유전자 알고리즘을 이용한 언어식별

전화성°, 정성원, 장길진, 오영환
한국과학기술원 전산학과

Language Identification using Genetic Algorithms

Hwaseong Jeon°, Sung-won Jung, Gil-Jin Jang, Yung-Hwan Oh
Spoken Language Lab., Dept. of Computer Science, KAIST
hsjeon@bulsai.kaist.ac.kr

요 약

본 논문에서는 통계적인 언어 모델을 이용하여 한국어, 중국어, 스페인어를 식별할 수 있는 언어식별기를 구현하고, 유전자 알고리즘을 이용하여 그 성능을 향상시키는 방법에 대하여 연구를 수행하였다. 언어모델은 통계적 모델의 하나인 바이그램(bigram)을 이용하였고, 유전자 알고리즘으로 각 바이그램에 최적의 가중치를 주는 방법을 제안하였다. 유전자 코드는 두 가지 방법으로 평가하였으며, 각각의 성능을 경험적(heuristic)으로 주는 가중치와 비교평가 하였다

1. 서론

언어식별이란 입력음성이 어느 언어인지를 판별할 수 있는 기술이며 음성인식기술과 더불어 최근 세계화와 함께 주목받고 있다. 언어식별시스템은 국제 전화망에 적용되기 위한 목적으로 연구되며, 현재 EU로 통합된 유럽에서 가장 활기 있게 연구가 추진중이다.

최근 언어식별을 위한 접근방법으로는 첫째, 음향학적 모델을 이용한 방법, 둘째 음소별 음향학적 모델과 음소결합법칙을 이용한 방법, 셋째 음소별 음향학적 모델과 음소결합법칙, 그리고 운율정보를 모두 이용한 방법이 있다. 보편적으로 두번째 방법이 가장 많이 쓰이며, 성능 또한 가장 뛰어나다. 여기에는 언어종속적인 방법과 언어독립적인 방법이 있다. 본 논문은 두번째 방법 중 언어독립음소인식기를 이용한 방법이다. 언어독립음소인식기에서 나온 음소열을 각 언어의 음소결합확률을 구하고, 최대값을 해당언어로 식별한다. 본 연구의 OGI DB (Oregon Garuduate Institute Database)는 음소가 아닌 7개의 유소집합으로 분할되어 있기 때문에 음소집합인식기라는 용어를 쓰도록 한다[1][2].

본 논문은 기존의 방법론을 한국어를 비롯한 동양권언어에 적용하고, 인식률을 향상시키는 방법을 제안한다. 시스템의 전체적인 구조는 발성음으로부터 음소를 인식할 수 있는 음소인식기와 인식된 음소열로부터 각 나라의 확률적인 언어적 특징을 이용하여 언어를 식별하는 언어모델(음소결합확률모델)로 이루어진다. 음소집합인식기는 OGI DB를 이용하여 학습한 후 다시 이를 이용하여 실험하였다. 음소결합모델을 이용한 언어식별 성능을 향상시키기 위해 음소결합모델에 주는 적합한 가중치 테이블을 만드는 방법으로 유전자 알고리즘을 이용하였으며, 경험적(heuristic)으로 가중치를 두는 방법과 비교평가[2][3][5][6][7]하였다.

논문의 구성은 다음과 같다. 2장에서는 언어식별 시스템에 대해서 살펴본다. 3장에서는 유전자 알고리즘을 이용한 음소결합모델 가중치에 대해 방법과 과정을 살펴본다. 4장에서는 실험 및 결과를 보인 후, 5장에서 결론을 맺는다.

2. 언어식별 시스템

2.1. OGI 다국어 음성 데이터베이스

OGI DB는 10개국어(영어, 아랍어, 불어, 독어, 한국어, 일어, 중국어, 스페인어, 인도어, 베트남어)에 대해 각각 여러 명의 본토 발음의 발성의 집합으로 이루어져 있다. 각 발성은 음소단위의 7개의 음소집합으로 분할되어 있다. 각 음소집합의 분류는 다음과 같다. 모음(VOC: vowel), 마찰음(FRIC: fricative), 파열음(STOP: stop-sound)과 공명의 위치에 따라 앞쪽공명음(PRVS: prevocalic sonorant), 중간공명음(INVS: inter-vocalic sonorant), 뒤쪽공명음(POVS: post-vocalic sonorant)으로 나뉘어지고, 그 밖의 묵음이나 배경잡음(CLOS: silence, background noise), 이러한 7개의 집합으로 구분된다.

2.2. 언어식별 시스템의 구조

언어식별 시스템은 크게 두가지 단계로 분류된다. 첫 번째는 발성음을 음소단위로 분할하고 인식하는 음소인식 단계이며, 두 번째는 음소인식기의 결과물인 음소열(phone sequence)의 각 언어의 통계적 모델에 대한 점수를 구하고, 최대값을 발성음의 언어로 식별하는 언어모델링 단계이다. 그림 1은 각 단계의 관계를 나타낸다[5].

언어모델은 이웃하는 단어사이의 연관성을 나타내는 정보이며, 각 나라의 언어적 특성이 반영된다. 언어모델의 종류에는 구구조 문법에 기반한 모델과 통계적 모델이 있으며, 본 연구에서는 통계적 모델 중 인접한 두 음소간의 결합 정보만을 이용하는 바이그램(bigram)을 사용하였다[4][6].

2.3. 바이그램의 구현과 점수산출에 의한 언어식별

바이그램은 범언어적인 음소집합인식기의 결과물인 음소집합열을 이용한 언어모델이다. 각 바이그램에는 다음 식과 같이 이전 음소를 기준으로 다음 음소가 나올 확률이 주어진다.

$$P(W) = P(w_1)\prod_{i=2}^{N} P(w_i \mid w_{i-1}) \qquad (1)$$

바이그램 수식을 이용한 점수산출은 우선 학습자료로부터 얻은 음소집합열을 이용해 7x7크기의 바이그램 확률 테이블을 구성한다. 그리고 입력음성의 음소집합열을 각 언어의 바이그램 확률 테이블에 적용시켜 사상되는 확률을 합하여 얻은 값이 가장 높은 언어를 해당언어로 식별한다.

3. 유전자알고리즘을 이용한 음소바이그램 가중치

3.1 유전자 알고리즘의 개요

유전자 알고리즘은 어떤 문제의 답을 찾거나, 혹은 그 문제에 존재하는 어떤 값을 최소화 혹은 최대화하는데 유전자의 개념을 적용하는 알고리즘이다. 문제에 대해 오답과 정답이 있을 수 있으며, 오답들은 얼마나 정답에 가까운 오답인지 평가할 수 있는 경우가 있다. 여러 개의 특징 변수들을 조절하여 어떤 시스템의

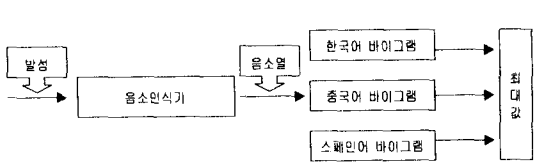

그림 1 . 언어식별기

성능을 최대화하고자 하는 경우, 가능한 변수 값들의 조합의 수는 굉장히 많다. 유전자 알고리즘은 존재할 수 있는 답의 경우들, 혹은 변수들의 상태를 각각 유전자 코드로 변환하여 가지고 있게 된다. 유전자 알고리즘은 진화와 퇴보의 과정이 있으며, 생각할 수 있는 여러 가지 답들을 어떤 방법을 통해 유전자 코드로서 생성한다. 하나의 답을 나타내는 유전자 코드들의 집합을 개체집단이라 한다. 만들어진 개체집단 내부의 각각의 유전자 코드들은 얼마나 정답에 가까운지, 좋은 답인지에 대한 평가 값을 갖는다. 개체집단으로부터 발전적인 다음 세대의 그것을 만들어내는 작업을 반복하는 것이 바로 유전자 알고리즘이다. 다음 세대의 개체집단을 만들어내는 작업은 재생산, 교배, 돌연변이 세연산으로 수행된다. 이 연산들으로 인해 한 세대의 개체집단으로부터 새로운 유전자 코드가 생성되며, 구 세대의 개체집단은 폐기된다. 어떤 세대에서 다음 세대로의 진화가 이루어지며, 이것을 특정 횟수 혹은 정답을 발견할 때까지 반복하며 수행한다. 유전자 알고리즘은 많은 변수의 수에 의한 탐색공간이 방대한 경우에 사용되며, 답을 찾는 정형화된 규칙이 존재하지 않거나, 쉽게 경험성을 적용할 수 없는 문제의 답을 최적화하여 찾는 방법으로 사용된다 [7].

3.2 유전자 알고리즘을 이용한 음소바이그램 가중치

본 논문에서 생성한 음소 바이그램은 7개의 음소 집합로 이루어져 있기 때문에 7x7 크기의 테이블로 표현된다. 이것을 3개국어에 대해 생성하였으므로 3개의 테이블이 존재한다. 각각의 음소 바이그램에 주고자 하는 가중치 테이블은 마찬가지로 7x7의 크기를 가지며, 가중치 값들의 총 수는 7x7x3 = 147개가 된다. 음소 바이그램을 사람이 직접 보고 경험(heuristic)에 의해 가중치를 주어 언어 식별 성능을 높이는 것은 쉽지 않은 일이다. 본 논문의 목적은 언어 식별 성능을 높이기 위한 음소 바이그램에 주는 가중치 테이블을 유전자 알고리즘의 최적화 능력으로 수행하고자 하는 것이다.

유전자 알고리즘을 사용하기 위해서는 문제를 유전자 알고리즘에 적용할 수 있는 형태로 변환하고, 유전자 알고리즘에 사용되는 알고리즘을 본 문제에 맞게 구현하는 작업이 필요하다. 이를 위해 가중치 테이블을 유전자 코드화 하였으며, 이를 평가하는 방법을 기술하였다. 이는 유전자 코드가 의미하는 가중치들을 음소 바이그램 확률계산에 적용하였을 때의 언어식별 능력을 기준으로 이루어졌고, 이렇게 코드화된 유전자들에게 적용시킬 수 있는 재생산, 교배, 돌연변이 작업을 정의하였다. 이러한 유전자 코드들과 유전자 알고리즘 작업들을 이용하여, 유전자 코드화된 가중치 테이블들로 이루어진 개체집단을 유전자들의 평가 값이 좋아지도록, 즉 언어식별 능력이 향상될 수 있는 방향으로 진화시키는 작업을 수행하였다.

3.3 음소바이그램에 대한 유전자 알고리즘 적용과정

3.3.1 초기 개체집단의 생성

유전자 알고리즘을 적용하기 위해서 우선 초기 개체집단, 즉 초기 유전자 코드들의 집합을 생성해야한다. 하나의 유전자 코드는 스페인어, 중국어, 한국어의 음소 바이그램에 대한 가중치 테이블 3개를 모두 포함한다. 모든 가중치 값들은 가중치가 1.0이 될 확률을 0.99, 가중치가 0.95가 될 확률을 0.01로 설정하였다. 기본적으로는 모든 음소바이그램에 가중치 1.0을 부여하도록 하였으며, 유전자 코드에 변이를 주기 위해 특정 확률(0.01)로 조금 틀린 값(0.95)의 가중치를 주었다. 초기 개체집단에 존재하는 이러한 조그마한 변이는, 세대를 반복해 나가면서 또 다른 변이를 계속 유발하여 결과적으로 보다 나은 가중치 테이블을 발견할 수 있는 시초가 된다. 7x7 크기의 가중치 테이블은 길이가 49인 하나의 일차원 배열로 바뀌고, 스페인어, 중국어, 한국어 3개 국어에 대한 가중치 테이블 각각을 의미하는 길이가 49인 3개의 일차원 배열이 차례로 연결되어 하나의 유전자 코드를 형성한다. 하나의 유전자 코드는 음소 바이그램에 주어질 수 있는 가중치이다. 본 실험에서는 이러한 유전자 코드 100개로 개체집단을 구성하였으며, 유전자 알고리즘의 수행에 따라 세대가 반복되어도 유전자 코드의 수는 일정하게 100개를 유지하도록 하였다.

3.3.2 유전자 코드의 평가

어떤 유전자 코드를 평가하는 작업은 유전자 코드를 3개 국어에 대한 가중치 테이블로 변환시키고 가중치 테이블의 각 가중치 값을 음소 바이그램의 값에 곱하여 새로운 음소 바이그램을 생성한 후 생성된 3개 국어의 음소 바이그램을 실험 데이터에 적용, 언어 식별 능력을 평가하는 것이다.

유전자 코드는, 하나의 언어에 대한 길이 49의 가중치 값의 일차원 배열이 3개가 모여서 이루어진다. 따라서 각각의 언어를 나타내는 부분을 분리하여 일차원 배열을 차례로 다시 2차원 배열 형태로 변환한다. 각 나라의 가중치 테이블은 하나의 유전자 코드로부터 생성되며, 언어 각각에 대해 음소 바이그램과 가중치 테이블을 얻는다. 가중치 테이블의 각 위치의 값들은 식 2와 같이 음소 바이그램의 동일한 위치에 대한 가중치를 의미한다.

$$B^{(i,j)}_{New} = B^{(i,j)} \times W^{(i,j)} \quad (2)$$

$B^{(i,j)}$ 는 원래 음소 바이그램의 i번째 행, j번째 열의 값을 의미하며, $W^{(i,j)}$ 는 가중치 테이블의 i번째 행, j번째 열 가중치 값이다. 이 과정을 통해, 가중치가 반영된 3개 국어에 대한 새로운 음소 바이그램을 형성하고, 이것을 실험 데이터에 적용하여 언어식별 능력으로 유전자 코드의 평가 값을 결정한다.

유전자 코드는 두가지 방법으로 평가하였다. 첫 번째는 데이터별 오류율 평균의 역수를 사용하는 것이다. 음소 바이그램을 언어식별에 사용하는 경우, 각 언어별로 어떤 데이터가 그 언어일 가능성이 있는지에 대한 상대적인 값을 보이는 언어를 식별한다. 여기에서, 어떤 데이터를 식별할 때의 오류율을 계산하는 방법을 다음과 같이 정의하였으며 오인식한 경우, 최대 평가값에서 원래 언어에 대한 평가값을 뺀 수치가 되며, 맞게 인식한 경우는 0이다. 예를 들어, 어떤 음소열이 중국어의 발성 데이터로부터 나온 것일 때, 그 데이터를 새로이 형성된 음소 바이그램에 적용하여 얻어진 값이 스페인어 0.54, 중국어 0.35, 한국어 0.87이라 하자. 이 경우 최대 값이 한국어에 대한 값 0.87이므로, 발성 데이터를 한국어로 오인식하게 된다. 따라서 이 경우 오류율은 0.87 - 0.35 = 0.52 가 된다. 반대로 이 데이터를 중국어로 맞게 인식한 경우 오류율은 0이 된다. 유전자 코드를 평가하기 위한 평가값은 식 3과 같이 보다 좋은 유전자에 대한 평가값이 보다 높은 값이 되도록 오류율의 역수로 정의하였다.

$$Eval(\text{유전자}_i) = \frac{1}{\text{전체 오류의 합} \;/\; \text{데이터의 수}} = \frac{1}{\text{평균오류}} \quad (3)$$

두 번째 방법은 언어식별 정확도를 사용하는 것이다. 유전자 코드를 이용하여 새로이 생성된 음소 바이그램을 이용한 언어식별 정확도 자체를 유전자 코드의 평가값으로 이용한다. 어떤 유전자 코드가 의미하는 가중치가 반영된 새로운 음소 바이그램을 모든 데이터에 적용하였을 경우, 각 언어별 정확도 값을 평균하여 유전자 코드의 평가값으로 사용한다. 예를 들어, 스페인어에

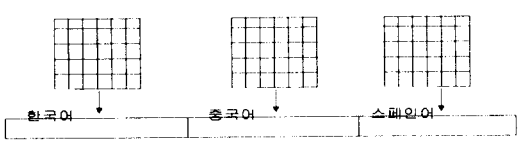

그림 2 . 유전자 코드의 형성

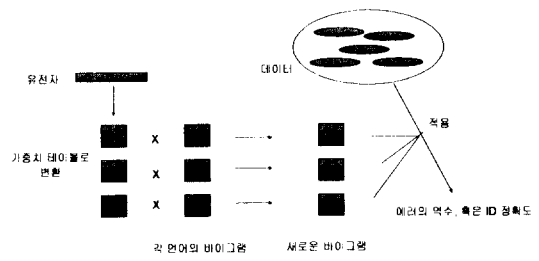

그림 3. 가중치가 반영된 새로운 음소바이그램 형성

대한 식별 정확도가 80%, 중국어에 대한 식별 정확도가 57%, 한국어에 대한 식별 정확도가 74%인 경우, 이 때의 유전자 코드에 대한 평가값은 세 정확도의 평균인 (80 + 57 + 74) / 3 = 70.3 이 된다.

본 연구에서는 전자와 후자의 방법을 실험 1과 실험 2로 명명하고 이의 두 가지 유전자 코드 평가값을 각각 따로 사용하여 2번의 실험을 수행하였다.

3.3.3 유전자 알고리즘 작업들의 정의

재생산은 어떤 세대의 개체집단으로부터 다음 세대의 개체집단을 만들기 위해 필요한 2개의 유전자 코드를 확률적으로 선택하는 작업이다. 각 유전자 코드들은 미리 정해진 방법에 의해 결정된 평가값을 가지며, 그 평가값에 비례하도록 재생산할 때 선택될 확률을 갖는다. 보다 높은 평가값을 갖는 유전자 코드일수록 다음 세대의 유전자 코드를 생성하기 위해 사용될 확률이 높게 된다. 교배는 재생산단계에서 선택되어 복사된 2개의 유전자 코드를, 각각 일부분을 잘라 맞바꿈으로써 새로운 2개의 유전자 코드를 생성하는 작업이다. 이 작업은 복사된 유전자 코드를 사용하여 이루어지므로 전 세대의 개체집단에 존재하는 유전자 코드들은 변경되지 않은 채로 남아 있다. 본 실험에서 사용한 유전자 코드는 각 언어별 가중치 부분이 구분되어 있다. 각각의 언어에 대해 서로 반대쪽 부분을 맞바꿔 새로운 가중치 부분 2개를 생성한다. 이 결과 각 언어별로 새로운 가중치 테이블을 의미하는 1차원 배열이 2개씩 생겨나게 되며, 이것을 언어별로 하나씩 이어서 새로운 유전자 코드 2개를 생성하게 된다.

돌연변이은 특정 확률로 유전자 코드에 변이를 생성하게 만든다. 이 작업에 의해 유전자 알고리즘은 극부 최소값에 빠질 가능성을 줄일 수 있으며, 유전자 코드의 집합인 개체집단에 새로운 유전자 정보가 삽입되는 것을 의미한다. 재생산과 교배작업을 통해 다음 세대의 개체집단에 포함되는 새로운 2개의 유전자 코드가 형성되어 있다. 유전자 코드의 i번째 값, 즉 어떤 언어에 대한 가중치 값에 변이를 적용하는 방법은 다음 식과 같다.

$$W_{New} = \begin{cases} W, & 0 \le P \le 0.95 \\ W + 0.05, & 0.95 < P \le 0.975 \\ W - 0.05, & 0.975 < P \le 1 \end{cases} \quad (4)$$

여기에서 P는 실수로 표현된 0과 1사이의 난수이다. 즉, 어떤 가중치 값을 0.95의 확률로 그대로 보존시키고, 0.025의 확률로 가중치 값을 0.05만큼 증가시키거나 감소시키는 것이다. 이 결과 새로운 가중치 값이 1이 넘거나 0보다 작아지는 것을 방지하기 위해 식 5을 덧붙였다.

$$W_{New} = \begin{cases} 0, & W_{New} < 0 \\ 1, & W_{New} > 1 \\ W_{New}, & \text{otherwise} \end{cases} \quad (5)$$

재생산, 교배, 돌연변이를 한번 적용하면 다음 세대를 위한 새로운 유전자 코드 2개가 생성되므로, 이의 반복적 적용을 통해 이전 세대와 동일한 수의 유전자 코드를 가지도록 다음 세대의 개체집단을 반복적으로 생성하여 보다 나은 유전자 코드를 찾아내는 작업을 수행한다.

4. 실험 및 결과

4.1. 실험조건

본 실험에서는 각 언어별로 음소열이 색인된 문장을 58개의 이용하여 음소 바이그램을 생성하였고, 그 데이터들을 그대로 유전자 코드의 평가에 사용하였다. 또한 개체집단에 존재하는 유전자 코드의 수는 100개로 하였으며, 500번째 세대까지 유전자 알고리즘을 수행하였다. 돌연변이가 발생할 확률은 앞에서 기술한 바와 같이 0.01로 정하였다.

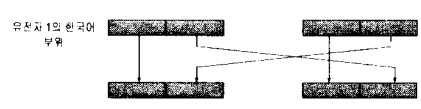

그림 4. 유전자 코드의 교배

4.2 실험방법

실험은 유전자 코드를 평가하는 방법에 따라, 실험 1(데이터 식별 오류를 줄이는 방향)과 실험 2(언어식별 정확도의 평균치를 증가시키는 방향)로 분류하였다. 이와 비교평가될 경험적(heuristic)으로 가중치를 두는 방법은 비슷한 점수대의 오인식률을 보완하기 위한 두가지 방법이 있다. 첫 번째는 이전음소를 기준으로 하여 최대값의 분포가 다른 나라와 다를 경우 큰 최대값에 1.5배를 한다. 다른 방법은 이전음소를 기준으로 하여 최대값의 분포가 다른 나라와 다른 이전 음소 행의 모든 값에 1.3배를 하는 것이다. 전자를 휴리스틱 1, 후자를 휴리스틱 2라 하겠다.

4.3. 실험결과

가중치를 음소 바이그램에 적용하지 않은 경우, 경험적으로 가중치를 적용한 방법1과 방법2, 유전자 알고리즘에 의해 수행된 실험 1과 실험 2를 종합적으로 비교한 결과는 표 1과 같다. 유전자 알고리즘을 적용한 결과, 전체적인 언어식별 능력이 일반적으로 향상되었음을 알 수 있다.

표 1. 실험결과

	스페인어	중국어	한국어	평균
가중치 없음	80.0%	56.9%	74.1%	**70.3%**
휴리스틱 1	89.1%	56.7%	56.7%	**67.5%**
휴리스틱 2	81.8%	39.7%	79.3%	**66.9%**
유전자 실험 1	81.8%	70.7%	69.0%	**73.8%**
유전자 실험 2	80.0%	72.4%	79.3%	**77.2%**

5. 결론

본 논문은 경험적인방법과 유전자 알고리즘을 이용한 가중치 부여를 통한 음소 바이그램을 각각 구현하였다. 이와 관련하여 음소바이그램을 이용한 언어식별 성능에 대한 실험을 수행하였다. 경험적으로 음소 바이그램에 가중치를 적용한 경우와 유전자 알고리즘을 이용하여 음소 바이그램에 가중치를 적용한 경우 모두 두가지 방법으로 실험을 하였다. 경험에 의해 전체적인 언어식별 시스템의 성능을 높이는 문제는 그리 간단한 것이 아니지만, 음소 바이그램에 유전자 알고리즘을 이용하여 가중치를 주는 것은 언어 식별 시스템의 성능을 높일 수 있었다. 현재 더 많은 언어에 대해서 효율적으로 가중치를 주는 방법과, 다양한 언어모델링에 방법에 대하여 연구가 진행중이다.

참고문헌

[1] Yeshwant K. Muthusamy, Etienne Barnard, and Ronald A. Cole, "Reviewing Automatic Language Identification," IEEE Signal Processing Magazine, pp.33-41, October 1994.

[2] Marc A. Zissman, "Comparison of Four Approaches to Automatic Language Identification of Telephone Speech," IEEE Transactions on ASSP, Vol.4, No.1, pp.31-44, 1996.

[3] C. Corredor-Ardoy, J.L. Gauvain, M. Adda-Decker, L. Lamel, "Language Identification with Language-Independent Acoustic Models," Eurospeech' 97, pp. 5-8, 1997.

[4] E.G. Schukat-Talamazzini, F. Gallwitz, S. Harbeck, and V. Warnke. "Interpolation of maximum likelihood predictors in stochastic language modeling," Eurospeech' 97, pp2731-2734, 1997.

[5] J. Navratil, W. Zuhlke, "An Efficient Phonotactic-Acoustic System for Language Identification," ICASSP'98, pp. 781-784, 1998.

[6] V. Warnke, S. Harbeck, E. Noth, H. Niemann, and M. levit. "Discriminative estimation of interpolation parameters for language model classifiers," ASSP'99, pp. 525-528, 1999.

[7] Z. Michalewicz, "Genetic Algorithms + Data Structures = Evolution Program," Copyright (c) Springer-Verlag Berlin Heidelberg 1996.