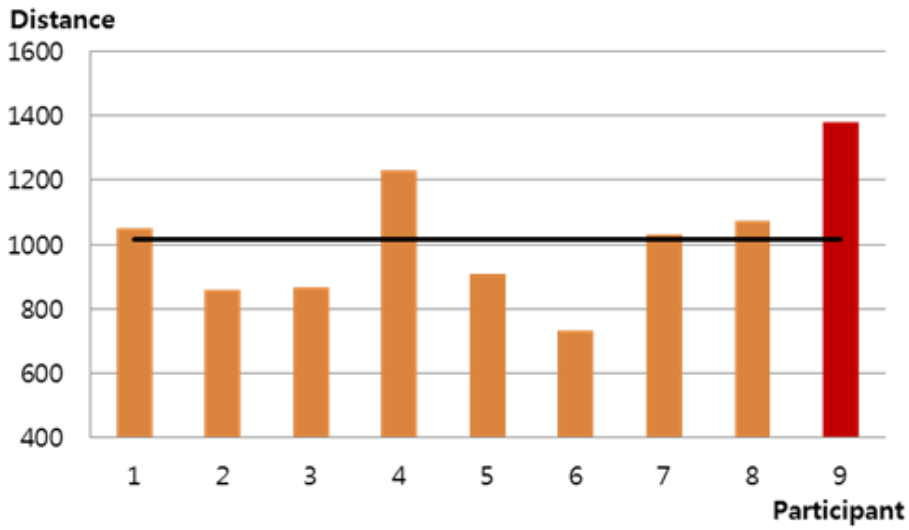

그림 10 실험자 별 타 실험자와의 추천 식단 거리 차. 그림 9의 사용자별 추천 식단 분포그래프에서 실험자 별 타 사용자와의 분포 차이를 합한 것이다. 실험자 9가 다른 모든 사용자와 가장 다른 식단을 추천 받음을 확인 할 수 있다.

하이퍼네트워크의 에지들을 그려놓은 것으로 실제로 실험자 6 & 8이 비슷하고, 실험자 8 & 9가 다른지를 확인할 수 있다. 실험자 6과 8은 490번 노드와 136번 노드를 중심으로 여러 노드들이 크게 하나로 연결되어 있는 것을 확인할 수 있다. 그에 비해 실험자 9는 490번 노드 이외에도 851번 노드를 중심으로한 다른 집단의 연결이 발견되는 것을 직접 눈으로 확인 할 수 있다.

실험자들의 식단 추천 결과를 전체적으로 살펴보면 그림 9과 같다. 각 실험자 별로 입력한 식단과 추천받은 식단의 카테고리 분포를 나타낸 그래프이다. 그래프를 통해 실험자들은 전체적으로 조금씩 다른 식습관을 가지고 있음을 알 수 있다. 전체 실험자를 대상으로 개발된 알고리즘이 실험자의 특성에 맞춰 식단을 추천 것을 확인할 수 있다. 그림 10 은 그림 9의 실험자 별 추천 식단 그래프를 바탕으로 어떤 실험자가 타 실험자들실 비교하였을 때 가장 다른 식단을 추천 받는지 나타낸 그래프이다. 실험자 9가 가장

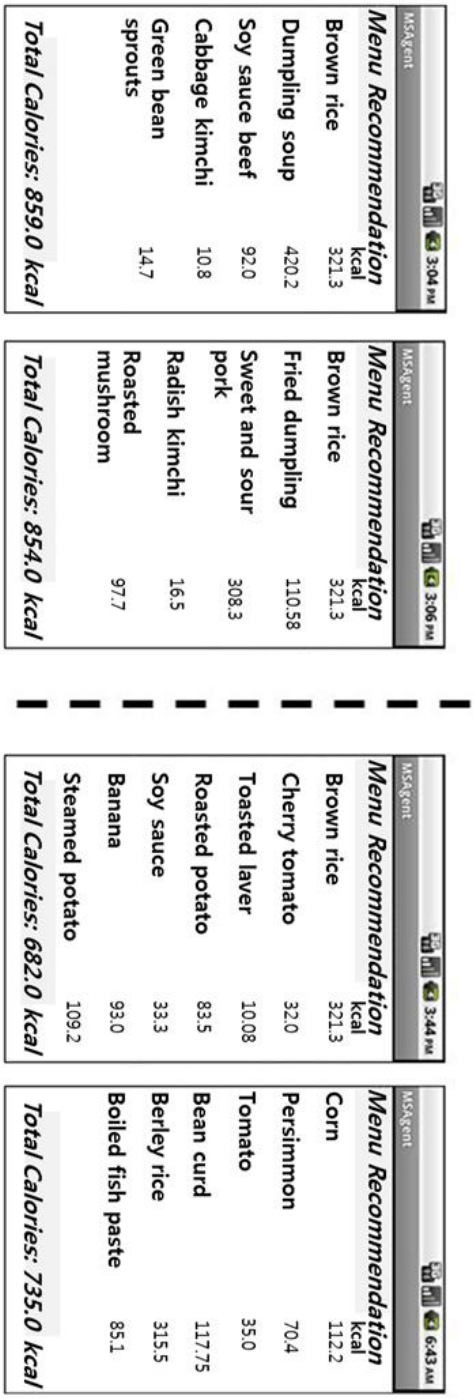

그림 11 실험자 8(좌)와 실험자 9(우)의 추천 식단 예제. 실험자 8과 9의 실제 어플리케이션에서 추천 받는 식단을 캡쳐한 것으로, 실험자 9가 추천 받는 식단은 채식주의자로서 육류가 포함되지 않는 것을 확인할 수 있다.

타 사용자들에 비해 다른 식단을 추천 받는 것으로 나타났다. 실제로 실험자 9는 채식주의자로써 다른 실험자들과 달리 홀로 육식을 하지 않는 특징을 가지고 있어, 가장 다른 식단을 추천 받는다. 그림 11에서 실제로 확인 할 수 있는데 채식주의자인 실험자9과 일반적인 실험자의 하나인 실험자 8의 식단 추천 결과 예시 화면이다. 실제 추천 결과를 살펴보면 실험자 9의 경우 채식주의자 답게 추천 받은 식단에 육류가 없어, 다른 사용자의 추천 결과와 큰 차이가 있음을 직접 확인 할 수 있다.

이처럼 개발된 식단 추천 알고리즘은 사용자가 자신이 먹은 식단을 입력함에 따라 사용자의 식성에 맞춰 식단을 추천할 수 있다. 또한, 데이터를 학습한 하이퍼네트워크모델을 통하여 데이터 분포의 특성을 직접 확인하고 분석할 수 있다.

5. 결론 및 향후 연구

현대인의 건강을 위해선 체중 조절은 간과할 수 없는 부분이지만, 전문지식이 없는 일반인이 자신의 식단을 분석하고, 적절한 칼로리의 식단을 추천하는 것은 매우 요원하다. 하지만 본 논문에서 설명한 하이퍼네트워크를 이용한 식단 추천 알고리즘을 이용하면, 자신의 식단을 입력하면 자신의 식성을 잘 표현하며 적절한 칼로리를 지닌 식단을 추천 받을 수 있다. 개발된 식단 추천 알고리즘은 사용자가 자신의 식단을 꾸준히 입력하면 사용자의 식성을 잘 표현할 수 있고 이를 통해 사용자는 개인화된 식단 추천을 받을 수 있다. 이후로는 사용자의 피드백과 자신과 비슷한 타인의 데이터를 같이 이용하여 식단 추천에 이용하는 방법을 개발하고자 계획하고 있다.

또한 식단 추천 알고리즘을 적용한 DietAdvisor 어플리케이션을 사용하면 사용자는 체중 관리에 필요한 전반적인 부분에서 서비스를 제공 받을 수 있으며, 스마트폰 센서를 이용한 자동화된 데이터 수집 방식으로 다양한 서비스를 얻기 위해 번거로운 조작을 할 필요가 없이 도움을 얻을 수 있다.

DietAdvisor는 현재 안드로이드os 버전으로 제작되었으며, DietAdvisor에 관한 자세한 정보는 http://bi.snu.ac.kr/Research/DietAdvisor/ 에서 확인할 수 있다.

DietAdvisor는 사용자들의 편하게 사용할 수 있도록 제작되었지만 추천되는 운동이 걷기와 달리기로 제한되고, 항상 스마트폰을 소지해야하는 불편함이 남아있다. 이를 해결하기 위해 웹캠을 통한 사용자의 실내에서 스마트폰을 소지하지 않아도 사용자의 데이터를 수집하고 운동상태를 파악할 수 있도록

연구하여, 외부센서와 스마트폰을 연결한 유비쿼터스적인 건강 관리 어플리케이션을 개발할 계획을 가지고 있다.

참고 문헌

[1] OECD, “OECD Health Data 2011", 2011.

[2] American Diabetes Association, "Standards of Medical Care in Diabetes-2007", Diabetes Care, Vol. 30, no. 1, pp.S4-S41, 2007.

[3] 임병권, 김지섭, 유준희, 장병탁, “DietAdviser: 스마트폰 상에서의 개인 건강 관리 어플리케이션", 한국정보과학회 가을학술발표 논문집, Vol. 38, no. 2D, pp.115-118, 2011.

[4] G. Linden, B. Smith and J. York, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering", IEEE Internet Computing, Vol. 7, pp.76-80, 2003.

[5] J. Balintfy, "Menu Planning by Computer", Communications of the ACM, Vol. 7, pp.255-259, 1964.

[6] E. Eckstein, "Menu Planning by Computer: the random approach", JaAm Dieat Assoc, Vol. 51, pp.529-533, 1967.

[7] N. Yang, "An Expert System on Menu Planning [master's thesis]", Case Western Reserve University, 1989.

[8] K. Ganeshan, J. Farmer, "Menu Planning system for a Large Catering Corporation", In: Proceedings of the Third International Conference on the Practical Application of Prolo, pp.262-265, 1995.

[9] G. Balázs, V. István, K. György, An Evolutionary Divide and Conquer Method for Long-Term Dietary Menu Planning, Artificial Intelligence in Medicine:Lecture Notes in Computer Science, Vol. 3581, pp.419-423, 2005.

[10] M. Pazzani, A Framework for Collaborative, Content-Based, and Demographic Filtering, Artificial Intelligence Rev, pp.393-408, 1999.

[11] M. Balabanovic, Y. Shoham, "Fab: Content-Based, Collaborative Recommendation", Communications of the ACM, Vol. 40, no. 3, pp. 66-72, 1997.

[12] H. Si, Y. Kawahara, H. Kurasawa, H. Morikawa, and T. Aoyama. "A Context-Aware Collaborative Filtering

Algorithm for Real World Oriented Content Delivery Service", In Ubicom2005 Metapolis and Urban Life Workshop, 2005.

[13] J. Aberg, "Dealing with Malnutrition: A Meal Planning Systems for Elderly", AAAI Spring Symp. on Argumentation for Consumers of Health Care, pp. 1253-1258, 2006.

[14] J. Freyne, S. Berkovsky, "Recommending Food: Reasoning on Recipes and Ingredients", In: Proceedings of UMAP, Vol. 6075, pp.381-386, 2010.

[15] A. Felfernig, G. Friedrich, L. Thieme, "Recommender Systems", IEEE Intelligent Systems, Vol. 22, no. 3, pp.18-21, 2007.

[16] C. Lee, M. Wang, H. Hagras, "A Type-2 Fuzzy Ontology and its Application to Personal Diabetic-Diet Recommendation", IEEE Trans Fuzzy Systems, Vol. 18, no. 2, pp.374-395, 2010.

[17] B.-T. Zhang, “Hypernetworks: a molecular evolutionary architecture for cognitive learning and memory”, IEEE Computational Intelligence Magazine, Vol. 3, no. 3, pp.49-63, 2008.

[18] 하정우, 김병희, 이바도, 장병탁, “잡지기사 관련 상품 연계 추천 서비스를 위한 하이퍼네트워크 기반의 상품이미지 자동 태깅 기법”, 2010 한국컴퓨터종합학술대회(KCC2010)논문집, Vol. 37, no. 1A, pp.123-124, 2010.

[19] 농촌자원개발연구소, "식품성분표, 제 7개정판", 2006.