딥러닝 활용 : 언어분석(RNN, 자연어 처리)

2018.07.05

|  |  |  |
| --- | --- | --- |
| 작성자 | 검토자 | 승인자 |
| 이가영 |  |  |

© Copyright 2018 Medicisoft Co.,Ltd.

[1. RNN 2](#_Toc518574687)

[2. 딥러닝 활용 언어분석 – Rnn 모델을 이용한 글자 생성 예제 2](#_Toc518574688)

[3. 참고문헌 5](#_Toc518574689)

# RNN

##### **정의**

* Hidden Node가 방향을 가진 엣지로 연결돼 순환구조를 이루는 (directed cycle) 인공신경망의 한 종류이다.
* 음성, 문자 등 순차적으로 등장하는 데이터에 대한 처리에 적합한 모델이다.

##### 구조

그림 1) RNN의 구조

* Sequence 길이에 관계없이 input과 output을 받아들일 수 있는 네트워크 구조  
  필요에 따라 다양하고 유연하게 구조를 만들 수 있다는 점이 가장 큰 장점이다.
* 기본 구조는 그림 1과 같다. 녹색 박스는 hidden state를 의미한다. 빨간 박스는 input x, 파란 박스는 output y이다. 현재 상태의 hidden state ht는 직전 시점의 hidden state ht−1를 받아 갱신된다.  
  여기서 현재 상태의 output yt는 ht를 전달받아 갱신되는 구조이다. 수식에서도 알 수 있듯 hidden state의 활성함수(activation function)은 비선형 함수인 하이퍼볼릭탄젠트(tanh)이다.

# 딥러닝 활용 언어분석 – Rnn 모델을 이용한 글자 생성 예제

* 첨부 source 코드 파일 참고

##### 개발 환경

* OS : Windows10
* 언어 : Python3.4.3

##### Preprocess

* 글자 생성을 진행하기 위해서, 학습에 사용할 데이터 셋을 준비한다.
* Ex) 다음과 같은 dataset[3]으로 학습을 진행한다고 가정을 한다.

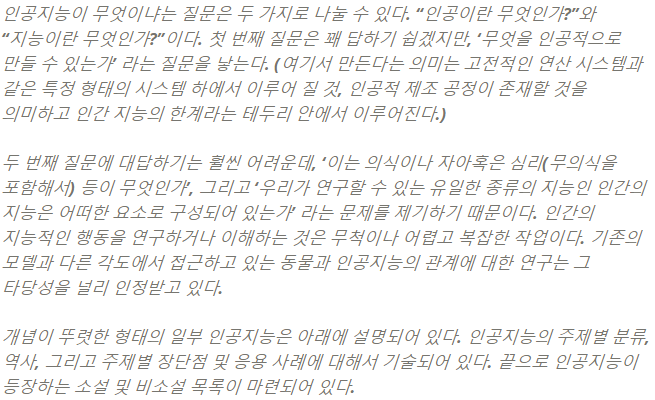


그림 2) 학습에 사용하는 글자 묶음 예시[3]

* 이 데이터를 Neural Networks 학습에 사용하기 위해서는 전처리가 필요하다.

문장들을 모두 개별 글자 단위로 쪼개고, 각각의 글자를 나타내는 숫자를 하나씩 할당해준다. 이후, 데이터셋에 포함된 전체 글자들로 사전을 만들고 모든 글자들을 할당된 숫자로 mapping한다.



그림 3) 전체 문장 데이터셋을 글자 단위로 쪼개고, 글자마다 숫자를 할당

##### Training

* 전처리(Preprocess)가 끝난 데이터를 이용해서 RNNs을 이용해서 학습을 진행한다.
* 학습을 진행하기 위해서는 input data와 target data가 필요하다. 이때 input data와 target data를 아래와 같이 구성한다.  
  - Input Data : 전체 문장 중 일정 길이의 글자들의 배열
* Target Data : 전체 데이터중 Input Data를 한 글자 뒤로 민 배열

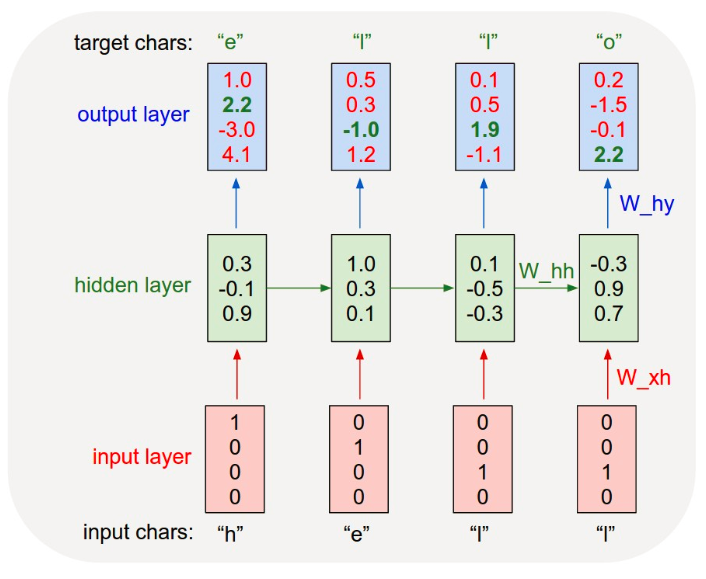


그림 4) Input data와 Target data 구성

* Ex) 전체 데이터 셋이 “hello”라는 글자로 구성되어 있다면, 일정길이(4글자)로 글자를 나눈다.  
  - Input Data : hell (전체 문장 중 일정 길이의 글자들의 배열)
* Target Data : ello (전체 데이터 중 input data를 한 글자 뒤로 민 배열)

이 때, RNNs의 학습을 진행하면 RNNs는 ‘“hell” 다음에 올 글자는 “o”이다.’라는 사실을 학습할 수 있다.

* 이런 과정을 전체 글자 데이터셋에 대해 진행하면, 결과적으로 RNNs는 주어진 데이터 셋에 대해서 “주어진 글자묶음 이후에 어떤 글자가 오는 것이 적절한지”를 학습할 수 있다.

##### Sampling

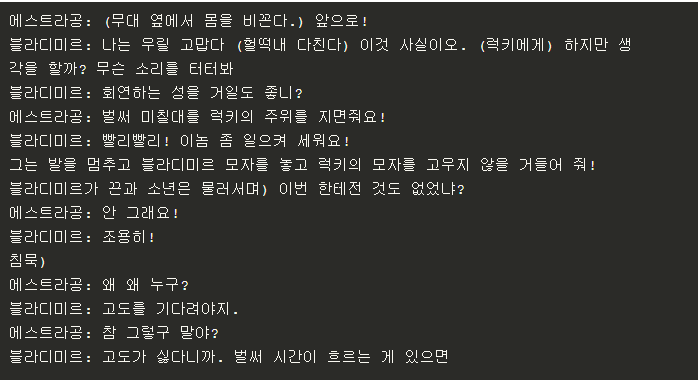
* snmpling이란 학습된 확률분포에서 임의의 샘플들을 추출하는 기법이다.
* 학습된 결과를 토대로 샘플링을 통해 글자를 생성해본다.
* 우리가 볼 예제에서는 첫 시작글자(‘ ‘-공백)를 RNN의 input으로 넣고, 공백으로부터 다음 글자를 prediction한다. 이후, 다시 이 prediction 값을 RNN의 input으로 넣고 prediction을 진행하는 과정을 반복한다.
* Ex) 사무엘 베케트의 희곡 “고도를 기다리며”를 데이터 셋으로 학습을 진행한 결과  
  

그림 5) 사무엘 베케트의 희곡 “고도를 기다리며”를 데이터셋으로 학습을 진행한 이후에 샘플링한 결과

# 참고문헌

##### <https://deeplearning4j.org/kr/lstm>

##### 텐서플로우(TensorFlow)를 이용해서 글자 생성 해보기 – RNNs 예제 – Char-RNN <http://solarisailab.com/archives/1620>

##### <https://ratsgo.github.io/natural%20language%20processing/2017/03/09/rnnlstm/>