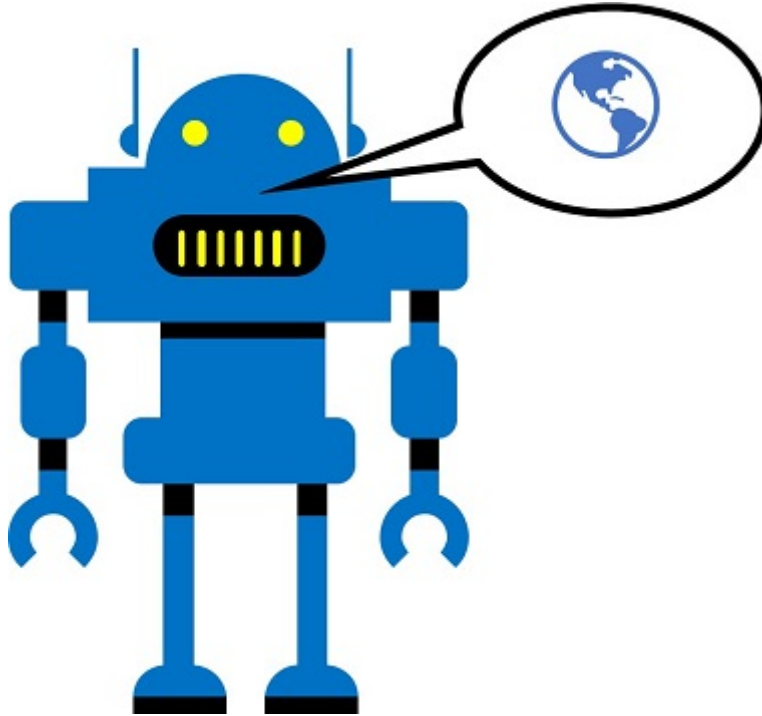


번역(Translation)

인류 문명이 발전할 수 있었던 원동력 중 하나는 서로 의사소통하는 능력이다. 대부분의 인간의 노력에서 의사소통이 핵심이다.



인공지능(AI)은 언어 간 텍스트 또는 음성을 번역하여 의사소통을 단순화하는 데 도움을 줄 수 있으며 국가 및 문화 간 의사소통의 장벽을 제거하는 데 도움이 될 수 있다..

Cognitive Services resource

Azure에서 여러 언어끼리 번역을 하기 위해서 Cognitive 서비스를 사용할 수 있다.

이미 만들어진 Cognitive 서비스가없다면 다음과 같은 순서로 Azure 구독에서 **Cognitive Services** 리소스를 생성할 수 있다.

1. 브라우저의 새로운 탭을 열고, Azure portal(<https://portal.azure.com>)을(<https://portal.azure.com>)을 입력하고 , Microsoft계정으로 로그인한다..
2. + 리소스 만들기 버튼을 클릭하고, *Cognitive Services* 서비스를 찾은 다음, **Cognitive Services** 리소스를 다음과 같은 내용으로 생성한다.
 - 이름: 유일한 이름을 입력한다(가능하면 영문과 숫자사용).
 - 구독: *Azure* 구독선택.
 - 위치: 가능한 위치(한국 중부 추천):
 - 가격책정계층: 표준 S0
 - 리소스 그룹: 원하는 유리한 이름(가능하면 영문과 숫자사용).
3. 배포가 완료될 때까지 기다린다. 그런 다음 Cognitive Services 리소스로 이동하여 *개요 페이지에서 링크를 클릭하여 서비스 키를 관리한다. 클라이언트 응용 프로그램에서 Cognitive Services 리소스에 연결하려면 엔드포인트와 키가 필요하다.

Cognitive Services 리소스에 있는 키와 엔드포인트 가져오기

Cognitive Services 리소스를 사용하기 위해서는, 클라이언트 응용프로그램에서는 엔드포인트와 인증 키가 필요하다.

1. Azure portal에서, Cognitive Services 리소스를 선택하고 **키 및 엔드포인트** 페이지를 선택한 다음 **키1** 을 복사하여 아래의 **YOUR_COG_KEY**.를 붙여 넣는다.
2. 리소스에 있는 **엔드포인트** 를 복사해서 아래의 **YOUR_COG_ENDPOINT**.에 붙여 넣는다.
3. 리소스에 있는 **위치** 를 복사해서 아래의 **YOUR_COG_REGION**.에 붙여 넣는다.
4. 셀을 선택한 다음 셀 왼쪽에있는 **셀 실행**(▶) 버튼을 클릭하여 아래 코드를 실행한다.

In []:

```
cog_key = 'YOUR_COG_KEY'
cog_endpoint = 'YOUR_COG_ENDPOINT'
cog_region = 'YOUR_COG_REGION'

print('Ready to use cognitive services in {} using key {}'.format(cog_region, cog_key))
```

텍스트 번역하기

Translator Text 서비스는 이름에서 알 수 있듯이 한 언어에서 다른 언어로 텍스트를 변환할 수 있도록 지원한다.

이 서비스에는 Python SDK가 없지만 REST 인터페이스를 사용하여 HTTP를 통해 엔드포인트에 요청을 제출할 수 있으며, **requests** 라이브러리를 사용하면 Python에서 비교적 쉽게 수행할 수 있다. 번역할 텍스트와 그 결과 번역된 텍스트에 대한 정보는 JSON 형식으로 교환한다.

다음 셀을 실행하여 이 작업을 수행하는 함수를 만든 다음 영어에서 프랑스어로 간단한 번역에 함수를 테스트해보자.

In []:

```
# Text Translation로 REST 요청을 보내는데 사용할 함수를 생성한다.
def translate_text(cog_region, cog_key, text, to_lang='fr', from_lang='en'):
    import requests, uuid, json

    # Text Translator 서비스 REST 요청을 위한 URL을 만든다.
    path = 'https://api.cognitive.microsofttranslator.com/translate?api-version=3.0'
    params = '&from={}&to={}'.format(from_lang, to_lang)
    constructed_url = path + params

    # Cognitive 리소스에 있는 키와 지역으로 요청 헤더를 구성한다.
    headers = {
        'Ocp-Apim-Subscription-Key': cog_key,
        'Ocp-Apim-Subscription-Region': cog_region,
        'Content-type': 'application/json',
        'X-ClientTraceId': str(uuid.uuid4())
    }

    # 번역할 텍스트를 Body에 추가한다.
    body = [{
        'text': text
    }]

    # 번역본을 가져온다.
    request = requests.post(constructed_url, headers=headers, json=body)
    response = request.json()
    return response[0]["translations"][0]["text"]

# 함수를 테스트한다.
text_to_translate = "Hello"

translation = translate_text(cog_region, cog_key, text_to_translate, to_lang='fr', from_lang='en')
print('{} -> {}'.format(text_to_translate, translation))
```

이 서비스는 영어 텍스트 "Hello"를 프랑스어 "Bonjour"로 번역했어야 했다.

언어는 표준 언어 약어 시스템을 사용하여 지정되며, 영어의 경우 *en*, 프랑스어의 경우 *fr*를 사용한다(한국어는 *ko*이다). 특정 문화를 포함하는 약어를 사용할 수도 있다. 이 약어는 동일한 언어가 서로 다른 지리적 영역(대개 철자법)에서 사용될 때 유용하다. 예를 들어 *en-US*는 미국식 영어를, *en-GB*는 영국식 영어를 나타낸다.

다음 셀을 실행하여 영국 영어로 된 텍스트를 이탈리아어로 번역해보라.

In []:

```
text_to_translate = "Hello"

translation = translate_text(cog_region, cog_key, text_to_translate, to_lang='it-IT', from_lang='en-GB')
print('{} -> {}'.format(text_to_translate, translation))
```

이제 US 영어를 중국어로 번역하는 테스트를 해보자

In []:

```
text_to_translate = "Hello"

translation = translate_text(cog_region, cog_key, text_to_translate, to_lang='zh-CN', from_lang='en-US')
print('{} -> {}'.format(text_to_translate, translation))
```

음성 번역(Speech Translation)

Speech 서비스를 사용하여 구어를 번역할 수 있다.

Speech 서비스는 마이크나 오디오 파일에서 음성 입력을 변환하는 데 사용할 수 있는 Python SDK를 제공하므로 이를 설치해야 한다..

In []:

```
! pip install azure.cognitiveservices.speech
```

이제 다음 셀을 실행하여 Speech SDK를 사용하여 소리 음성으로 번역하는 함수를 테스트 할 수 있다.

주의: 소리를 듣기 위해서 스피커가 필요하다.

In []:

```
# 하나의 언어에서 다른 언어로 텍스트를 번역하기 위한 함수를 만든다.
def translate_speech(cog_region, cog_key, audio_file=None, to_lang='fr-FR', from_lang='en-US'):
    from azure.cognitiveservices.speech import SpeechConfig, AudioConfig, ResultReason
    from azure.cognitiveservices.speech.translation import SpeechTranslationConfig, TranslationRecognizer

    # 음성 번역 서비스를 설정한다
    translation_config = SpeechTranslationConfig(subscription=cog_key, region=cog_region)
    translation_config.speech_recognition_language = from_lang
    translation_config.add_target_language(to_lang)

    # 음성 입력을 구성한다
    if audio_file is None:
        audio_config = AudioConfig() # 기본 입력을 사용(마이크)
    else:
        audio_config = AudioConfig(filename=audio_file) # 입력 파일 사용

    # 번역 인식도구를 생성하고 그것을 이용하여 음성 입력을 번역한다
    recognizer = TranslationRecognizer(translation_config, audio_config)
    result = recognizer.recognize_once()

    # 제대로 가져왔는가?
    translation = ''
    speech_text = ''
    if result.reason == ResultReason.TranslatedSpeech:
        speech_text = result.text
        translation = result.translations[to_lang]
    elif result.reason == ResultReason.RecognizedSpeech:
        speech_text = result.text
        translation = 'Unable to translate speech'
    else:
        translation = 'Unknown'
        speech_text = 'Unknown'

    # 번역본을 되돌려준다
    return speech_text, translation

# 함수를 테스트 한다
import os, IPython

file_name = 'english.wav'
file_path = os.path.join('data', 'translation', file_name)
speech, translated_speech = translate_speech(cog_region, cog_key, file_path, to_lang='es', from_lang='en-US')
result = '{} -> {}'.format(speech, translated_speech)

# 오디오를 실행하고 번역된 텍스트를 나타낸다
IPython.display.display(IPython.display.Audio(file_path, autoplay=True),
                        IPython.display.HTML(result))
```

"to" 언어에는 2자로 된 언어 코드(예: *en*)를 사용하여 식별해야 하며, "from" 언어에는 문화 지표(예: *en-US*)가 포함되어야 한다(한국어의 경우 *ko-kr*임).

불어에서 영어로 번역해 보자

In []:

```
import os, IPython

file_name = 'french.wav'
file_path = os.path.join('data', 'translation', file_name)
speech, translated_speech = translate_speech(cog_region, cog_key, file_path, to_lang='en', from_
lang='fr-FR')
result = '{} -> {}'.format(speech, translated_speech)

# 오디오를 실행하고 번역된 텍스트를 나타낸다.
IPython.display.display(IPython.display.Audio(file_path, autoplay=True),
                        IPython.display.HTML(result))
```

심화학습

번역에 대해서 좀더 살펴보려면 [텍스트 번역기\(Translator Text\)](https://docs.microsoft.com/azure/cognitive-services/translator/) (<https://docs.microsoft.com/azure/cognitive-services/translator/>)와 [Speech](https://docs.microsoft.com/azure/cognitive-services/speech-service/index-speech-translation) 서비스를 활용한 음성번역 서비스 (<https://docs.microsoft.com/azure/cognitive-services/speech-service/index-speech-translation>) 문서를 참조하기 바란다.