토큰(token), 토큰화(tokenization)

‘He loves the cat.’ → [he, loves, the, cat] → [10, 14, 12, 13]

영어 문장의 토큰화 : 모두 소문자, 구두점 삭제, 공백 기준 분리.

‘그는 고양이를 사랑한다 ' → [그, -는, 고양이, -를, 사랑, 하-, -ㄴ다]

한국어 문장의 토큰화 : 형태소 분석 (조사가 발달), KoNLPy

(형태소 : 뜻을 가진 가장 작은 말의 단위)

원-핫 인코딩

[10, 14, 12, 13] : mapping을 위한 수, 크기를 의도하지 않음

he, loves : 그대로 사용하면 loves가 he보다 더 큰 활성화 출력

10 -> [0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. ....]

14 -> [0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. ....]

다중 분류에서의 확률, 크로스 엔트로피 손실 계산 가능.

원-핫 인코딩의 단점 : 공간적 낭비

희소 표현, 희소 벡터 : 500차원의 원-핫 인코딩 배열의 경우

500개 성분 중 한 성분만 1, 나머지 성분은 0

원-핫 인코딩 x 뉴런 개수
+은닉 상태 x 뉴런 개수
+뉴런 개수 = 파라미터

500 x 8 + 8 x 8 + 8
= 4072 (너무 많은 가중치)

```
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
simple_rnn (SimpleRNN)       (None, 8)                 4072
_________________________________________________________________
dense (Dense)                (None, 1)                 9
=================================================================
Total params: 4,081
Trainable params: 4,081
Non-trainable params: 0
_________________________________________________________________
```

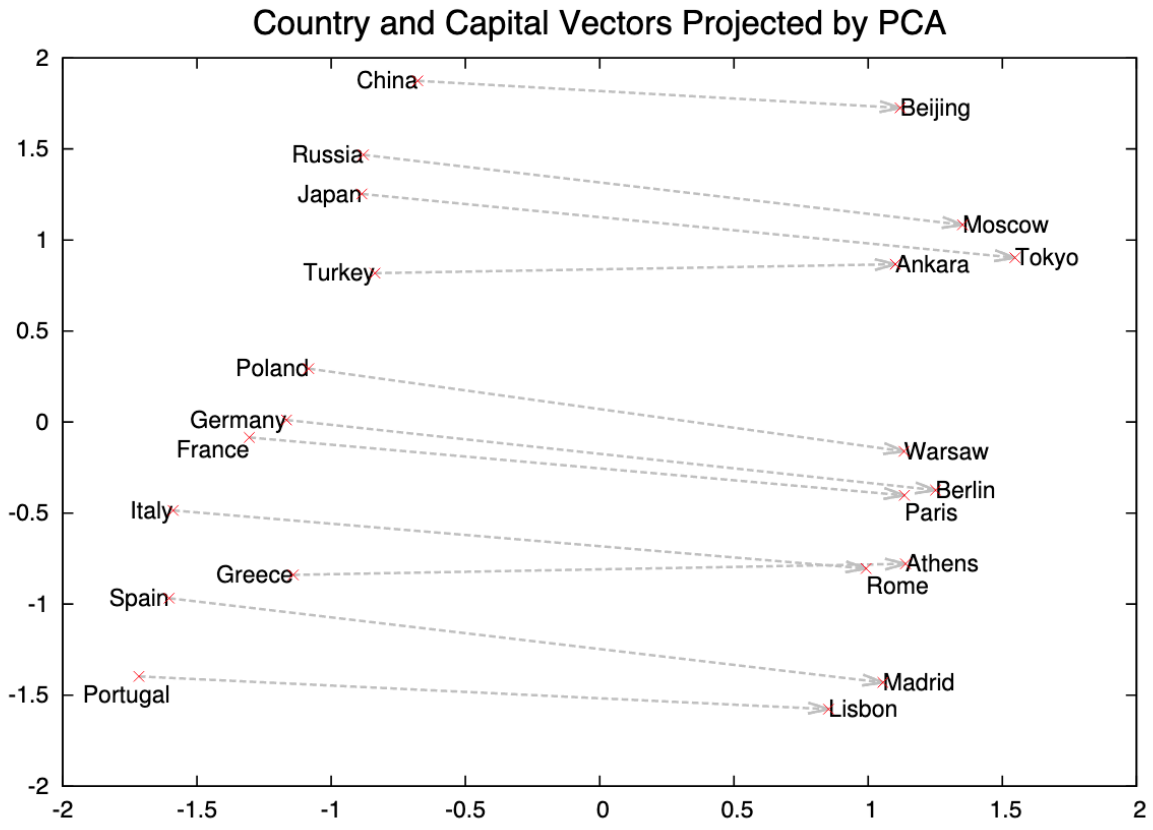
단어 임베딩 (word embedding)

Embed (타동사) : 끼워 넣다, 꽂아 넣다, 파묻다

Spain = (-1.6, -1), Madrid = (1, -1.5)
France = (-1.3, -0.1), Paris = (1.2, -06.)

Madrid – Spain + France
= (1 + 1.6 - 1.3 , -1.5 + 1 – 0.1)
= (1.3 , - 0.6) ~ Paris

마드리드 – 스페인 + 프랑스
= 스페인의 수도 – 스페인 + 프랑스
= [ ] 의 수도 + 프랑스
= 프랑스의 수도 = 파리



(Two-dimensional PCA projection of the 1000 -dimensional Skip-gram vectors of countries and their capital cities.)

Word2vec Korean Word2Vec

: 단어의 효율적인 의미 추정 기법

한국 – 서울 + 파리 = 프랑스
사랑 + 이별 = ?

맥도날드 – 빅맥 + 와퍼 = ?
기계학습 + 인문학 = ?
도서관 – 책 = ?

단어 임베딩

cat -> [0.2, 0.1, 1.3, ...]

임베딩 벡터 x 뉴런 개수
+ 은닉 상태 x 뉴런 개수
+ 8개 절편

128 + 64 + 8 =200

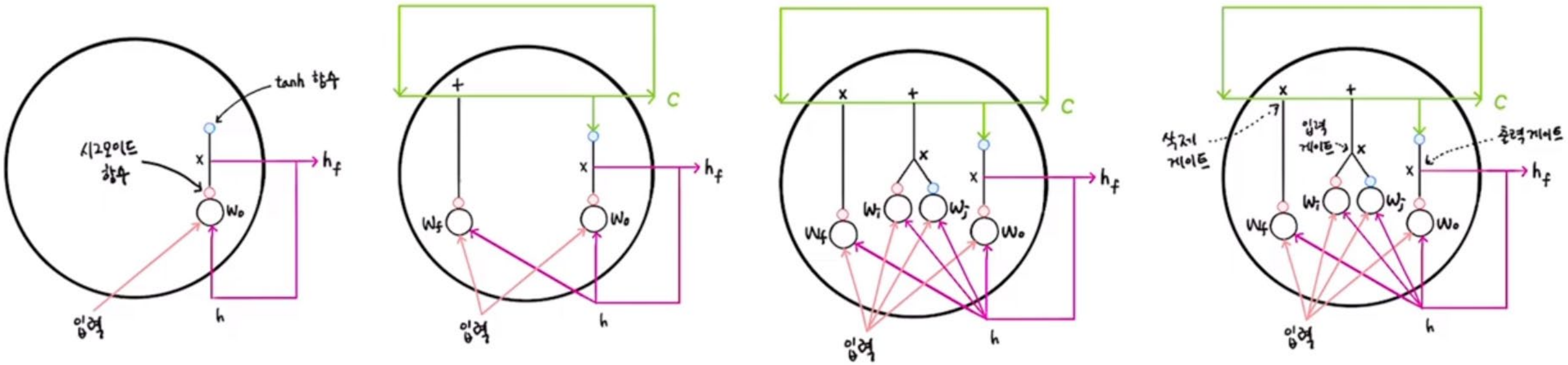
SimpleRNN층의 파라미터가 원-핫 인코딩(4072)에 비해 확연하게 줄어듦.

```
model2.summary()
```

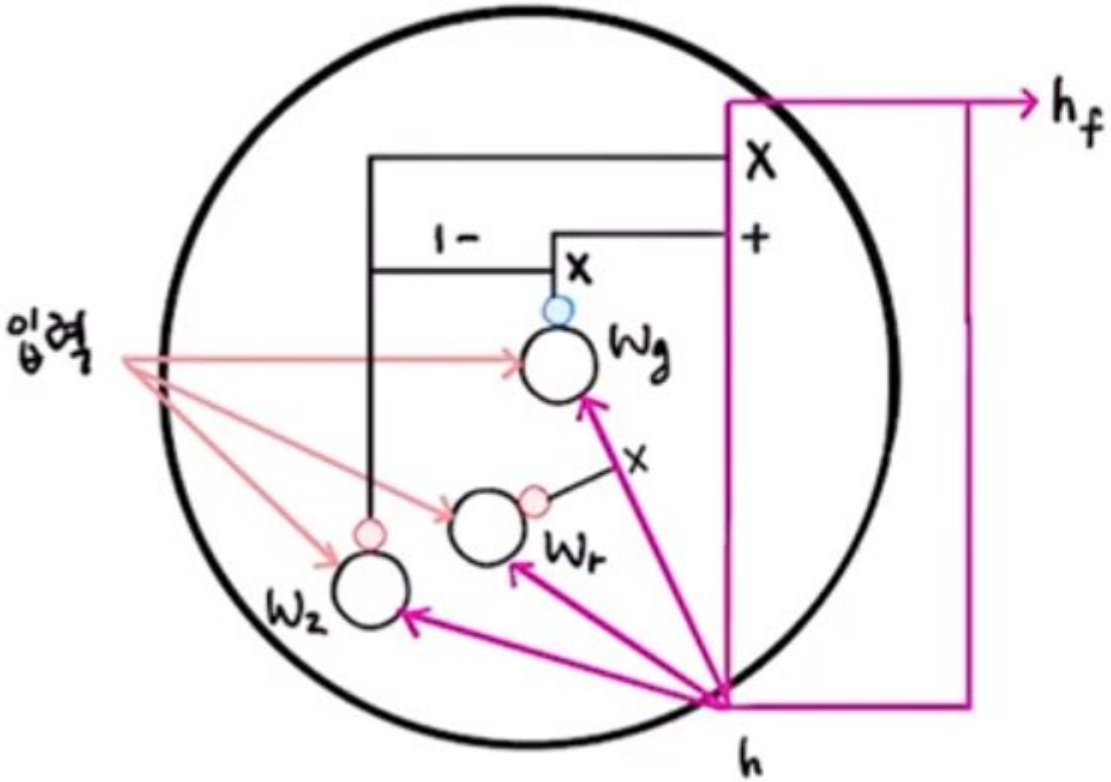
```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, 100, 16)           8000
_________________________________________________________________
simple_rnn_1 (SimpleRNN)     (None, 8)                 200
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 9
=================================================================
Total params: 8,209
Trainable params: 8,209
Non-trainable params: 0
_________________________________________________________________
```

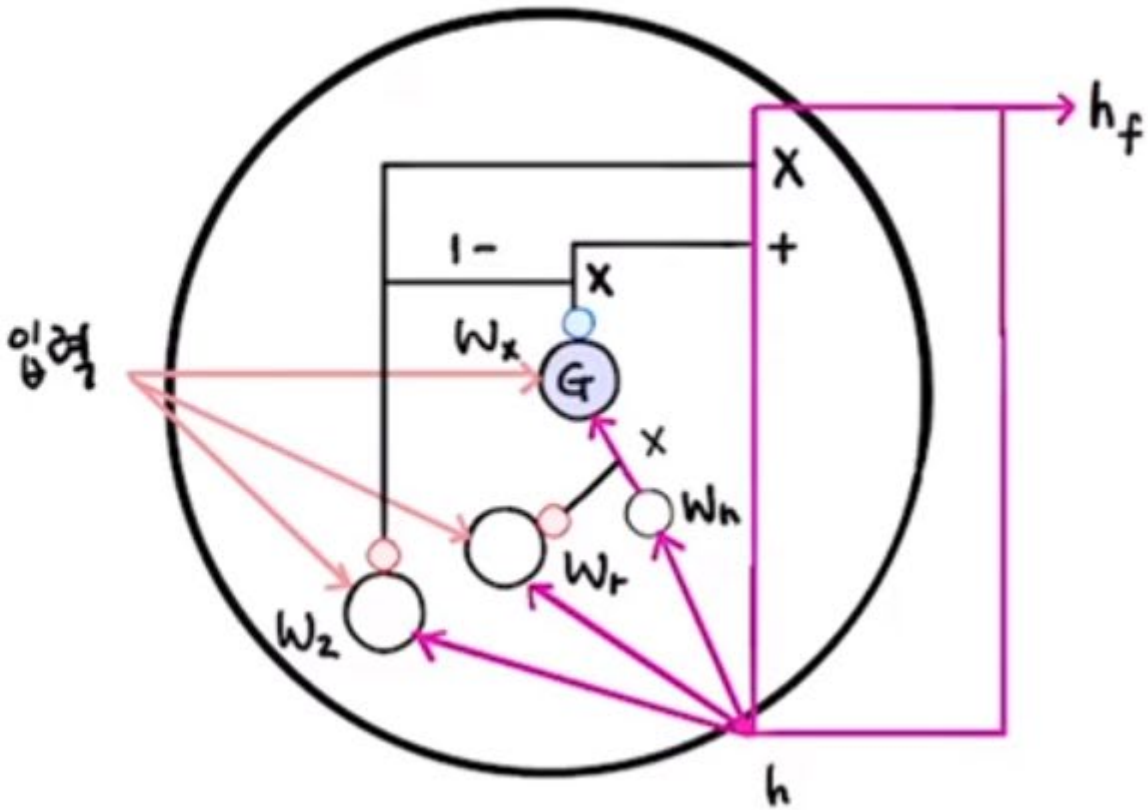
LSTM (Long Short – Term Memory), 장단기 기억

h : 은닉 상태, 단기 상태, c : 셀(cell) 상태, 장기 상태



GRU 셀





SimpleRNN

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
simple_rnn (SimpleRNN)       (None, 8)                 4072
_________________________________________________________________
dense (Dense)                (None, 1)                 9
=================================================================
Total params: 4,081
Trainable params: 4,081
Non-trainable params: 0
_________________________________________________________________
```

```
Epoch 54/100
313/313 [==============================]
- 26s 82ms/step - loss: 0.3964
- accuracy: 0.8282
- val_loss: 0.4532
- val_accuracy: 0.7930
```

LSTM

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, 100, 16)           8000
_________________________________________________________________
lstm (LSTM)                  (None, 8)                 800
_________________________________________________________________
dense (Dense)                (None, 1)                 9
=================================================================
Total params: 8,809
Trainable params: 8,809
Non-trainable params: 0
_________________________________________________________________
```

```
Epoch 53/100
313/313[==============================]
- 3s 9ms/step - loss: 0.4007
- accuracy: 0.8229
- val_loss: 0.4324
- val_accuracy: 0.7948
```

참고 자료

- 혼자 공부하는 머신러닝 + 딥러닝
- Tomas Mikolov 외, Distributed Representations of Words and Phrases and their Compositionality
- Tomas mikolob 외, **Efficient Estimation of Word Representations in Vector Space**
- https://www.youtube.com/watch?v=ub8S29bF6rk