Text Generation

(using

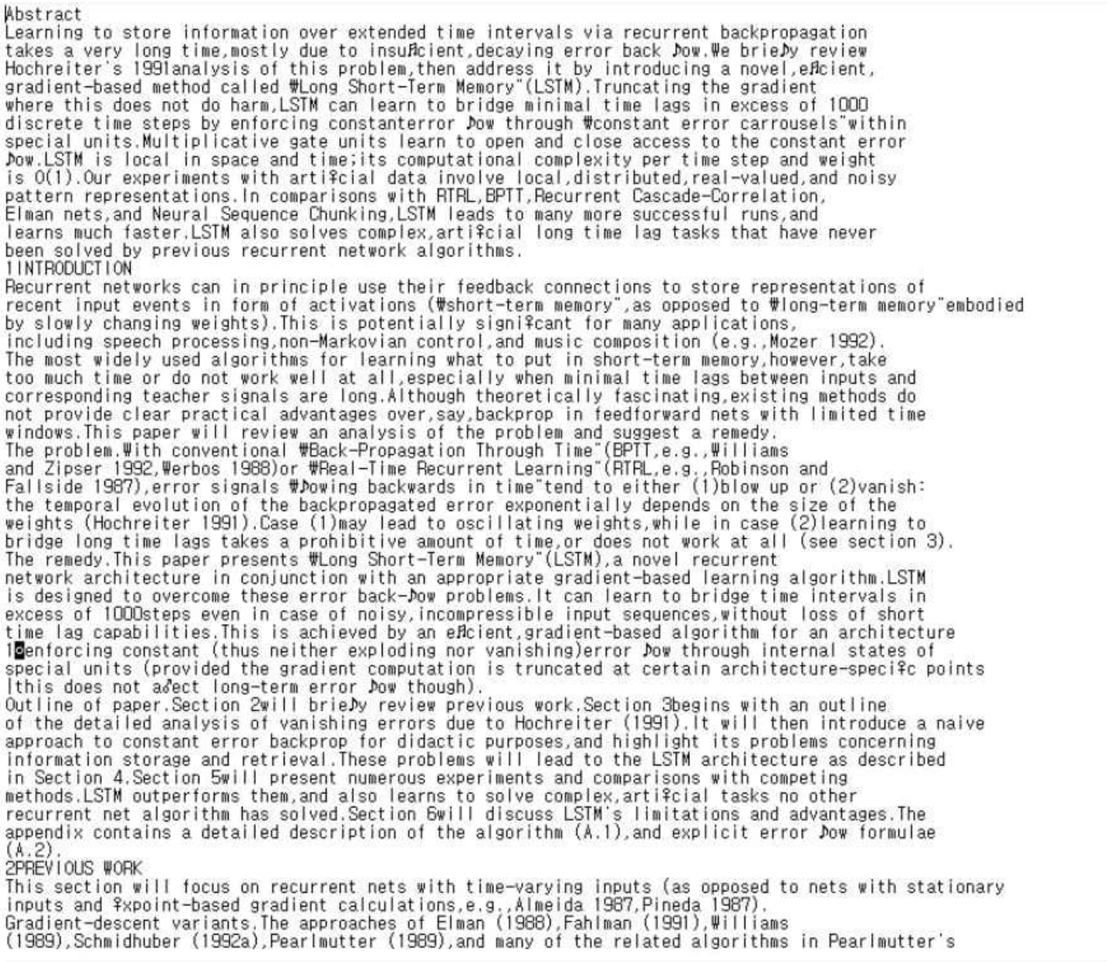
http://www.bioinf.jku.at/publications/older/2604.pdf

dataset)

V2016120 김태형

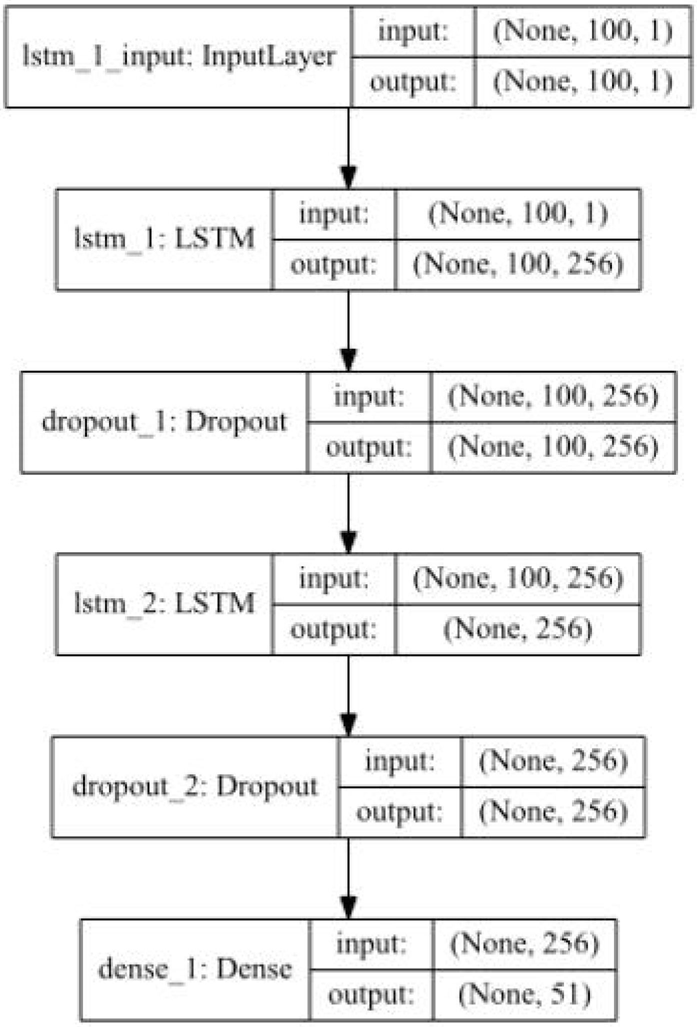
이번시간에는 RNN model을 기반으로 generative models을 만들어 보겠습니다.
추가적으로 예측모델(Predictive models)을 만드는데 그럴듯한 스퀀스를 생성합니다.

이 예제에서는 원하는 large text를 이용하여 학습을 시켜 스퀀스 data를 생성할 수 있습니다.

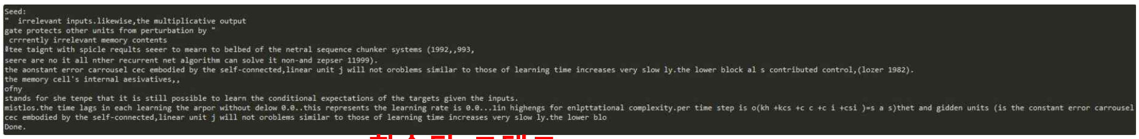


```
Abstract
Learning to store information over extended time intervals via recurrent backpropagation
takes a very long time,mostly due to insufficient,decaying error back flow.We briefly review
Hochreiter's 1991analysis of this problem,then address it by introducing a novel,efficient,
gradient-based method called ₩Long Short-Term Memory"(LSTM).Truncating the gradient
where this does not do harm,LSTM can learn to bridge minimal time lags in excess of 1000
discrete time steps by enforcing constanterror flow through ₩constant error carrousels"within
special units.Multiplicative gate units learn to open and close access to the constant error
flow.LSTM is local in space and time;its computational complexity per time step and weight
is O(1).Our experiments with artificial data involve local,distributed,real-valued,and noisy
pattern representations.In comparisons with RTRL,BPTT,Recurrent Cascade-Correlation,
Elman nets,and Neural Sequence Chunking,LSTM leads to many more successful runs,and
learns much faster.LSTM also solves complex,artificial long time lag tasks that have never
been solved by previous recurrent network algorithms.
1INTRODUCTION
Recurrent networks can in principle use their feedback connections to store representations of
recent input events in form of activations (₩short-term memory",as opposed to ₩long-term memory"embodied
by slowly changing weights).This is potentially significant for many applications,
including speech processing,non-Markovian control,and music composition (e.g.,Mozer 1992).
The most widely used algorithms for learning what to put in short-term memory,however,take
too much time or do not work well at all,especially when minimal time lags between inputs and
corresponding teacher signals are long.Although theoretically fascinating,existing methods do
not provide clear practical advantages over,say,backprop in feedforward nets with limited time
windows.This paper will review an analysis of the problem and suggest a remedy.
The problem.With conventional ₩Back-Propagation Through Time"(BPTT,e.g.,Williams
and Zipser 1992,Werbos 1988)or ₩Real-Time Recurrent Learning"(RTRL,e.g.,Robinson and
Fallside 1987),error signals ₩flowing backwards in time"tend to either (1)blow up or (2)vanish:
the temporal evolution of the backpropagated error exponentially depends on the size of the
weights (Hochreiter 1991).Case (1)may lead to oscillating weights,while in case (2)learning to
bridge long time lags takes a prohibitive amount of time,or does not work at all (see section 3).
The remedy.This paper presents ₩Long Short-Term Memory"(LSTM),a novel recurrent
network architecture in conjunction with an appropriate gradient-based learning algorithm.LSTM
is designed to overcome these error back-flow problems.It can learn to bridge time intervals in
excess of 1000steps even in case of noisy,incompressible input sequences,without loss of short
time lag capabilities.This is achieved by an efficient,gradient-based algorithm for an architecture
1enforcing constant (thus neither exploding nor vanishing)error flow through internal states of
special units (provided the gradient computation is truncated at certain architecture-specific points
|this does not affect long-term error flow though).
Outline of paper.Section 2will briefly review previous work.Section 3begins with an outline
of the detailed analysis of vanishing errors due to Hochreiter (1991).It will then introduce a naive
approach to constant error backprop for didactic purposes,and highlight its problems concerning
information storage and retrieval.These problems will lead to the LSTM architecture as described
in Section 4.Section 5will present numerous experiments and comparisons with competing
methods.LSTM outperforms them,and also learns to solve complex,artificial tasks no other
recurrent net algorithm has solved.Section 6will discuss LSTM's limitations and advantages.The
appendix contains a detailed description of the algorithm (A.1),and explicit error flow formulae
(A.2).
2PREVIOUS WORK
This section will focus on recurrent nets with time-varying inputs (as opposed to nets with stationary
inputs and fixpoint-based gradient calculations,e.g.,Almeida 1987,Pineda 1987).
Gradient-descent variants.The approaches of Elman (1988),Fahlman (1991),Williams
(1989),Schmidhuber (1992a),Pearlmutter (1989),and many of the related algorithms in Pearlmutter's
```

[Data]

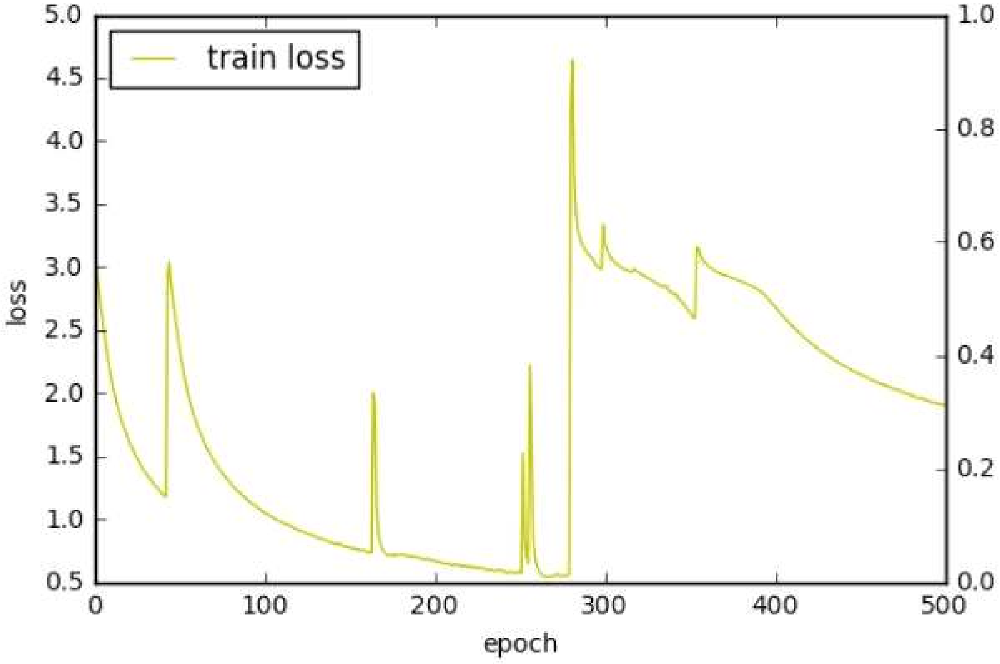


[모델 정의]



```
Seed:
"  irrelevant inputs.likewise,the multiplicative output
gate protects other units from perturbation by "
 crrrently irrelevant memory contents
#tee taignt with spicle reqults seeer to mearn to belbed of the netral sequence chunker systems (1992,,993,
seere are no it all nther recurrent net algorithm can solve it non-and zepser 11999).
the aonstant error carrousel cec embodied by the self-connected,linear unit j will not oroblems similar to those of learning time increases very slow ly.the lower block al s contributed control,(lozer 1982).
the memory cell's internal aesivatives,,
ofny
stands for she tenpe that it is still possible to learn the conditional expectations of the targets given the inputs.
mistlos.the time lags in each learning the arpor without delow 0.0..this represents the learning rate is 0.0...1in highengs for enlpttational complexity.per time step is o(kh +kcs +c c +c i +csi )=s a s)thet and gidden units (is the constant error carrousel
cec embodied by the self-connected,linear unit j will not oroblems similar to those of learning time increases very slow ly.the lower blo
Done.
```

[학습된 모델로 Text Generation]


train loss
loss
epoch
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0
100
200
300
400
500
1.0
0.8
0.6
0.4
0.2
0.0

[Epochs 시각화]