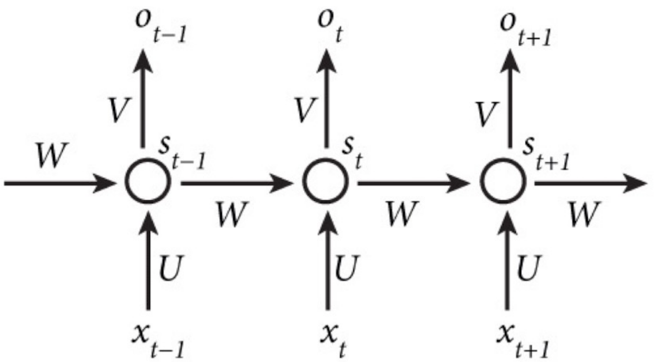
Recurrent Neural Network (RNN)

2-2

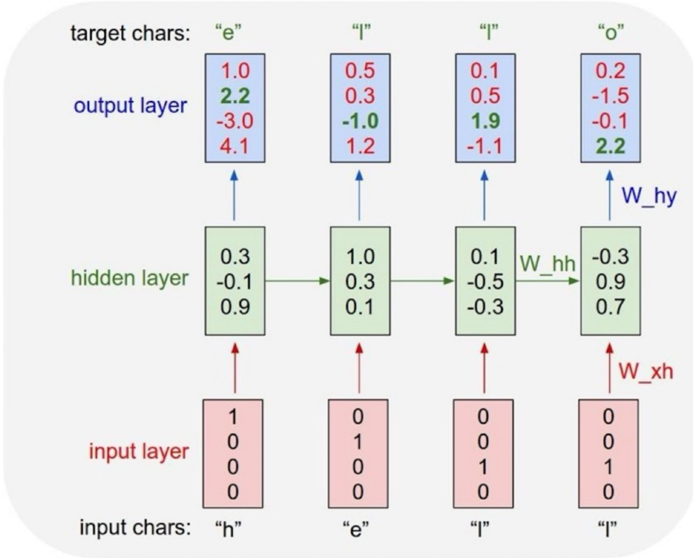
Intro to RNN

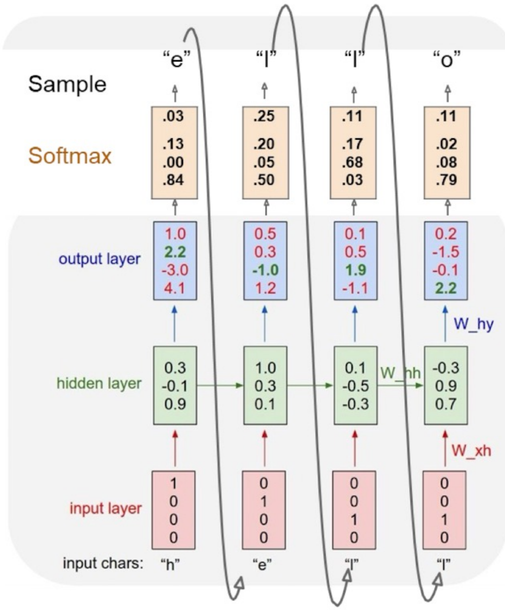
RNN 기본구조



- Parameters: W, V, U
- Inputs: x_t
- Hidden State: $s_t = f(Ux_t + Ws_{t-1})$
- Outputs: $o_t = \text{softmax}(Vs_t)$

RNN 예시: Character-level Language Model

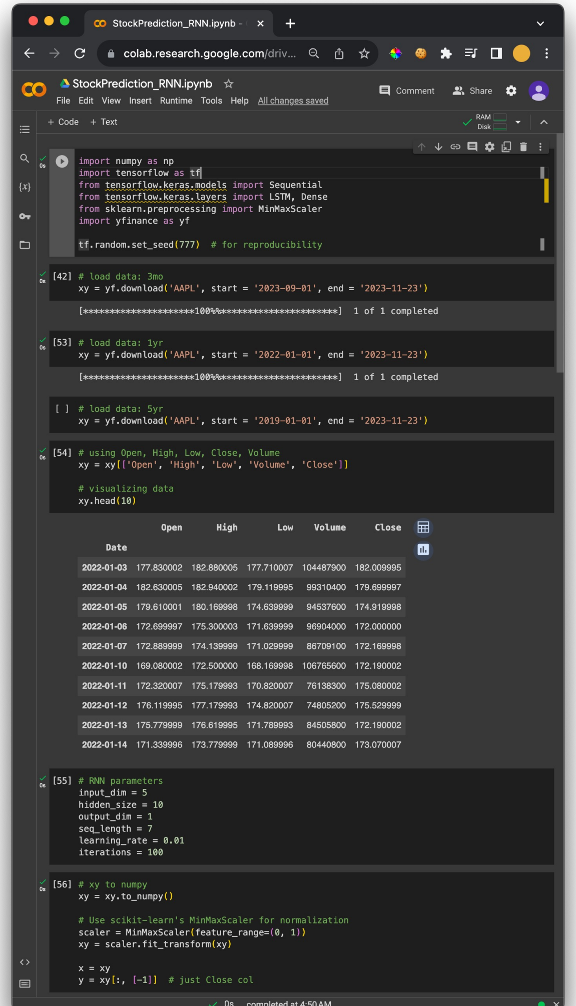


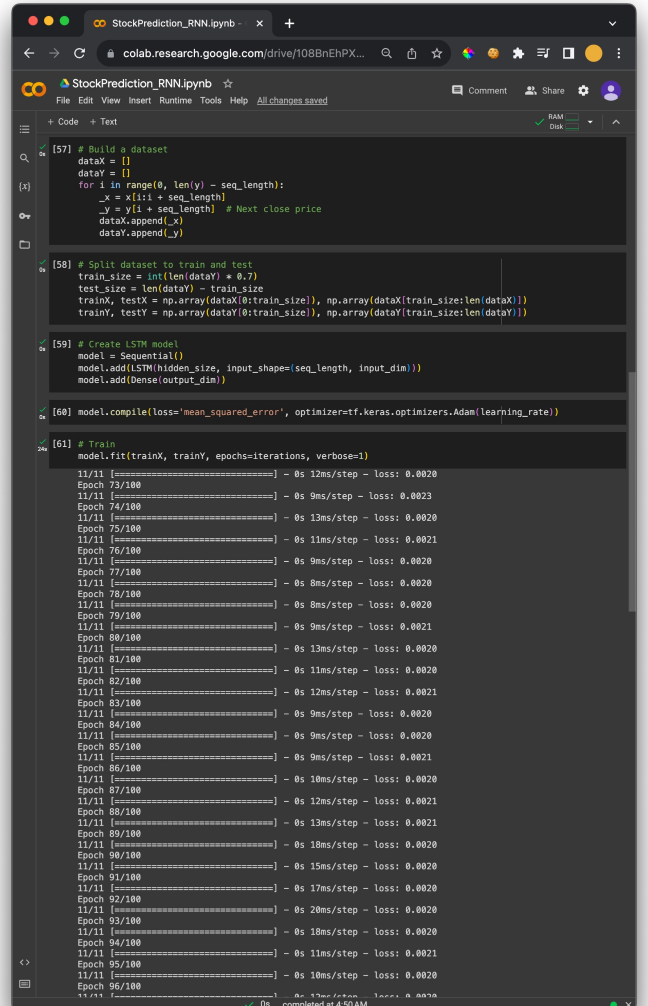


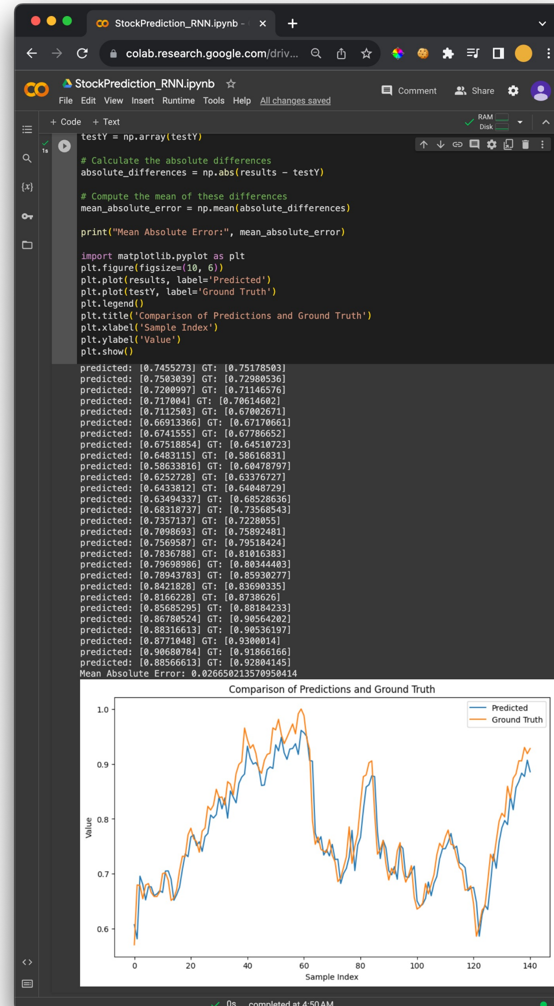
- RNN은 순차적인(sequential) 데이터를 활용하는 머신러닝 기법이다
- 순차적으로 의존적인 여러 NN layer를 활용해 Input 값과 가중치 parameter를 기반으로 예측하고 싶은 Output 값을 산출한다
- 지도학습을 통해 Input과 Output이 주어졌을 때 평균오차를 최소화하는 가중치 parameter를 학습시키고 학습된 parameter를 테스트하는 과정을 반복
- RNN의 활용 예시로 [h, e, l, o] 가 주어졌을 때 "hello"를 출력하는 Language Model의 일종이 있다

Source: Nature, Stanford, YIG

Implementation: Python (keras, sklearn)





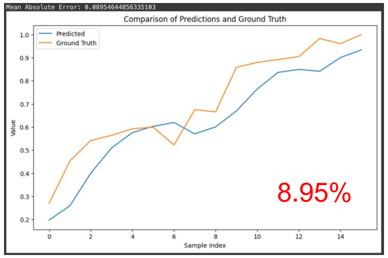


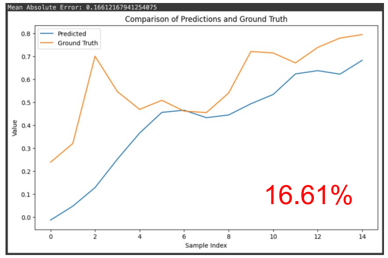
Source: AIC2120, YIG, colab link: https://drive.google.com/file/d/108BnEhPXHNgVEARk_H_Bz2tnQeg1W3PO/view?usp=sharing

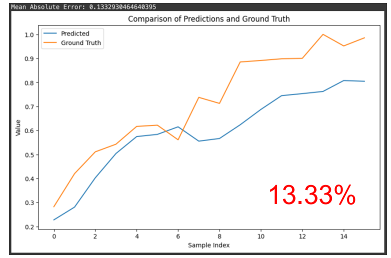
Result analysis: Region effect

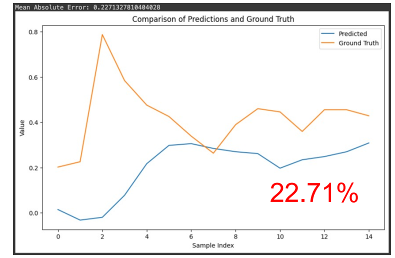
	S&P500 (SPY)	KOSPI 200 (KODEX 200)	NASDAQ 100 (QQQ)	KOSDAQ 150 (TIGER 코스닥 150)
3mo	8.95%	16.61%	13.33%	22.71%
1yr	2.24%	2.18%	3.15%	3.88%
5yr	1.81%	1.24%	1.58%	2.17%

3mo

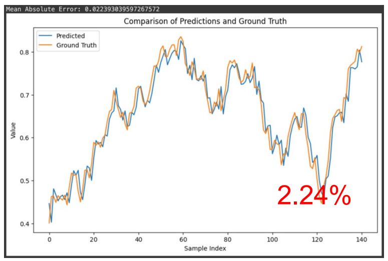


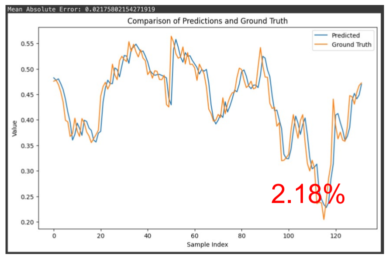


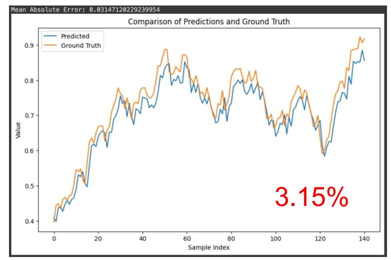


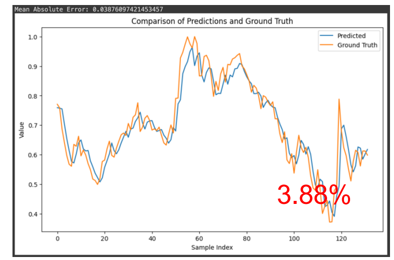


1yr

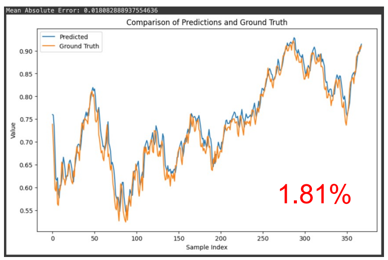


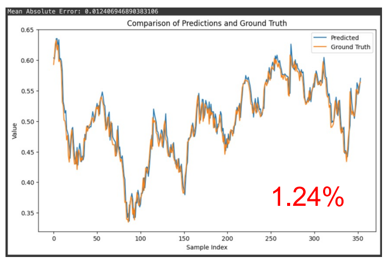


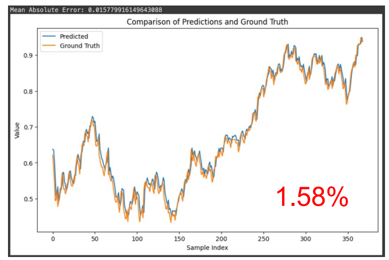


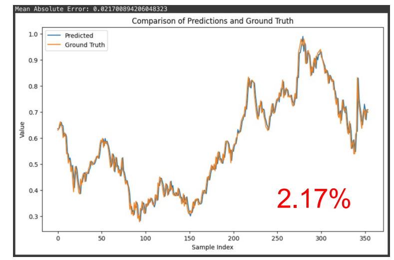


5yr









- RNN 기법을 활용한 주가 예측의 정확도가 지역의 영향을 받는지 비교를 위해 미국과 한국 주식시장을 비교
- 1년치보다 많은 데이터를 사용할 시 지역의 영향은 미미하지만 단기 데이터를 사용할 경우, 한국시장에 대한 정확도가 상대적으로 많이 감소

Source: YIG

Result analysis: Industry effect

	엔터			조선			식품			변압기		
	JYP	HYBE	SM	삼성중공업	HSD엔진	한국카본	삼양식품	롯데웰푸드	우양	제룡전기	HD일렉	효성중공업
3mo	22.61%	13.45%	7.67%	9.24%	15.78%	12.02%	17.94%	46.47%	80.31%	9.37%	14.85%	14.44%
1yr	2.99%	2.52%	4.77%	7.02%	3.91%	3.73%	4.91%	2.77%	5.20%	2.75%	4.54%	6.34%
5yr	1.75%	1.64%	2.90%	1.67%	1.81%	2.14%	1.65%	1.34%	3.97%	2.80%	3.74%	3.19%

- RNN 기법을 활용한 주가 예측의 정확도가 산업의 영향을 받는지 비교를 위해 엔터, 조선, 식품, 변압기 산업을 비교
- 같은 산업 내 정확도 수치 간 유효한 연관성 부족
- 변동성이 더 심한 개별주식 데이터를 다루기에 충분히 오차를 줄이기 위해 5년치 이상의 데이터 필요
- 결론: 결국 온전히 수치적인 접근이기에 지역, 산업 등을 떠나 모델 features와 데이터 기간에 따라 정확도 변화

Source: YIG