

독서 기본 : Actualize

Actualize [ˈæktʃuəlaɪz]

[타동사] 실현하다, 현실로 만들다

[VERB] to make actual or real

수능 국어의 새로운 기준

김민수

김민수

학력)

고려대학교 경영학과

약력)

오르비 클래스 국어영역 강사

김민수 수능국어 CLASS 운영

파라투스 국어 대표 강사

<빠른 정답>

[2013학년도 수능 43~45번]

- 43번 : ③
- 44번 : ③
- 45번 : ④

[2012학년도 수능 17~20번]

- 17번 : ④
- 18번 : ⑤
- 19번 : ①
- 20번 : ④

[2011학년도 수능 44~46번]

- 44번 : ③
- 45번 : ④
- 46번 : ③

[2011학년도 수능 32~36번]

- 32번 : ③
- 33번 : ③
- 34번 : ②
- 35번 : ⑤
- 36번 : ③

[2010학년도 9월 모의평가 13~17번]

- 13번 : ④
- 14번 : ②
- 15번 : ①
- 16번 : ④
- 17번 : ①

[2017학년도 6월 모의평가 16~19번]

- 16번 : ③
- 17번 : ⑤
- 18번 : ③
- 19번 : ③

[2017학년도 수능 16~20번]

- 16번 : ②
- 17번 : ④
- 18번 : ⑤
- 19번 : ⑤
- 20번 : ②

[2017학년도 수능 33~36번]

- 33번 : ⑤
- 34번 : ④
- 35번 : ①
- 36번 : ③

[2018학년도 6월 모의평가 30~34번]

- 30번 : ④
- 31번 : ③
- 32번 : ②
- 33번 : ⑤
- 34번 : ②

[2018학년도 9월 모의평가 27~32번]

- 27번 : ③
- 28번 : ④
- 29번 : ②
- 30번 : ⑤
- 31번 : ⑤
- 32번 : ③

[2018학년도 수능 27~32번]

- 27번 : ①
- 28번 : ⑤
- 29번 : ①
- 30번 : ④
- 31번 : ③
- 32번 : ②

[2019학년도 6월 모의평가 35~38번]

- 35번 : ③
- 36번 : ①
- 37번 : ④
- 38번 : ②

음성 인식 기술은 컴퓨터가 사람이 말하는 소리를 인식하여 해당 문자열로 바꾸는 기술이다. 사람의 말은 음소들의 시간적 배열로 볼 수 있다. 컴퓨터는 각 단어의 음소들의 배열을 ‘기준 패턴’으로 미리 저장해 두고, 이를 입력된 음성에서 추출한 ‘입력 패턴’과 비교하여 단어를 인식한다.

음성을 인식하기 위해서 먼저 입력된 신호에서 잡음을 제거한 후 음성 신호만 추출한다. 그런 다음 음성 신호를 하나의 음소로 판단되는 구간인 ‘음소 추정 구간’들의 배열로 바꾸어 준다. 그런데 음성 신호를 음소 단위로 정확히 나누는 것은 쉽지 않다. 이를 해결하기 위해 먼저 음성 신호를 일정한 시간 간격의 ‘단위 구간’으로 나누고, 이 단위 구간 하나만으로 또는 연속된 단위 구간을 이어 붙여 음소 추정 구간들을 만든다.

음성의 비교는 음소 단위로 이루어지는데 음소 추정 구간에 해당하는 음소를 알아내기 위해서 각 구간에서 ‘특징 벡터’를 추출한다. 각 음소 추정 구간에서 추출하는 특징 벡터는 1개이다. 특징 벡터는 음소를 구별하는 데 필요한 정보를 수치로 나타낸 것으로, 음소 추정 구간의 길이에 상관없이 1개로만 추출된다. 특징 벡터는 음소의 특성을 잘 나타내는 정보들을 이용하지만 사람마다 다른 특성을 보이는 정보는 사용하지 않는다. 사용하는 정보의 가짓수가 많을수록 음소를 더 정확하게 인식할 수 있지만 그만큼 필요한 연산량이 많아져 처리 시간은 길어진다.

음성을 인식하려면 ㉠ 입력 패턴의 특징 벡터와 기준 패턴의 특징 벡터를 비교해야 한다. 이를 위해서 음소 추정 구간이 비교하려는 기준 패턴의 음소 개수와 동일한 개수가 되도록 단위 구간을 조합한다. 그리고 각 음소 추정 구간에서 추출된 특징 벡터를 구간 순서대로 배열하여 입력 패턴을 생성한다.

예를 들어 ㉡ 입력된 음성 신호를 S_1, S_2, S_3 3개의 단위 구간으로 나눈 경우를 생각해 보자. 만일 비교하려는 기준 패턴의 음소가 3개라면 3개의 음소 추정 구간으로부터 입력 패턴이 구성되어야 하므로 $[S_1, S_2, S_3]$ 의 음소 추정 구간 배열을 설정하고, 이로부터 입력 패턴을 생성한다. 그런 다음 이것을 순서대로 기준 패턴의 음소와 일대일 대응시키고 각각의 특징 벡터의 차이를 구한 뒤 이것들을 모두 합하여 ‘패턴 거리’를 구한다. 만일 기준 패턴의 음소가 2개라면 3개의 단위 구간을 조합하여 $[S_1, S_2 \sim S_3], [S_1 \sim S_2, S_3]$ 로 2개의 음소 추정 구간 배열을 설정하고, 이로부터 입력 패턴을 생성한다. 이와 같이 1개의 기준 패턴에 대해 여러 개의 입력 패턴이 만들어질 수 있는 경우에는 ㉢ 생성 가능한 입력 패턴과 기준 패턴 사이의 패턴 거리를 모두 구하고, 그중의 최솟값을 그 기준 패턴에 대한 패턴 거리로 정한다. 만일 기준 패턴의 음소가 3개보다 크면 두 패턴을 일대일로 대응시킬 수 없으므로 비교가 불가능하다.

단위 구간의 시간 간격을 짧게 하여 그 개수를 늘리면 음소 추정 구간을 잘못 설정하여 발생하는 오류를 줄일 수 있다. 하지만 연산량이 많아져 처리 시간은 길어진다.

이와 같은 방법으로 컴퓨터에 저장된 모든 기준 패턴에 대해 패턴 거리를 구하고 그중 최솟값이 되는 기준 패턴을 선정한다. 최종적으로, 이 기준 패턴에 해당하는 문자열을 입력된 음성 신호에 대해 인식된 단어로 출력한다.

43. 위글의 내용과 일치하지 않는 것은?

- ① 음성 인식에서 말소리는 음소들의 시간적 배열로 본다.
- ② 입력 신호가 들어오면 잡음을 제거하고 음성 신호를 추출한다.
- ③ 개인의 독특한 목소리는 음성 인식을 위한 특징 벡터로 사용하기에 적당하다.
- ④ 입력 패턴은 음소 추정 구간의 특징 벡터들을 구간 순서로 배열한 것이다.
- ⑤ 패턴 거리가 최솟값인 기준 패턴에 해당하는 문자열을 인식된 단어로 출력한다.

44. 하나의 기준 패턴에 대해 ㉠을 ㉡에 적용할 때, 이에 대한 설명으로 옳지 않은 것은?

- ① 기준 패턴의 음소 개수가 3개이면 입력 패턴에 들어 있는 특징 벡터는 3개이다.
- ② 기준 패턴의 음소 개수가 3개이면 산출되는 패턴 거리는 1개이다.
- ③ 기준 패턴의 음소 개수가 2개이면 조합되는 음소 추정 구간 배열은 1개이다.
- ④ 기준 패턴의 음소 개수가 2개이면 생성 가능한 입력 패턴은 2개이다.
- ⑤ 기준 패턴의 음소 개수가 4개이면 패턴 비교가 불가능하다.

45. ㉢의 처리 시간을 증가시키는 요인으로 옳은 것은?

- ① 특징 벡터를 구성하는 정보의 가짓수의 감소
- ② 기준 패턴을 구성하는 음소 개수의 감소
- ③ 저장된 기준 패턴 가짓수의 감소
- ④ 단위 구간의 시간 간격의 감소
- ⑤ 음소 추정 구간 개수의 감소

단락별 정리

음성 인식 기술은 컴퓨터가 사람이 말하는 소리를 인식하여 해당 문자열로 바꾸는 기술이다. 사람의 말은 음소들의 시간적 배열로 볼 수 있다. 컴퓨터는 각 단어의 음소들의 배열을 '기준 패턴'으로 미리 저장해 두고, 이를 입력된 음성에서 추출한 '입력 패턴'과 비교하여 단어를 인식한다.

음성을 인식하기 위해서 먼저 입력된 신호에서 잡음을 제거한 후 음성 신호만 추출한다. 그런 다음 음성 신호를 하나의 음소로 판단되는 구간인 '음소 추정 구간'들의 배열로 바꾸어 준다. 그런데 음성 신호를 음소 단위로 정확히 나누는 것은 쉽지 않다. 이를 해결하기 위해 먼저 음성 신호를 일정한 시간 간격의 '단위 구간'으로 나누고, 이 단위 구간 하나만으로 또는 연속된 단위 구간을 이어 붙여 음소 추정 구간들을 만든다.

음성의 비교는 음소 단위로 이루어지는데 음소 추정 구간에 해당하는 음소를 알아내기 위해서 각 구간에서 '특징 벡터'를 추출한다. 각 음소 추정 구간에서 추출하는 특징 벡터는 1개이다. 특징 벡터는 음소를 구별하는 데 필요한 정보를 수치로 나타낸 것으로, 음소 추정 구간의 길이에 상관없이 1개로만 추출된다. 특징 벡터는 음소의 특성을 잘 나타내는 정보들을 이용하지만 사람마다 다른 특성을 보이는 정보는 사용하지 않는다. 사용하는 정보의 가짓수가 많을수록 음소를 더 정확하게 인식할 수 있지만 그만큼 필요한 연산량이 많아져 처리 시간은 길어진다.

음성을 인식하려면 ㉠ 입력 패턴의 특징 벡터와 기준 패턴의 특징 벡터를 비교해야 한다. 이를 위해서 음소 추정 구간이 비교하려는 기준 패턴의 음소 개수와 동일한 개수가 되도록 단위 구간을 조합한다. 그리고 각 음소 추정 구간에서 추출된 특징 벡터를 구간 순서대로 배열하여 입력 패턴을 생성한다.

예를 들어 ㉡ 입력된 음성 신호를 S₁, S₂, S₃ 3개의 단위 구간으로 나눈 경우를 생각해 보자. 만일 비교하려는 기준 패턴의 음소가 3개라면 3개의 음소 추정 구간으로부터 입력 패턴이 구성되어야 하므로 [S₁, S₂, S₃]의 음소 추정 구간 배열을 설정하고, 이로부터 입력 패턴을 생성한다. 그런 다음 이것을 순서대로 기준 패턴의 음소와 일대일 대응시키고 각각의 특징 벡터의 차이를 구한 뒤 이것들을 모두 합하여 '패턴 거리'를 구한다. 만일 기준 패턴의 음소가 2개라면 3개의 단위 구간을 조합하여 [S₁, S₂~S₃], [S₁~S₂, S₃]로 2개의 음소 추정 구간 배열을 설정하고, 이로부터 입력 패턴을 생성한다. 이와 같이 1개의 기준 패턴에 대해 여러 개의 입력 패턴이 만들어질 수 있는 경우에는 ㉢ 생성 가능한 입력 패턴과 기준 패턴 사이의 패턴 거리를 모두 구하고, 그중의 최솟값을 그 기준 패턴에 대한 패턴 거리로 정한다. 만일 기준 패턴의 음소가 3개보다 크면 두 패턴을 일대일로 대응시킬 수 없으므로 비교가 불가능하다.

단위 구간의 시간 간격을 짧게 하여 그 개수를 늘리면 음소 추정 구간을 잘못 설정하여 발생하는 오류를 줄일 수 있다. 하지만 연산량이 많아져 처리 시간은 길어진다.

이와 같은 방법으로 컴퓨터에 저장된 모든 기준 패턴에 대해 패턴 거리를 구하고 그중 최솟값이 되는 기준 패턴을 선정한다. 최종적으로, 이 기준 패턴에 해당하는 문자열을 입력된 음성 신호에 대해 인식된 단어로 출력한다.

전체 구조 잡기

물랐던 어휘 CHECK

[2013학년도 수능 43~45번]

[지문 분석]

1문단

① 음성 인식 기술은 컴퓨터가 사람이 말하는 소리를 인식하여 해당 문자열로 바꾸는 기술이다. ② 사람의 말은 음소들의 시간적 배열로 볼 수 있다. ③ 컴퓨터는 각 단어의 음소들의 배열을 '기준 패턴'으로 미리 저장해 두고, 이를 입력된 음성에서 추출한 '입력 패턴'과 비교하여 단어를 인식한다.

④ 음성 인식 기술은 / (컴퓨터가 / 사람이 말하는 소리를 인식하여 / 해당 문자열로 바꾸는) 기술이다.

-> 음성 인식 기술에 대한 정의가 첫 문장에서 제시가 되었네요 이렇게 처음에 어떤 소재에 대해서 정의한다면 당연히 그러한 정의된 소재에 대해서 구체화하면서 글이 전개될 가능성이 높습니다. 즉 우리는 '음성 인식 기술' -> 그래서 음성 인식 기술과 관련해 무엇을 말하고 싶은데?를 잡아줘야 한다는 겁니다. 또한 그간 기술 지문들을 본다면 이러한 [과학/기술]지문에서 어떤 기술이나 원리, 장치에 대해서 정의가 된다면 그러한 것들의 '구성 요소, 작동 원리' 등에 대해서 구체화할 가능성이 높다는 것도 배울 수 있습니다. 정리하면 아래와 같겠죠.

1) 음성 인식 기술 = [컴퓨터 -> 사람의 말하는 소리 인식 -> 해당 문자열로 바꾸는 기술]

-> 따라서 아마도 이 글은 음성 인식 기술에 대해서 구체화할 가능성이 높고, 더 구체적으로는 음성 인식 기술에서 '어떻게 컴퓨터가 사람의 말하는 소리를 인식하여, 해당 문자열로 바꾸는 것인가'에 대해서 설명할 수도 있겠다는 생각을 할 수 있습니다.

매우 뻘하고, 당연한 말이지만 이러한 뻘하고 당연한 사고 과정이 의식적인 행동 교정을 통해 습관화되어야 합니다. 그러한 당연한 과정이 안 되니까 글이 어려운 것이겠죠.

또한 저는 특정한 개념 정의되면 반드시 워딩을 찍어주고 넘어갑니다. 정보량을 줄이기 위한 저의 습관이지요. 워딩을 찍는다는 것이 무엇이나 '음성 인식 기술'에 대해 서술된 '정의'를 읽고 난 뒤, 다시 한 번 '음성 인식 기술'이라는 워딩을 보고 왜 그렇게 명명되었는지를 '이해'하고 '납득'할 수 있다면 납득하고 넘어간다는 겁니다. 즉 ①의 문장을 왼쪽에서 오른쪽으로 쪽 읽고 넘어가는 것이 아니라, 읽은 뒤 다시 '아 그래서 음성 / 인식 / 기술 이구나' 하고 납득을 하고 넘어간다는 거죠. 이 작은 습관 하나가 정보량이 많은 지문에서 상당히 큰 도움이 됩니다. 즉 말씀드리고 싶은 것은 '단순히 정보량이 많아'가 아니라, '왜 많은가? -> 낯선 개념이 정의되면 머릿속에 남지 않아서 -> 그렇다면 정의를 이해했다면 의식적으로 남길 수 있는 것은 남기고 넘어가자'와 같은 행동 요령이 만들어져야 한다는 겁니다.

② 사람의 말은 / (음소들의 시간적 배열)로 / 볼 수 있다.

-> 첫 문장에서 음성 인식 기술을 정의하고 두 번째 문장에서 [사람의 말]에 대해서 이야기 하고 있네요. 여기서부터 문장이 불편합니다. 다만 학생들은 그러한 불편한 이유를 고민하지 않습니다. '그냥 어렵네' 정도로 생각을 합니다. 여러분이 문장을 읽어가면서 '불편한 느낌'이 드는 경우 중 대다수의 경우는 '주어가 변환될 때입니다. 즉 이는 해당 문장에서 서술하는 서술 대상이 바뀐다는 말이고, 그렇기에 앞문장과 흐름이 끊기는 경우가 많습니다. 그래서 문장이 불편한 겁니다. 그렇다면 이럴 땐 어떻게 해야 할까요? 그렇죠. 의식적으로 바뀐 서술 대상을 파악하고, 그 대상이 앞문장과 어떤 관련이 있는지를 생각하고 연결해나가야 합니다. 즉 해당 문장에서의 '사람의 말'은 단순히 '인간의 언어'라는 의미가 아니라, 앞문장에서 말한 '컴퓨터가 인식하는 사람의 말하는 소리'와 문맥적으로 같은 의미가 되겠죠.

1) 사람의 말 = 음소들의 시간적 배열

-> 또한 해당 지문이 어려웠다면 ②부터 문제가 있지 않은지 반드시 살펴 보아야 합니다. 그 이유는 '어휘' 때문입니다. '음소'라는 어휘를 읽고 '이미지화'하지 못하고, 그냥 '음소'라는 텍스트 자체로 받아들이고 넘어가게 되면, 이를 바탕으로 후술되는 내용 및 과정들이 모두 추상적으로 다가올 가능성이 높고, 이해가 안 될 수밖에 없습니다.

무슨 말이나 하면 여기서 [음소]라는 어휘에 대한 명확한 이미지가 있어야 합니다. 우리가 고교 과정 내에서 배우는 음소는 국어에서 생각해보면 생각하면 '자음과 모음'입니다. 즉 이와 같이 추상적인 어휘나 개념이 등장하게 된다면 최대한 '구체적'인 사례로 생각을 해주어야 하며, 그래야 이미지가 형성되고, 글이 선명하게 다가옵니다. 즉 정리하면

[사람의 말] = [자음과 모음의 시간적 배열]

과 같이 정보가 처리되었어야 합니다. 이 사소한 [음소 = 자음과 모음]이라는 단어의 이미지를 떠올리지 못했다면 뒤의 내용을 읽으면서 어려울 수밖에 없어요. 늘 말하지만 글자만 읽는 것이 아니라 읽고 생각해주셔야 합니다.

-> 그렇다면 왜 갑자기 두 번째 문장에서 [사람의 말]에 대해서 이야기하고 있는 걸까요? 아마도 [음성 인식 기술]에 대해서 설명하기 위해서 필요한 선행 개념이기 때문에 먼저 설명해준 거죠. 즉 앞 문장에서 [음성 인식 기술]에 대해서 정의한 내용에 따르면 [음성 인식 기술 -> 컴퓨터가 사람의 말 인식 -> 해당 문자열로 바꾸는 기술]이라고 했고, 두 번째 문장에서 컴퓨터가 인식하는 [사람의 말소리]에 대해서 추가적인 설명을 제시한 거죠. 두 문장을 잘 붙여서 읽었다면 '[아 컴퓨터가 인식하는 사람의 말에 대해서 추가적인 설명을 제시하는 거구나]'라고 생각할 수 있어야 합니다. 즉 음성 인식 기술을 [컴퓨터가 -> 사람의 말소리인 자음과 모음의 시간적 배열을 인식해서 -> 그에 해당하는 문자열, 즉 해당하는 자음과 모음으로 바꾸는 기술]정도로 정리하고 넘어갔다면 훨씬 수월하겠죠.

③ 컴퓨터는 / (각 단어의 음소들의 배열)을 / ('기준 패턴')으로 미리 저장해 두고, // 이를 / (입력된 음성에서 추출한 / '입력 패턴')과 비교하여 / 단어를 인식한다.

-> 빠르게 읽으면 뭉개고 넘어갔을 만한 문장이네요. 정확하게 쪼개서 읽어 줘야 합니다. 주어가 '컴퓨터'입니다. 여기서의 '컴퓨터' 더 이상 여러분들이 못할 때 사용하는 컴퓨터가 아니라, 문맥적으로 ①에서 제시된 '사람들의 말소리를 인식하여 해당 문자열로 바꾸는' 컴퓨터가 되겠죠. 별서 ②에서 익힌 행동 요령을 ③에서 사용할 수 있었고, 그렇다면 훨씬 수월하게 문장이 연결되었을 겁니다. 내용을 정리해보면 아래와 같습니다.

1) 컴퓨터는 [각 단어들의 음소들의 배열 = 각 단어들의 자음과 모음의 배열]을 -> [기준 패턴]으로 미리 저장해 해둠

-> 정리하면 '각 단어들의 자음과 모음들의 배열'이 '기준 패턴'으로 저장된다는 건데. 워딩을 찍어보면 뭔가 '기준이 되는 패턴' 정도가 되겠네요.

2) 그리고 컴퓨터는 [입력된 음성 = ①에서의 '사람이 말하는 소리']에서 [입력 패턴]을 추출함

-> 여기서도 사소한 생각하나가 불편함을 줄여줍니다. 단순히 '입력 패턴'으로 정보를 남기려 하면 당연히 남지 않습니다. 우리는 '입력 패턴'이 무엇인지에 대한 정보가 부족하니까요. 당연한 겁니다. 그렇다면 앞에 제시된 정보를 활용해 해당 정보를 이해해보려는 노력을 해야 합니다. 이 부분에서 우리가 이용할 수 있는 정보는 1)에 제시된 '기준 패턴'에 대한 정보가 되겠죠. '기준 패턴'은 '기준이 되는 패턴' 정도였으며, '음성을 인식하는 컴퓨터가 미리 저장해놓는 각 단어들의 자음과 모음의 배열'이라고 했습니다. 즉 미리 단어들의 자음과 모음들의 배열을 저장해놓고 입력되는 말소리에 대한 기준으로 쓰는 패턴이라는 말이 되겠죠.

그렇다면 이를 바탕으로 '입력 패턴'을 생각해보면 입력 패턴은 조금 더 구체적으로 무엇이 될까요? 물론 정확하게는 모르겠지만 '입력된 음성에서 추출해낸 단어의 자음과 모음의 배열이 되지 않을까?'라는 생각을 해볼 수 있겠죠. 이런 작은 생각하나가 해당 문장에 대한 이해를 훨씬 수월하게 해줍니다.

3) 그리고 컴퓨터는 1)에서와 같이 미리 저장되어 있던 [기준 패턴]과 2)에서 추출한 [입력 패턴]을 비교해서 -> 단어를 인식함.

-> 2)에서 위와 같은 생각을 해주었다면 3) 역시 훨씬 수월하게 이해가 됩니다. 즉 음성 인식 기술에서는 컴퓨터가 '각 단어의 자음과 모음의 배열을 '기준 패턴'으로 미리 저장해 둔 뒤, 입력된 음성에서 추출된 '자음과 모음의 배열'인 '입력 패턴'과 '비교'해서 단어를 '인식'한다는 말이 되겠죠.

-> 1문단은 한 문장, 한 문장 위와 같이 천천히 쪼개 읽고, 최대한 뽑아낼 수 있는 정보를 뽑아내야 합니다. 특히 과학 기술에서는 초반부에 후술되는 내용을 위해 필요한 개념이나, 구성 요소들에 대한 개념을 세팅하고 이를 엮어서 과정이나 원리로 구체화해나가는 과정이 많습니다. 그렇기에 초반부에 제시되는 부분을 몽개버리면 당연히 후술되는 내용들을 이해하는 건 불가능합니다. 따라서 이러한 유형의 지문들에서는 의식적으로 속도를 늦추고, 최대한 깊게 생각하고, 워딩에 대한 친숙도를 높이고 넘어가는 게 중요합니다.

결국 여기까지 읽고 정리하면 첫 문장에서 [음성 인식 기술]에 대해서 정의하고, 두 번째 문장에서 선행 개념을 설명하고 세 번째 문장에서 ①의 내용을 조금 더 구체화해서 제시하고 있는 흐름이었죠.

-> 여기까지 읽고 결국 1문단의 핵심 소재는 [음성 인식 기술]이었고, 구체화된 화제는 그러한 [음성 인식 기술의 과정]에 대한 이야기가 되겠죠. 또한 1문단을 잘 읽었다면 여러분 머릿속에는 아래와 같은 핵심 정보들이 정리되었어야 해요.

- 1) 사람의 말소리 = 음소들의 시간적 배열 = 자음과 모음의 시간적 배열
- 2) 음성 인식 기술은 -> 사람이 말하는 소리를 인식 -> 해당 문자열로 바꾸는 기술
- 3) [사람이 말하는 소리를 인식]하기 위해 컴퓨터는 [기준 패턴]과 [입력 패턴]을 비교함.

-> 따라서 2문단부터는 이러한 과정에 대해서 조금 더 구체적으로 서술할 가능성이 높겠죠. 완벽하게 '각 패턴에 대한 구체화 / 각 패턴을 비교하는 과정에 대해 설명하겠네'와 같은 생각을 하는 것은 불가능하더라도, 적어도 [음성 인식 기술의 과정]과 관련된 글이 될 것이며, 그러한 것들에 대해서 '설명할 수도 있겠다' 정도의 예측은 할 수 있는 겁니다. 그리고 이러한 예측을 하면서 읽게 되면 글을 읽는 게 훨씬 수월해집니다. 내 예측이 맞다면 글이 잘 읽어가고 있다는 지표가 될 것이고, 그렇지 않다면 그러한 다른 흐름에 민감하게 '반응'하게 될 거니까요. 아무 생각없이 그냥 '글자'만 읽는 게 제일 안 좋습니다.

1문단 정리

[음성 인식 기술 = 컴퓨터 -> 사람의 말소리를 인식 -> 문자열로 바꿈]

2문단

① 음성을 인식하기 위해서 먼저 입력된 신호에서 잡음을 제거한 후 음성 신호만 추출한다. ② 그런 다음 음성 신호를 하나의 음소로 판단되는 구간인 '음소 추정 구간'들의 배열로 바꾸어 준다. ③ 그런데 음성 신호를 음소 단위로 정확히 나누는 것은 쉽지 않다. ④ 이를 해결하기 위해 먼저 음성 신호를 일정한 시간 간격의 '단위 구간'으로 나누고, 이 단위 구간 하나만으로 또는 연속된 단위 구간을 이어 붙여 음소 추정 구간들을 만든다.

-> 1문단에서 [음성 인식 기술 -> 입력 패턴과 기준 패턴을 비교하여 -> 단어를 인식 -> 해당 문자열로 바꿈]에 대해서 전반적으로 설명했다면 2문단부터는 [단어를 인식]하기 위한 과정에 대해서 더욱 구체화하고 있죠. 2문단부터 학생들은 어려움을 많이 겪어요. 현장에서 지켜보면 대부분의 학생들이 문장을 몽개고 넘어가서 그렇습니다. [음소 추정 구간 / 음소 추정 구간들의 배열]과 같이 명시적으로 다른 정보를 몽개고 같은 정보로 받아들이기 때문이죠. 한 문장씩 천천히 잘 쪼개 읽어봅시다.

① 음성을 인식하기 위해서 / 먼저 입력된 신호에서 / 잡음을 제거한 후 / 음성 신호만 추출한다.

-> 즉 [입력된 신호 = 음성 + 잡음]인데 거기서 [잡음]을 제거한 후 [음성 신호]만 추출한다는 이야기네요.

② 그런 다음 / [음성 신호]를 / [하나의 음소로 판단되는 구간인 = 음소 추정 구간]들의 배열로 바꾸어 준다.

-> 해당 문장을 정확하게 쪼개지 못해요. 그래서 어려움을 겪는 경우가 많습니다. 정확하게 쪼개 읽었다면 아래와 같은 정보들이 받아들여져야 합니다.

- 1) 하나의 음소(자음or모음)로 판단되는 구간 = 음소 추정 구간
- 2) ①에서 추출한 음성 신호를 [음소 추정 구간들의 '배열']로 바꾸어 줌

-> 즉 정리하면 ①에서 음성 신호만 추출한 뒤 그러한 음성 신호를 [음소 추정 구간들의 배열]로 바꾸어준다는 말이죠. 여기서 중요한 것은 [음소 추정 구간]이 아니라 [음소 추정 구간들의 배열]로 바꾼다는 거죠. 왜 그럴까요? 그렇죠. 추출된 음성 신호는 하나의 음소, 즉 하나의 자음과 모음이 아니라 [그러한 자음과 모음의 시간적 배열인 = 단어]로 볼 수 있겠죠. 따라서 그러한 단어를 인식하기 위해서는 추출된 음성 신호를 [음소로 추정되는 구간 -> 그러한 구간들의 배열]로 바꿔줘야 되는 거죠.

-> ①~②까지 읽고 [단어를 인식하기 위해] -> [입력된 신호 -> 잡음 제거 후 -> 음성 신호 -> 음소 추정 구간들의 배열로 바꿈]의 과정이 머릿속에 남았으면 됩니다.

③ 그런데 [(음성 신호)를 (음소 단위)로 정확히 나누는 것은 / 쉽지 않다. ④ 이를 해결하기 위해 / 먼저 (음성 신호)를 (일정한 시간 간격의 = 단위 구간)으로 나누고, / (이 단위 구간 하나만으로) 또는 (연속된 단위 구간을 이어 붙여) / 음소 추정 구간들을 만든다.

- 1) 음성 신호를 -> 음소 단위로 정확히 나누는 것은 쉽지 않음
- 2) 1)과 같은 문제를 해결하기 위해 먼저 [음성 신호]를 [일정한 시간 간격의 단위 구간]으로 나눔
- 3) 그리고 그러한 [단위 구간 하나] 또는 [연속된 단위 구간을 이어 붙여] -> [음소 추정 구간들]을 만들

-> ③을 앞 문장과 잘 붙여 읽어야 합니다. ①~②에서 말한 것과 같이 [음성 신호 -> 음소 추정 구간들의 '배열']로 바꾸어야 하는데 그 과정에서 [음성 신호]를 [음소 단위] 즉 [자음과 모음]단위로 정확히 나누는 것이 쉽지 않다는 말이죠. 그래서 ④에서는 그러한 문제를 해결하기 위한 방법이 제시가 되네요. 즉 음성 신호를 [자음과 모음] 단위로 정확하게 나누는 것이 쉽지 않으므로 일단 먼저 [일정한 시간 간격 = 단위 구간]으로 나눠버린 다음

그것을 조합해 [음소 추정 '구간']들을 만든다는 거죠.

-> 여기까지 잘 읽었다면 하나의 의문이 들어야 합니다. 뭘까요? ④에서 만든 [음소 추정 구간]들을 [배열]하는 것에 관련한 설명이죠. 즉 [음소 추정 구간들의 '배열']이 어떻게 이루어지는가에 대한 설명이 빠졌죠. 따라서 아마도 다음 문단 중에서 그것에 대해서 설명할 가능성이 있겠다는 생각을 해줬다면 훌륭합니다. 이걸 파악할 수 있는 이유가 뭘까요? 분명 지문에서는 [음소 추정 구간 <-> 음소 추정 구간들의 배열]을 명시적으로 구분해서 적어놨어요. 다만 여러분들이 글을 읽을 때 정확하게 쪼개 읽지 않고 문개고 넘어가는 게 문제인 거죠. 정확하게 쪼개 읽도록 합시다. :)

3문단

①음성의 비교는 음소 단위로 이루어지는데 음소 추정 구간에 해당하는 음소를 알아내기 위해서 각 구간에서 '특징 벡터'를 추출한다. ②각 음소 추정 구간에서 추출하는 특징 벡터는 1개이다. ③특징 벡터는 음소를 구별하는 데 필요한 정보를 수치로 나타낸 것으로, 음소 추정 구간의 길이에 상관없이 1개로만 추출된다. ④특징 벡터는 음소의 특성을 잘 나타내는 정보들을 이용하지만 사람마다 다른 특성을 보이는 정보는 사용하지 않는다. ⑤사용하는 정보의 가짓수가 많을수록 음소를 더 정확하게 인식할 수 있지만 그만큼 필요한 연산량이 많아져 처리 시간은 길어진다.

④ 음성의 비교는 / 음소 단위로 이루어지는데 / (음소 추정 구간에 해당하는 음소)를 알아내기 위해서 / 각 구간에서 / '특징 벡터'를 추출한다.

-> 2문단까지 해서 [입력된 신호 -> 음성 신호 -> 음소 추정 구간들 만들]에 대해서 설명했고 4문단에서는 [음성의 비교]를 위해 필요한 과정에 대해서 설명하죠. 다만 그냥 [음성의 비교구]가 아니라, 이러한 음성의 비교는 1문단의 제시되었던 정보에 따르면 아마도 [입력 패턴과 기준 패턴]의 비교를 말하고 있다는 것을 파악했어야 합니다.

- 1) 음성의 비교는 음소 단위로 이루어짐
2) 2문단을 통해 만들어진 음소 추정 구간에 해당하는 음소를 알아내기 위해 특징 벡터를 추출함

-> 그렇다면 2)에서 음소 추정 구간에 해당하는 음소를 왜 알아내야 하는 거죠? 앞에 제시된 정보들을 통해 파악해야죠. 그렇죠. 1문단에 따르면 [입력된 말소리에서 추출된 입력 패턴]과 [기준 패턴]을 비교해서 단어를 인식한다고 했었어요. 그런데 3문단에 따르면 그러한 비교를 할 때 [음소 단위]로 이루어진다는 거죠. 즉 [자음과 모음]의 단위로 입력 패턴과 기준 패턴을 비교해서 단어를 인식한다는 말이죠. 그렇기 때문에 2문단의 과정을 통해 입력된 말소리에서 만들어진 [음소 추정 구간들]이 각각 어떤 음소에 해당하는지를 알아내야 하는 것이고, 그러기 위해서 [특징 벡터]를 추출한다는 말이에요. 어려운가요? 다시 한 번 정리하면

[입력된 말소리 -> 음성 신호 추출 -> 단위 구간으로 나눠서 조합 -> 음소 추정 구간들 만들 -> 그런데 음성의 비교는 음소 단위로 이루어짐 즉 입력 패턴과 기준 패턴을 비교할 때 각각의 자음과 모음 단위로 비교함 -> 따라서 앞서 만들어진 음소 추정 구간들이 각각 어떤 음소에 해당하는지를 알아야 음소 단위로 입력 패턴과 기준 패턴을 비교할 수 있음 -> 그래서 음소 추정 구간들에서 특징 벡터를 추출함]

과 같은 과정으로 정리가 되겠죠. :)

② (각 / 음소 추정 구간에서 추출하는 / 특징 벡터)는 1개이다. ③ 특징 벡터는 / [(음소를 구별하는 데 / 필요한 정보)를 (수치)로 나타낸 것]으로, / 음소 추정 구간의 길이에 상관없이 / 1개로만 추출된다.

-> ②~⑤는 ①에서 말한 [특징 벡터]를 구체화한 내용들에 해당하죠. 먼저 ②~③을 읽어보면 아래와 같은 정보가 정리가 되겠네요.

- 1) 특징 벡터 = [음소를 구별하는데 필요한 정보]를 [수치]로 나타낸 것
2) 음소 추정 구간 1개당 길이에 상관없이 1개의 특징 벡터만 추출 됨

-> 사실은 당연한 정보죠. 음소 추정 구간은 1개의 음소로 추정되는 구간이므로 [그러한 음소를 구별하기 위해 필요한 정보]를 [수치]로 나타낸 특징 벡터 역시 구간의 길이에 상관없이 1개로만 추출이 될 테니까요.

④ 특징 벡터는 / 음소의 특성을 잘 나타내는 정보들을 이용하지만 / 사람마다 다른 특성을 보이는 정보는 사용하지 않는다. ⑤ 사용하는 정보의 가짓수가 많을수록 / 음소를 더 정확하게 인식할 수 있지만 / 그만큼 필요한 연산량이 많아져 / 처리 시간은 길어진다.

-> ④를 보면 그러한 특징 벡터는 당연히 음소의 특징을 잘 나타내는 정보를 이용해야겠죠. 그래야 음소가 구별이 될 테니까요. 그리고 하나의 음소에 대해서 사람마다 다른 특성을 보이는 정보는 음소를 구별해줄 수 있는 정보가 아니므로 사용하지 않겠네요. 늘 말하지만 그냥 글자만 읽는 것이 아니라 [읽고 -> 생각하고 -> 납득]하면서 읽어줘야 기억에 남습니다. 천천히 쪼개 읽고 생각하고 납득해주세요.

-> ⑤를 보면 [사용하는 정보]는 [음소의 특성을 잘 나타내는 정보]가 되겠죠. 따라서 그러한 정보의 가짓수가 많으면 당연히 음소를 더 정확하게 인식할 수 있겠죠. 다만 정보의 가짓수가 많아질수록 처리하는데 필요한 연산량이 많아질 것이고 처리 시간은 길어지게 되겠네요. 화살표를 위, 아래 표시하는 게 중요한 게 아니라 납득할 수 있는 내용들은 납득하고 넘어가주세요 하니까.

-> 또한 여기서 중요한 것은 [사용하는 정보의 가짓수]가 많아진다고 해서 [특징 벡터]가 여러 개가 되는 게 아니라는 겁니다. 착각하시면 안 돼요. 특징 벡터는 음소 추정 구간 1개당 1개만 나오게 되는데, 이러한 특징 벡터는 [음소를 구별하는데 필요한 정보를 -> 수치로 나타낸 것]이죠. 따라서 음소를 구별하는데 필요한 정보가 많아진다면 그러한 정보를 수치로 나타내기 위해 필요한 [연산량]과 [처리시간]이 길어진다는 것이지 결국 추출되는 특징 벡터는 1개의 수치가 되겠죠. :)

->아마도 다음 문단은 이러한 특징 벡터를 바탕으로 [음성의 비교]로 구체화할 가능성이 높고, 앞서 남았던 의문인 [음소 추정 구간들 -> 배열]에 대해서도 설명이 제시될 수 있겠죠.

2~3문단 정리

[단어를 인식하기 위해 -> 입력된 말소리 -> 음성 신호 -> 단위 구간으로 나누고 조합 -> 음소 추정 구간들 -> 특징 벡터 추출]

4문단

①음성을 인식하려면 입력 패턴의 특징 벡터와 기준 패턴의 특징 벡터를 비교해야 한다. ②이를 위해서 음소 추정 구간이 비교 하려는 기준 패턴의 음소 개수와 동일한 개수가 되도록 단위 구간을 조합한다. ③그리고 각 음소 추정 구간에서 추출된 특징 벡터를 구간 순서대로 배열하여 입력 패턴을 생성한다.

① 음성을 인식하려면 / (입력 패턴의 특징 벡터)와 (기준 패턴의 특징 벡터)를 비교해야 한다.

-> 3문단까지 읽으면서 앞과 같은 의문과 생각을 하면서 읽어준다면 4문단을 읽으면서 뭔가 퍼즐이 맞춰지는 것과 같은 느낌이 들어야 합니다. 그렇지 않고 ['...?']라는 생각이 든다면 앞서 제시된 정보들을 활용하지 못하면서

그냥 글자만 읽었을 가능성이 큼니다.

-> 1문단과 3문단에서 언급된 것과 같이 [컴퓨터 -> 음성을 인식]하기 위해서는 [입력 패턴]과 [기준 패턴]을 비교해야하는데 4문단은 그것에 대해서 구체화하고 있는 문단이죠. 즉 그냥 [입력 패턴]과 [기준 패턴]을 비교하는 게 아니라 각각의 패턴에서 [추출된 특징 벡터]를 비교하는 거죠. 이를 말하기 위해서 3문단까지 선행 과정에 대한 설명이 제시된 거네요.

② 이를 위해서 / [(음소 추정 구간)이 (비교하려는 기준 패턴의 음소 개수)와 동일한 개수가 되도록] / 단위 구간을 조합한다. ③ 그리고 각 음소 추정 구간에서 추출된 특징 벡터를 / 구간 순서대로 배열하여 / 입력 패턴을 생성한다.

-> 왜 [이 = 두 패턴의 특징 벡터를 비교]하기 위해서 [음소 추정 구간의 개수]가 [기준 패턴의 음소의 개수]와 동일하도록 단위 구간을 조합해야 할까요? 그렇죠. 음성의 비교는 [음소의 단위]로 이루어지기 때문이죠. 무슨 말이나면 [음소 추정 구간] 1개당 1개의 특징 벡터가 추출된다고 했으므로 [음소 추정 구간의 개수 = 기준 패턴의 음소의 개수]가 동일해야 [음소 추정 구간에서 추출된 특징 벡터의 개수 = 기준 패턴의 음소들에서 추출된 특징 벡터의 개수]가 되겠죠. 그리고 이렇게 만들어진 특징 벡터들을 구간 순서대로 [배열]해서 입력 패턴을 만들어야 [입력 패턴]과 [기준 패턴]의 특징 벡터들을 각각 음소 단위로 비교할 수 있겠죠.

4문단 정리

[단어를 인식하기 위해 -> 입력된 말소리 -> 음성 신호 -> 단위 구간으로 나누고 조합 -> 음소 추정 구간들 -> 특징 벡터 추출 -> 구간 순서로 배열해서 입력 패턴을 만들 -> 이렇게 만들어진 입력 패턴의 특징 벡터와 기준 패턴의 특징 벡터를 비교함]

5문단

① 예를 들어 입력된 음성 신호를 S1, S2, S3 3개의 단위 구간으로 나누는 경우를 생각해 보자. ② 만일 비교하려는 기준 패턴의 음소가 3개라면 3개의 음소 추정 구간으로부터 입력 패턴이 구성되어야 하므로 [S1, S2, S3]의 음소 추정 구간 배열을 설정하고, 이로부터 입력 패턴을 생성한다. ③ 그런 다음 이것을 순서대로 기준 패턴의 음소와 일대일 대응시키고 각각의 특징 벡터의 차이를 구한 뒤 이것들을 모두 합하여 '패턴 거리'를 구한다. ④ 만일 기준 패턴의 음소가 2개라면 3개의 단위 구간을 조합하여 [S 1, S 2 ~ S 3], [S 1 ~ S 2, S 3]로 2개의 음소 추정 구간 배열을 설정하고, 이로부터 입력 패턴을 생성한다. ⑤ 이와 같이 1개의 기준 패턴에 대해 여러 개의 입력 패턴이 만들어질 수 있는 경우에는 생성 가능한 입력 패턴과 기준 패턴 사이의 패턴 거리를 모두 구하고, 그중의 최솟값을 그 기준 패턴에 대한 패턴 거리로 정한다. ⑥ 만일 기준 패턴의 음소가 3개보다 크면 두 패턴을 일대일로 대응시킬 수 없으므로 비교가 불가능하다.

-> 5문단은 4문단까지 설명한 과정을 바탕으로 [음성의 비교]에 대해서 [예시]를 들어 구체적으로 설명하고 있는 문단이었죠. 따라서 앞 문단의 내용들을 정확하게 쫓아서 파악해줬다면 수월하게 이해가 됐을 거고 그렇지 않다면 이해하기 어려웠을 것 같아요. 5문단은 크게 [①~③], [④~⑤], [⑥]과 같은 3가지 경우를 가정해서 설명하고 있었으므로 각 경우의 차이를 파악하면서 읽어줬으면 됩니다.

① 예를 들어 / [(입력된 음성 신호)를 (S1, S2, S3 3개의 단위 구간)으로 나누는 경우]를 생각해 보자. ② 만일 (비교하려는 기준 패턴의 음소)가 3개라면 / 3개의 음소 추정 구간으로부터 입력 패턴이 구성되어야 하므로 / [S1, S2, S3]의 음소 추정 구간 배열을 설정하고, / 이로부터 입력 패턴을 생성한다. ③ 그런 다음 (이것)을 순서대로 (기준 패턴의 음소)와 일대일 대응시키고 / 각각의 특징 벡터의 차이를 구한 뒤 /

이것들을 모두 합하여 / '패턴 거리'를 구한다

-> ①~③은 [기준 패턴의 음소가 3개]인 경우가 되겠네요. 잘 읽었다면 아래와 같이 정보가 정리되면 됩니다.

- 1) 음성 신호를 S1, S2, S3의 3개의 단위 구간으로 나눔
- 2) 기준 패턴의 음소의 개수는 3개임
- 3) 따라서 1)에서의 단위 구간을 조합해 음소 추정 구간을 만들 때 -> 3개의 음소 추정 구간으로 만들어야 함
- 4) 그러므로 단위 구간 각 1개씩으로 S1, S2, S3와 같은 각각의 음소 추정 구간을 만들 -> 이를 구간 순서로 배열해서 -> [S1, S2, S3]와 같은 [음소 추정 구간 '배열']을 설정해 [입력 패턴]을 생성함.
- 5) 그런 다음 이러한 입력 패턴의 3개의 추정 구간과 기준 패턴의 각 3개의 음소를 일대일 대응시킴 -> 입력 패턴과 기준 패턴에서 각 3개씩의 특징 벡터가 서로 대응 될 것임 -> 그러한 특징 벡터들의 차이를 구한 뒤 -> 차이를 모두 합하여 패턴 거리를 구함

-> 아마도 이러한 패턴 거리가 작을수록 즉 특징 벡터들의 차이가 작을수록 기준 패턴의 해당하는 음소일 가능성이 높은 거겠죠. :)

④ 만일 기준 패턴의 음소가 2개라면 / 3개의 단위 구간을 조합하여 / [S1, S2 ~ S3], [S1 ~ S2, S3]로 2개의 음소 추정 구간 배열을 설정하고, / 이로부터 입력 패턴을 생성한다. ⑤ 이와 같이 / [1개의 기준 패턴에 대해 / 여러 개의 입력 패턴이 만들어질 수 있는 경우에는 / 생성 가능한 [(입력 패턴)과 (기준 패턴) 사이의 패턴 거리]를 모두 구하고, / (그 중의 최솟값)을 (그 기준 패턴에 대한 패턴 거리)로 정한다.

-> ①~③과 같은 과정으로 이해해주시면 됩니다. 이번엔 기준 패턴의 음소가 2개라면 3개의 단위 구간을 조합해서 음소 추정 구간을 만들 때 2개의 음소 추정 구간이 나오도록 배열을 해야겠죠. 그래야 일대일 대응을 시켜 비교할 수 있으니까요. 따라서 위의 내용과 같이 총 2개의 입력 패턴이 나오게 되겠고 2번의 비교를 통해 2개의 패턴 거리가 나오게 되겠죠. 그리고 2개의 패턴 거리 중 최솟값을 해당 기준 패턴에 대한 패턴 거리로 정한다고 하네요.

-> 결국 두 번째 가정에서와 같이 하나의 기준 패턴에 대해 여러 입력 패턴이 만들어지는 경우에는 모든 입력 패턴을 구하고, 각각을 모두 기준 패턴과 비교한 후 그 중 [최솟값]을 [그 기준 패턴에 대한 패턴 거리]로 정한다는 말이죠. 그렇다면 왜 [최솟값]일까요? 그렇죠. [패턴 거리]는 [입력 패턴의 특징 벡터]와 [기준 패턴의 특징 벡터]를 일대일 대응시킨 후 각각의 차이를 모두 합한 것인데, 그러한 차이가 작을수록 기준 패턴의 해당 음소에 더 가깝다고 볼 수 있기 때문이죠. :)

⑥ 만일 기준 패턴의 음소가 3개보다 크면 두 패턴을 일대일로 대응시킬 수 없으므로 비교가 불가능하다.

-> 마지막 사례에서는 [기준 패턴의 음소 개수 > 단위 구간의 개수]일 때는 단위 구간을 1개씩으로 음소 추정 구간을 만들어 배열해도 두 패턴이 일대일 대응이 되지 않으므로 비교가 불가능함을 제시하고 있는 거죠.

-> 따라서 여러분들이 5문단을 읽고 나서 들어가야 하는 생각은 아래와 같습니다.

5문단 정리

- 1) [입력 패턴 vs 기준 패턴]의 비교에 대해 예시를 통해 구체화
- 2) 총 3가지 사례에 대해 설명

① 단위 구간의 개수 = 기준 패턴의 음소의 개수
수능 국어의 새로운 기준 김민수

- > 1번의 비교를 통해 1개의 패턴 거리 생성
- ② 단위 구간의 개수 > 기준 패턴의 음소의 개수
 - > 만들어질 수 있는 모든 입력 패턴을 구하고, 각각을 모두 기준 패턴과 비교해서 패턴 거리를 구한 후, 그 중 최솟값을 선정
- ③ 단위 구간의 개수 < 기준 패턴의 음소의 개수
 - > 일대일 대응이 불가능 -> 비교 불가능

6문단

①단위 구간의 시간 간격을 짧게 하여 그 개수를 늘리면 음소 추정 구간을 잘못 설정하여 발생하는 오류를 줄일 수 있다. ②하지만 연산량이 많아져 처리 시간은 길어진다.

④ 단위 구간의 시간 간격을 짧게 하여 / 그 개수를 늘리면 / (음소 추정 구간을 잘못 설정하여 발생하는 오류)를 줄일 수 있다. ② 하지만 연산량이 많아져 / 처리 시간은 길어진다.

-> 두 문장밖에 안 돼서 문단이라고 하기도 민망하네요. 다만 이해하는 것은 쉽지 않았을 것 같아요.

- 1) 단위 구간의 시간 간격을 짧게 설정 -> 그렇다면 당연히 단위 구간의 개수 자체는 증가함
- 2) 그러한 단위 구간의 조합을 통해 음소 추정 구간을 만들
- 3) 따라서 단위 구간의 개수가 많아지면 조합할 수 있는 음소 추정 구간들의 종류가 증가하게 되고 그렇게 되면 만들어질 수 있는 입력 패턴도 증가하게 될 것이고, 그에 따라 만들어질 수 있는 패턴 거리 역시 증가
- 4) 그래서 단위 구간을 잘못 설정하여 발생할 수 있는 오류를 줄일 수 있음
- 5) 다만 위와 같은 과정에서 연산량은 늘어나기 때문에 처리 시간은 증가함

6문단 정리

[단위 구간의 시간 간격을 짧게 하면 -> 오류는 줄지만 처리 시간은 증가]

7문단

①이와 같은 방법으로 컴퓨터에 저장된 모든 기준 패턴에 대해 패턴 거리를 구하고 그중 최솟값이 되는 기준 패턴을 선정한다. ②최종적으로, 이 기준 패턴에 해당하는 문자열을 입력된 음성 신호에 대해 인식된 단어로 출력한다.

④ 이와 같은 방법으로 / (컴퓨터에 저장된 모든 기준 패턴)에 대해 / 패턴 거리를 구하고 / (그중 최솟값이 되는 기준 패턴)을 선정한다. ② 최종적으로, (이 기준 패턴에 해당하는 문자열)을 (입력된 음성 신호에 대해 / 인식된 단어로) 출력한다.

-> 5문단에서는 1개의 기준 패턴을 가정하고 패턴 거리를 구하는 것에 대해서 설명했다면 실제로는 그것과 같은 방법으로 컴퓨터 저장된 모든 기준 패턴에 대해서 패턴 거리를 구하고, 그중 최솟값을 기준 패턴으로 선정한다는 말이죠. 그리고 그 기준 패턴에 해당하는 문자(배)열을 인식된 단어로 출력한다는 거네요.

-> 결국 해당 지문은 [컴퓨터 -> 입력된 말소리 -> 단어로 인식 -> 출력]과 같은 [음성 인식 기술]의 과정에 대해서 구체화해서 설명하는 글이었고 특히 [입력된 신호 -> 음소 추정 구간들의 배열 -> 입력 패턴 -> 기준 패

턴과의 비교 -> 단어 인식]에 대해서 더 중심을 두고 쓰인 글이었네요. 개인적으로 역대 기출 기술 지문 중 가장 불친절하게 쓰인 글이라고 생각해요. 문제는 쉬웠지만 지문을 이해하는 게 어려웠던 지문이었습니다. :)

7문단 정리

[5문단과 같은 방법으로 입력 패턴과 기준 패턴의 비교를 통해 모든 패턴 거리를 구하고 단어로 인식 및 출력]

전체 구조 잡기

1문단 - [음성 인식 기술 -> 입력 패턴과 기준 패턴 비교 -> 단어 인식]
 2~7문단 - [단어를 인식하기 위해 -> 입력된 신호 -> 음성 신호만 추출 -> 단위 구간으로 나누고 조합 -> 음소 추정 구간들 만들 -> 특징 벡터 추출 및 배열 -> 입력 패턴 생성 -> 기준 패턴과 비교 -> 패턴 거리 생성 -> 모든 패턴 거리 중 최솟값 선정 -> 단어 인식 및 출력]

[문제 해설]

43번 : ③

-> 3문단에 따르면 [특징 벡터 = 음소를 구별하기 위한 정보를 수치로 나타낸 것]에 해당하는데 음소의 특성을 잘 나타내는 정보는 사용하되 사람마다 다른 특성을 보이는 정보는 사용하지 않는다고 했어요. 이는 사람마다 다른 특성을 보이는 정보로는 음소를 정확하게 구별하기가 힘들기 때문이겠죠. 따라서 [개인의 독특한 목소리]는 사람마다 다른 특성을 보이는 정보에 해당하므로 특징 벡터로 사용하지 않을 겁니다. 적절하지 않아요.

- ① -> 1문단에서 [말소리 = 음소들의 시간적 배열]이라고 했죠. 적절해요.
- ② -> 2문단에서 [입력 신호 -> 잡음 제거 -> 음성 신호만 추출]이라고 했죠. 적절해요.
- ④ -> 4문단에서 [음소 추정 구간들의 특징 벡터를 -> 구간 순으로 배열 -> 입력 패턴]이라고 했으므로 적절해요.
- ⑤ -> 마지막 문단 [이와 같은 방법으로 ~ 그 중 최솟값이 되는 기준 패턴을 선정 -> 그 기준 패턴의 해당하는 문자열을 입력된 음성 신호에 대해 인식된 단어로 출력]따르면 입력된 신호에 대해서 저장된 모든 기준 패턴과 비교해보고 그 중 패턴 거리가 최소로 나오는 기준 패턴에 해당하는 문자열을 인식된 단어로 출력한다고 했죠. 적절해요.

44번 : ③

-> [기준 패턴의 음소의 개수]가 2개이면 조합되는 음소 추정 구간의 개수 역시 2개가 되어야 하겠죠. 따라서 단위 구간의 조합에 따라 만들어질 수 있는 음소 추정 구간의 종류는 S1, S2~3 또는 S1~2, S3가 될 것이고 이를 구간 순으로 배열한 음소 추정 구간의 배열 역시 [S1, S2~3], [S1~2, S3]와 같이 2개가 되겠죠.

- ① -> 기준 패턴의 음소 개수가 3개이면 만들어지는 음소 추정 구간 역시 S1, S2, S3와 같이 3개가 되겠죠. 각 추정 구간에 1개의 특징 벡터가 추출되고 그러한 특징 벡터를 구간 순으로 배열한 것이 입력패턴에 해당했어요. 따라서 입력 패턴에 들어있는 특징 벡터는 3개가 되겠죠. 적절해요.
- ② -> 기준 패턴의 음소 개수가 3개인 경우에는 지문에 따르면 [S1, S2, S3]와 수능 국어의 새로운 기준 김민수

같이 입력 패턴이 1개만 생성되기 때문에 1개의 기준 패턴과의 비교를 통해 만들어지는 패턴 거리는 1개가 되겠죠. 적절해요.

④

-> [기준 패턴의 음소의 개수]가 2개이면 조합되는 음소 추정 구간의 개수 역시 2개가 되어야 하겠죠. 따라서 단위 구간의 조합에 따라 만들어질 수 있는 음소 추정 구간의 종류는 S1, S2~3 또는 S1~2, S3가 될 것이고, 이를 구간 순으로 배열한 음소 추정 구간의 배열 역시 [S1, S2~3], [S1~2, S3]와 같이 2개가 되겠죠. 따라서 입력 패턴 역시 2개가 나오겠죠. 적절해요.

⑤

-> 기준 패턴의 음소 개수가 4개인 경우에는 단위 구간을 조합해서 만들어지는 음소 추정 구간의 개수가 [기준 패턴의 음소의 개수]보다 작을 수밖에 없으므로 일대일 대응이 불가능하게 되고, 패턴 비교 역시 불가능하겠죠. 적절해요.

45번 : ④

-> [a = 생성 가능한 입력 패턴과 기준 패턴 사이의 패턴 거리를 모두 구하고]의 처리 시간을 증가시키는 요인으로 옳은 것을 고르는 거죠. 이러한 a의 처리시간이 증가하기 위해서는 [비교 횟수]가 증가하면 되겠죠. 그러한 요인으로는 [저장된 기준 패턴의 개수 증가], [단위 구간의 시간 간격 감소] -> 단위 구간의 개수 증가 -> 조합될 수 있는 음소 추정 구간의 개수 증가 -> 음소 추정 구간의 배열 증가 -> 입력 패턴의 개수 증가]가 되겠네요. 따라서 정답은 ④인 [단위 구간 시간 간격의 감소]가 되겠네요.

①

-> 오히려 a를 처리하는 시간 역시 감소하게 되겠죠. 다만 그냥 2문단의 [처리 시간이 길어진다]와 일대일 대응으로 푸는 게 아니죠. 특징 벡터를 구성하는 정보의 가짓수가 감소하게 되면 음소 추정 구간에서 특징 벡터를 추출하는 시간이 감소하게 됩니다. 이렇게 되면 [음소 추정 구간 -> 특징 벡터를 추출]하는 시간이 감소하게 되면서 [입력 패턴을 생성]하는 시간 자체가 감소하게 된다는 말이죠. 그렇다면 그렇게 만들어지는 입력 패턴과 기준 패턴을 비교해서 패턴 거리를 구하는데 걸리는 시간인 [a를 처리하는 시간] 역시 감소한다고 생각할 수도 있겠죠. 사실 [a를 처리하는 시간]을 보는 기준에 따라 [무관] 또는 [감소한다]가 될 수 있을 것 같아요. 만약 [음소 추정 구간 -> 특징 벡터 추출 -> 입력 패턴 생성 -> 비교]와 같은 과정을 모두 포함한다면 [감소]가 될 것이고, 단순히 [비교]하는 시간만을 의미한다면 [무관]이 되겠죠. 다만 뭐가 되었든 [증가]는 아니므로 적절하지 않다고 볼 수 있겠네요.

②

-> [기준 패턴을 구성하는 음소의 개수의 감소]가 a의 처리 시간을 증가시킨다고 보기는 힘들죠. 물론 지문에서는 [3개 -> 2개]로 감소하면서 만들어질 수 있는 입력 패턴의 개수가 증가해서 처리 시간이 증가했지만 [2개 -> 1개]로 줄게 된다면 만들어지는 입력 패턴은 다시 1개가 될 것이고 시간은 다시 감소하게 되니까요. 따라서 알 수 없다고 보는 게 적절하겠네요.

③

-> 저장된 기준 패턴의 가짓수가 감소하면 입력 패턴이 비교해야 하는 기준 패턴이 줄어들게 되는 것이므로 오히려 a의 처리 시간은 감소한다고 봐야겠죠.

⑤

-> 음소 추정 구간의 개수가 감소하게 되면 그러한 음소 추정 구간의 배열로 만들어지는 입력 패턴의 개수가 감소하게 되므로 처리 시간은 감소한다고 봐야겠죠. 또한 만약 음소 추정 구간의 개수가 기준 패턴의 음소의 개수보다 작아지는 경우에는 비교 자체가 불가능하다고 했어요.

