Deep Learning for Natural Language Understanding:

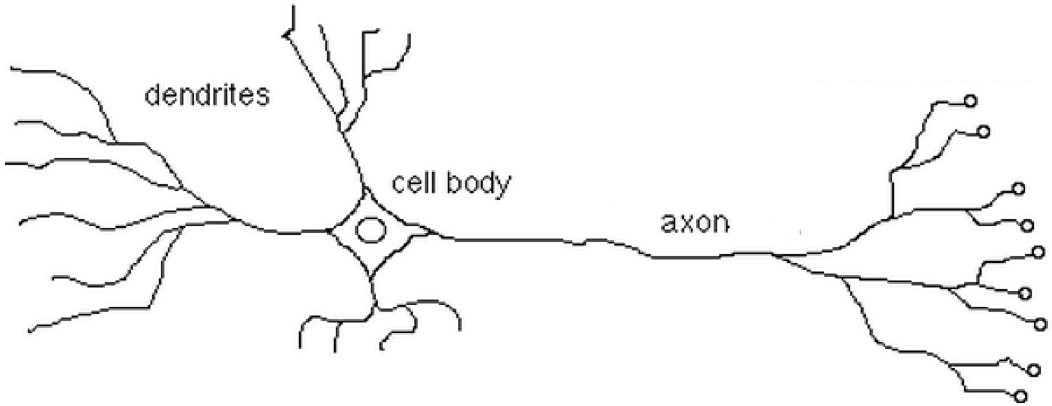
Modeling Meaning of Text

Xiaodan Zhu

National Research Council Canada, Ottawa

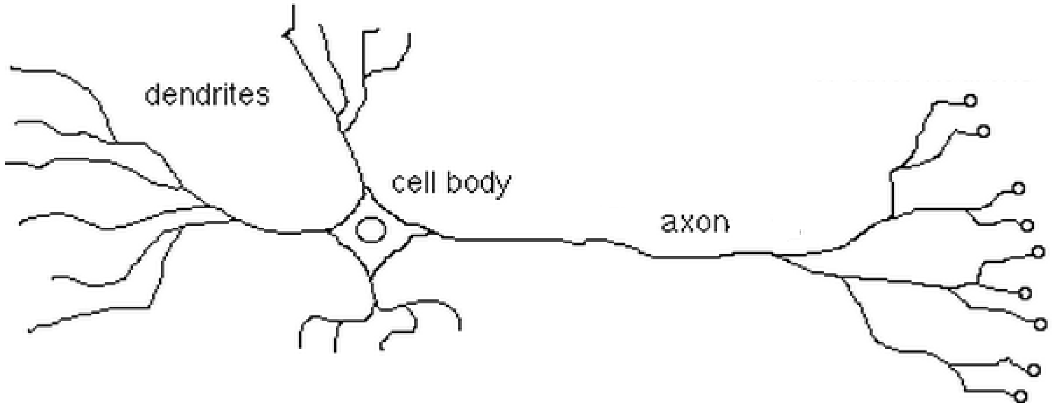
- Deep Learning
  - A set of machine learning algorithms that model high-level abstractions in data by using model architectures (often ***neural networks***).
  - It has significantly improved the states of the art on many problems in many fields.
    - Natural language processing
    - Speech recognition
    - Image/video processing

Biological Neuron



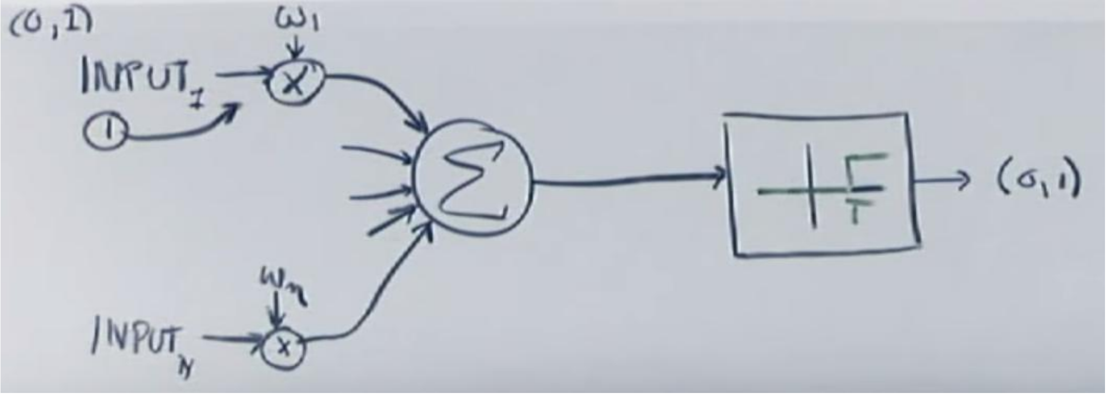
A network of simple, non-intelligent decisions can lead to intelligence.

Biological Neuron



A network of simple, non-intelligent decisions can lead to intelligence.

Artificial Neuron



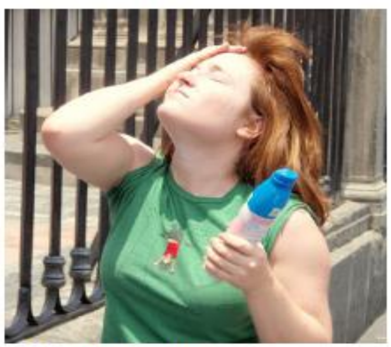
Deep Learning in Image Processing

Large-Scale Visual Recognition Challenge

(1000 classes, 1.2M training images, 150K testing images)

Siberian husky

Eskimo dog



GT: sunscreen
1: hair spray
2: ice lolly
3: sunscreen
4: water bottle
5: lotion

GT: flute
1: flute
2: oboe
3: panpipe
4: trombone
5: bassoon

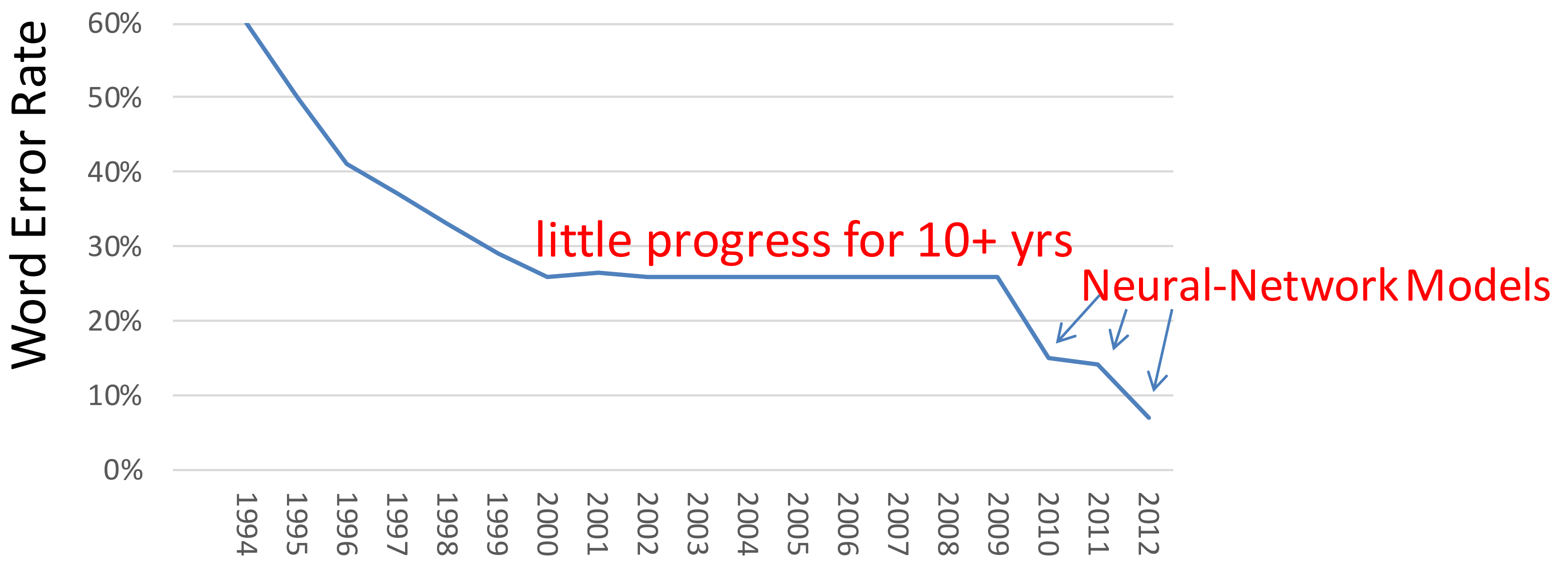
System	Year	Error
SIFT-based	2012	26.2%
SuperVision	2012	**16.4%**
Clarifai	2013	11.7%
GoogLeNet	2014	6.67%
Baidu	2015	5.98%
Microsoft	2015	**4.94%**
Google	2015	**4.90%**

Human: **5.10%**

Applications: automation for vehicles, surveillance or patrolling , image understanding, etc.

Deep Learning in Speech Recognition

Automatic Speech Recognition (speech-to-text)
(Switchboard data)



Brought ASR to more real-life use.

Applications: smart phones/watches, home appliances, cars, speech translation, etc.

Deep Learning in Text Processing

Translating texts from one language to another

System	Arabic-English	Chinese-English
OpenMT12 – 3rd Place	47.4	30.8
OpenMT12 – 2nd Place	47.5	32.2
OpenMT12 – 1st Place	49.5	32.6
BBN Neural Network Joint Model	52.8	34.7

[1] Evaluation matric: BLEU; larger is better

[2] NRC has implemented the BBN method

More recent work from Univ. of Montreal and Google.

Why Now?

- **Jürgen Schmidhuber**: It is a bit like the last neural network (NN) resurgence in the 1980s and early 1990s, but with million-times-faster computers. … Apparently, we will soon have the raw computational power of a human brain in a desktop machine. That is more than enough to solve many essential pattern recognition problems …

Recent technical advancement in Deep Learning:

See http://arxiv.org/abs/1404.7828 for a survey.

Who are Working on Deep Learning?

- Researchers and Engineers in both academia and industry:
  - Google(DeepMind), Microsoft, Facebook, Baidu, IBM (Watson), Universities…

Modeling the Meaning of Natural Languages

Two fundamental questions:

- How to represent the meaning of words?
- How to represent the meaning of sentences or larger spans of text?

Modeling the Meaning of Natural Languages

Two fundamental questions:

- How to represent the meaning of words?

- How to represent the meaning of sentences or larger spans of text?

love

Can a machine *fall in love*?

— *"The Emotion Machine" by* Marvin Minsky

Love:

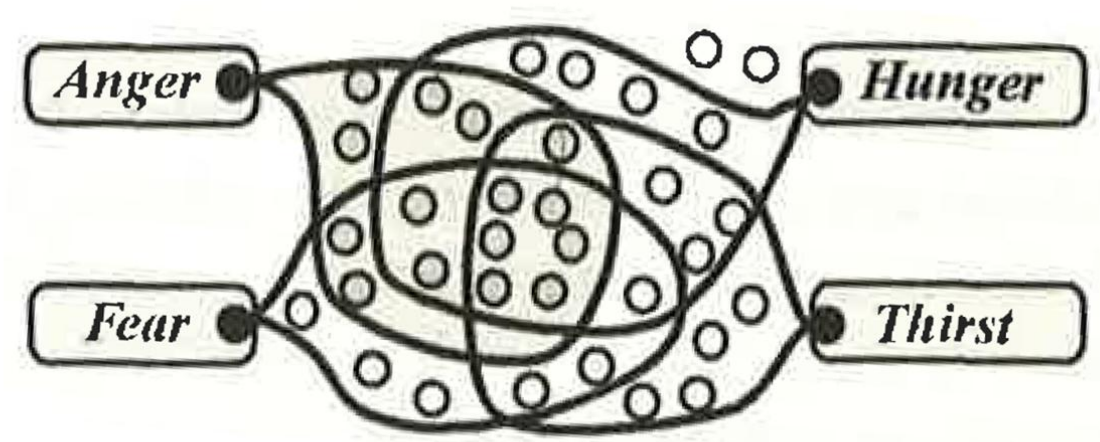
a (1) **:** strong affection for another arising out of kinship or personal ties <maternal *love* for a child> (2) **:** attraction based on sexual desire **:** affection and tenderness felt by lovers (3) **:** affection based on admiration, benevolence, or common interests <*love* for his old schoolmates>

… …

—*Merriam-Webster Dictionary*

Love, admiration, satisfaction …

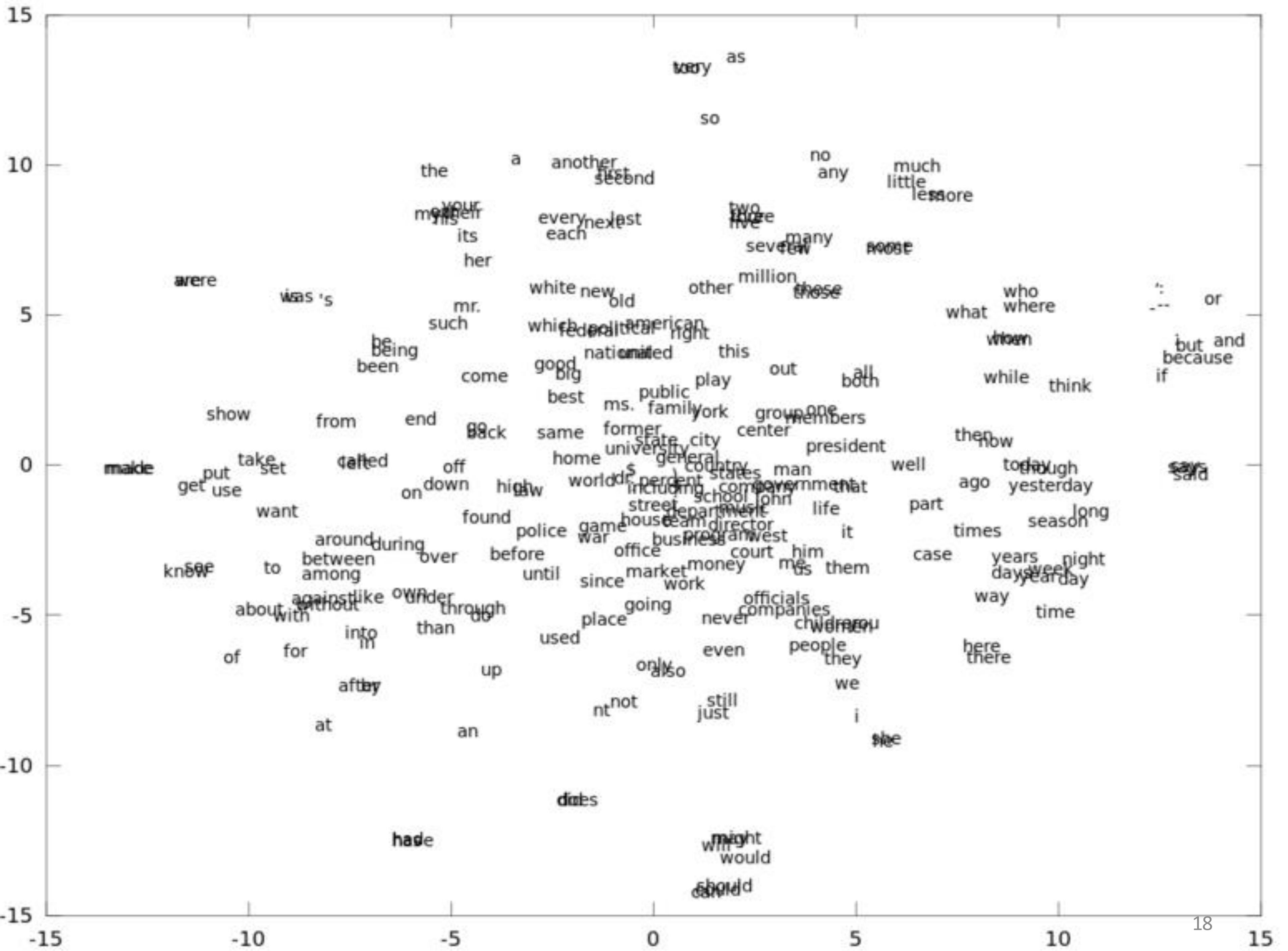
Anger, fear, hunger …



— *"The Emotion Machine" by* Marvin Minsky

- "You should know a word by the company it keeps" (Firth, 1957)

- “You should know a word by the company it keeps” (Firth, 1957)
  - Represent a word by its context (a window of surrounding words.)
    - You obtain a huge matrix.
  - Then dimension reduction is often performed, with different objectives.
    - PCA, LLE, SNE, Word2Vec, etc.

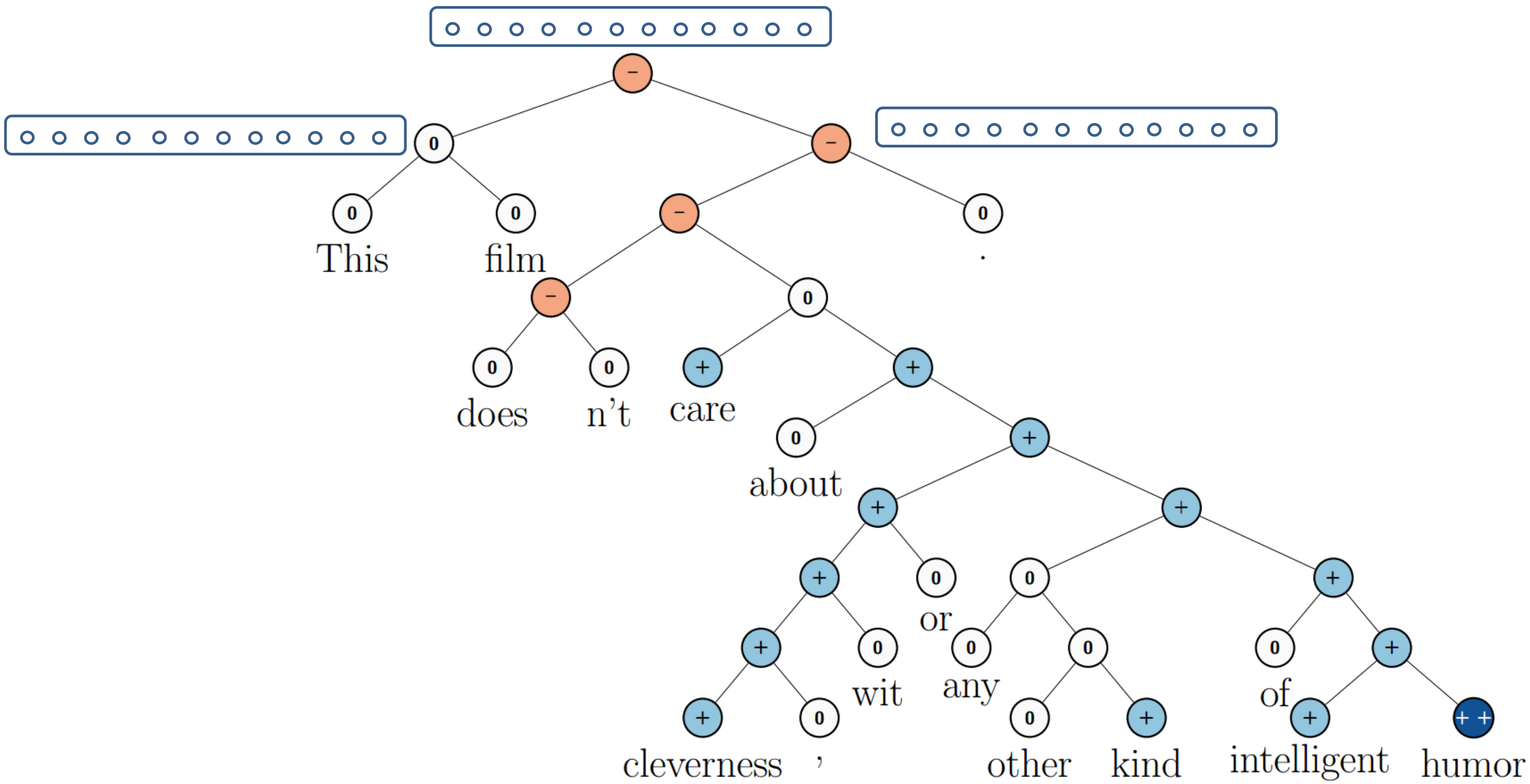


15
10
5
0
-5
-10
-15
-15
-10
-5
0
5
10
15
as
so
no
any
much
little
more
the
a
another
second
your
its
her
every
each
last
many
some
million
those
other
white
new
old
who
where
what
or
and
but
because
if
mr.
such
american
right
this
out
all
both
while
think
be
being
been
come
good
big
best
public
play
ms.
family
york
show
from
end
back
same
former
center
members
president
city
then
now
made
take
set
called
off
home
university
general
man
well
today
though
ago
yesterday
says
said
put
get
use
on
down
high
law
world
percent
including
states
government
that
school
john
part
want
found
street
house
life
season
long
police
game
war
director
business
west
it
times
around
during
between
over
before
office
court
him
case
years
night
know
see
to
among
until
since
market
money
me
us
them
week
day
way
about
with
against
like
own
under
through
do
place
going
officials
companies
never
children
women
time
into
in
than
used
people
even
they
here
there
of
for
up
only
also
we
after
by
not
nt
still
just
i
an
at
she
he
does
did
had
have
might
will
would
should
could

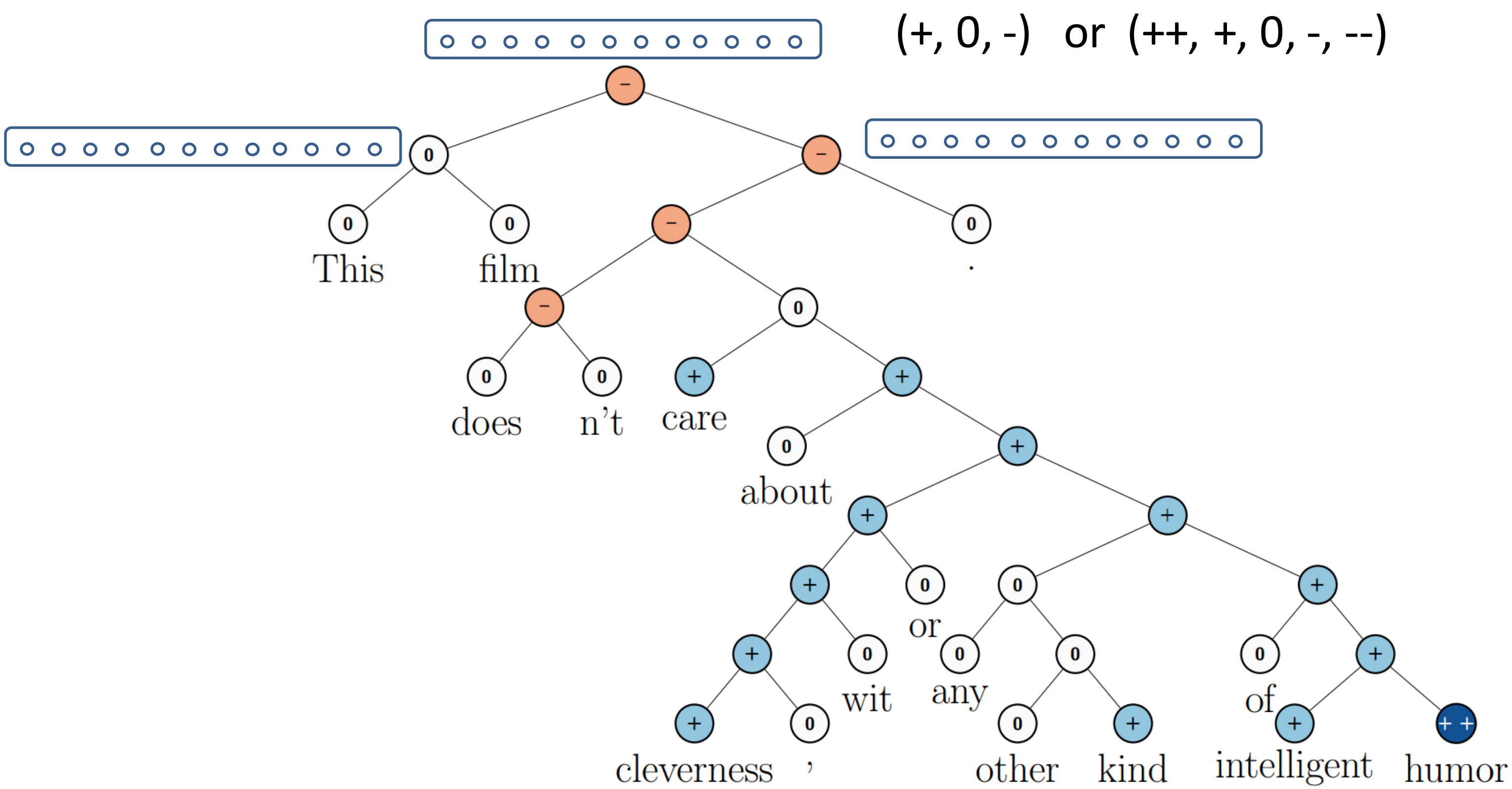
How to model the meaning of natural languages

Two basic questions

- How to represent the meaning of words?
- How to represent the meaning of sentences or larger spans of text?



Semantic Composition with Distributed Representation
(An example from [Socher et al. '13])



Semantic Composition with Distributed Representation
(An example from [Socher et al. '13])

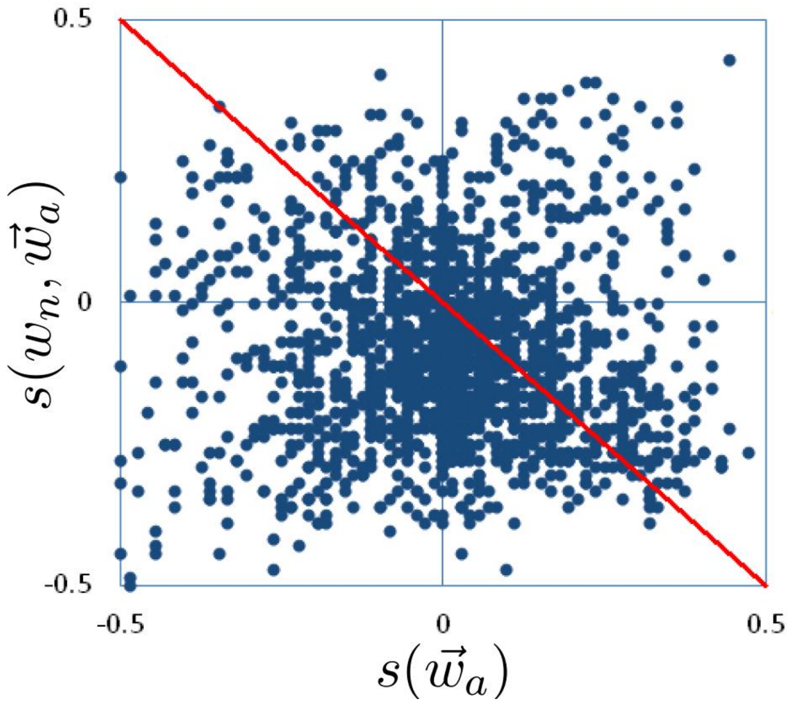
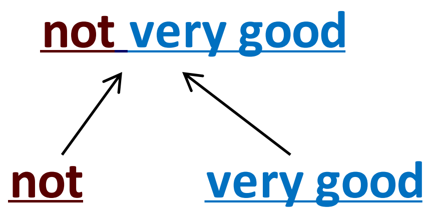


Figure 1. A dot in the figure corresponds to a negated phrase (e.g., *not very good*). The y-axis is its sentiment value and x-axis the sentiment of its argument (e.g., *very good*).

(Zhu al et. ACL-2014)

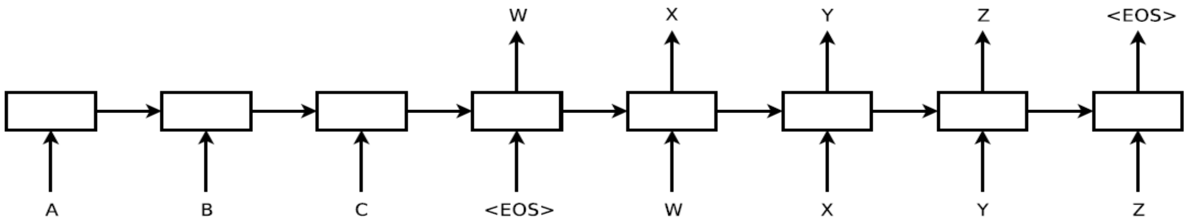
- Even one-layer composition can be a pretty complicated mapping/function.



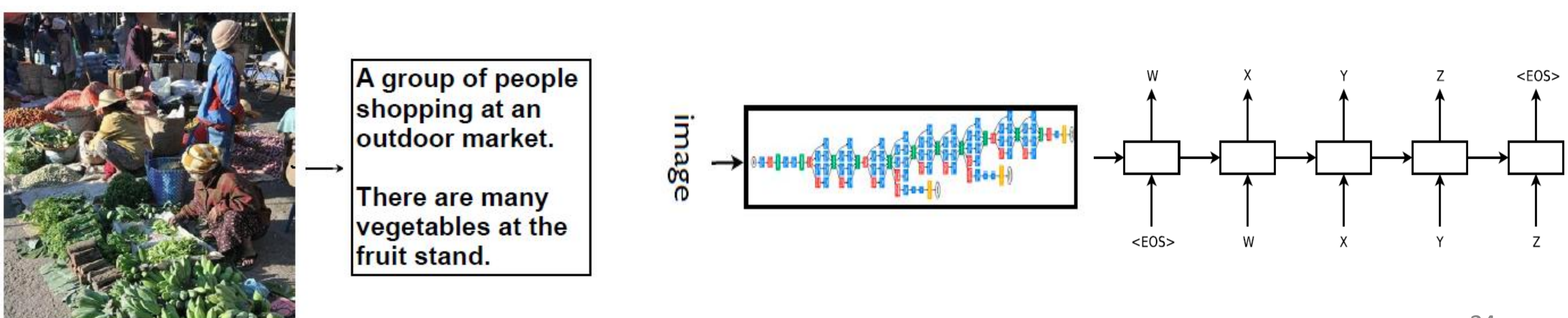
Case Study I: Using Long-Short Term Memory (LSTM) to Model Meaning (Semantics)

Long Short-Term Memory (LSTM)

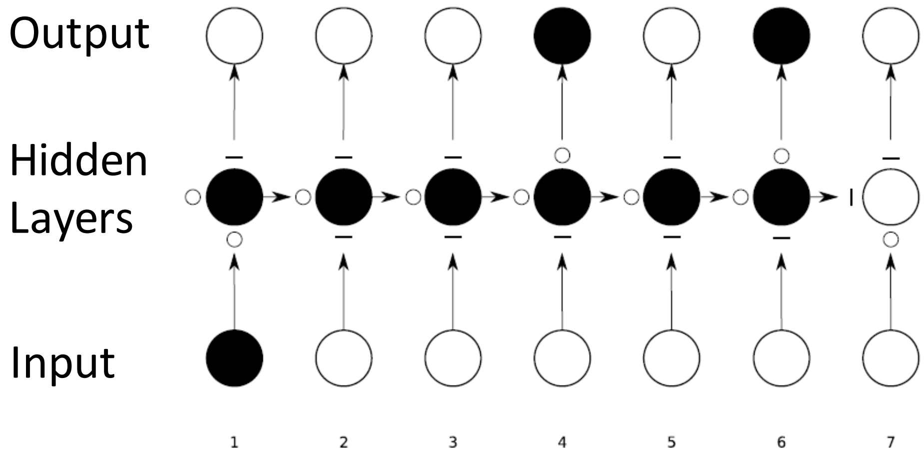
- LSTM [Hochreiter, '97] has showed to be effective in a wide range of problems.
  - Machine translation [Sutskever, '14; Cho, '14]



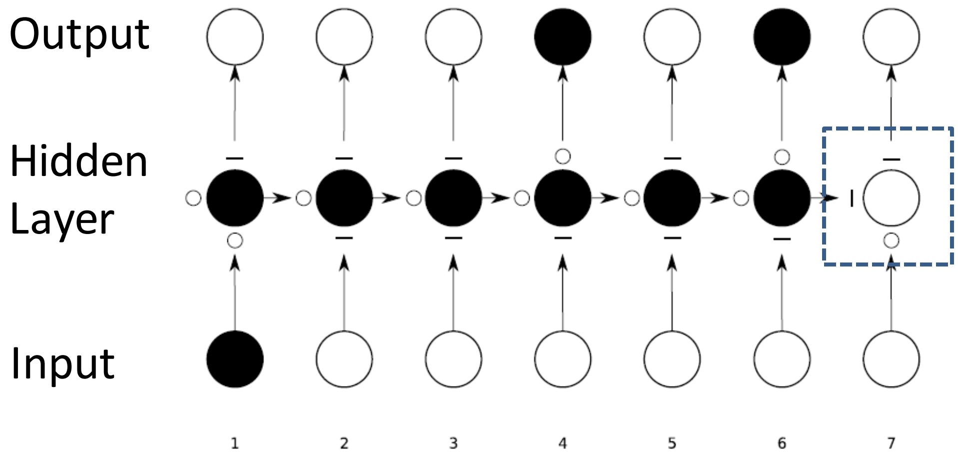
  - Image-to-text conversion [Vinyals, '14]

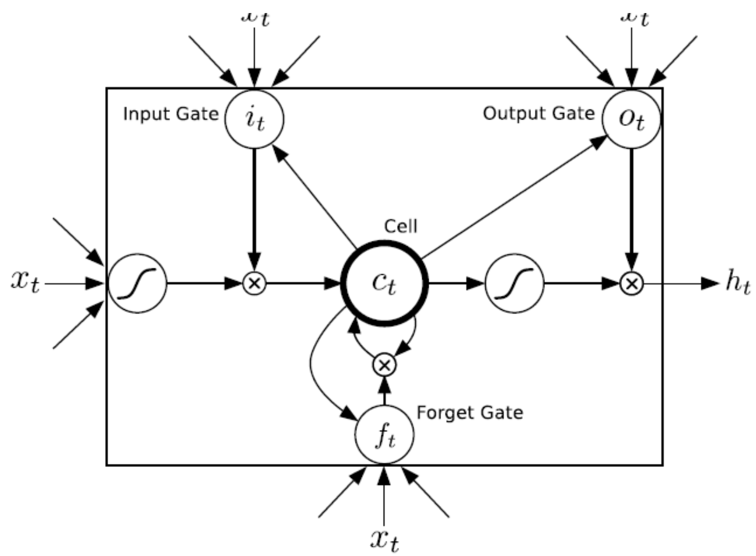


Linear-Chain LSTM

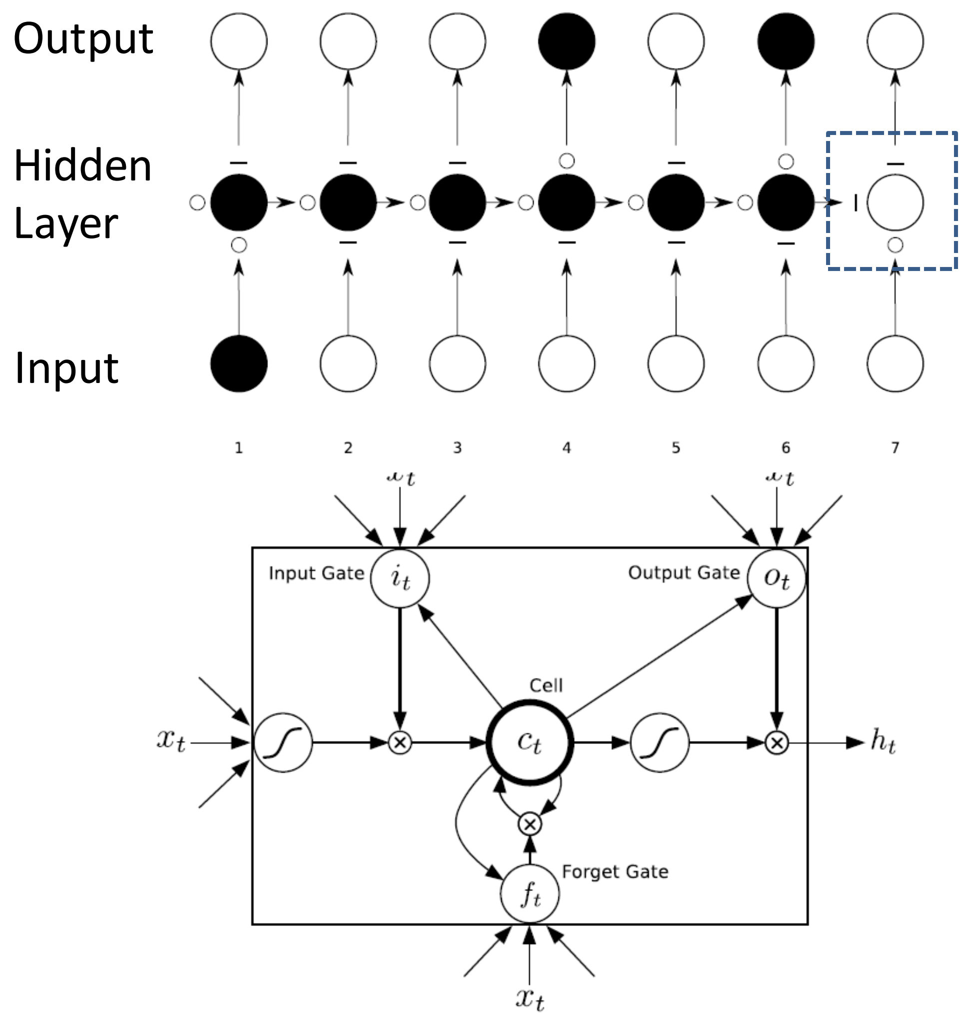


Linear-Chain LSTM





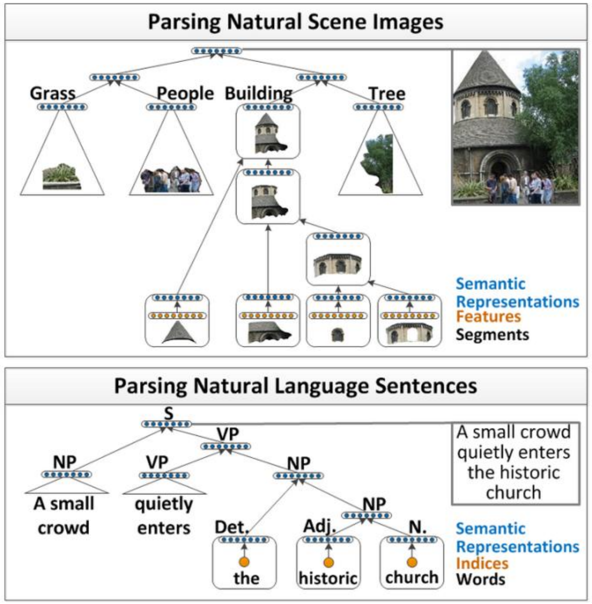
Linear-Chain LSTM

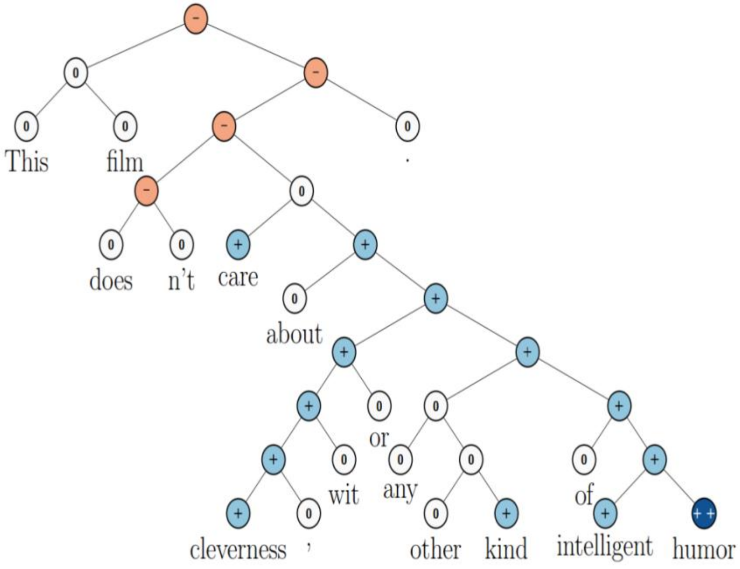


The model can remember pretty long history.

Recursive LSTM

- Recursion and the structures it forms are common in different modalities, e.g., trees [Socher, ’12; ’13].



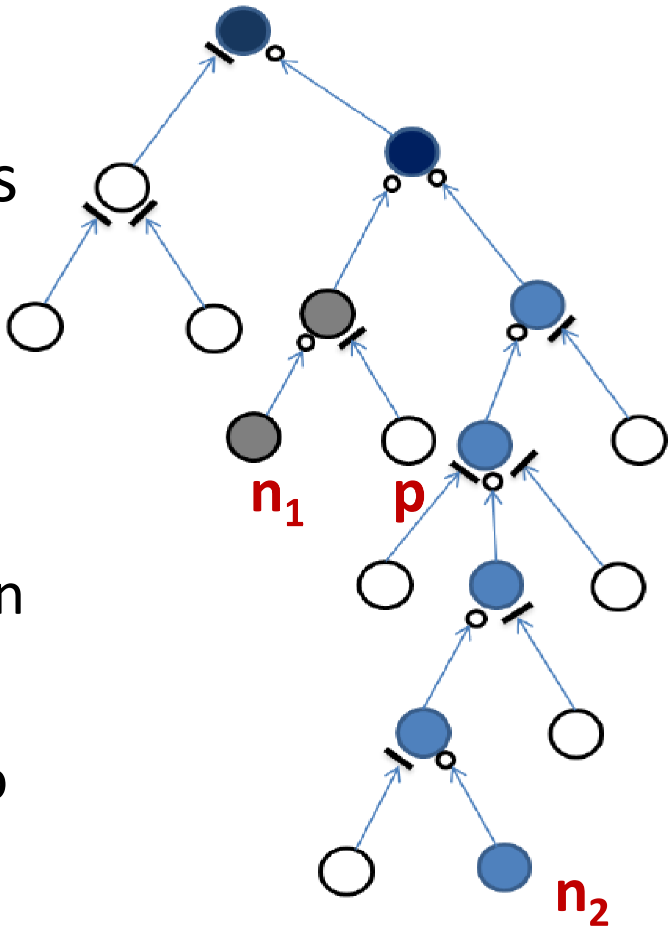


- While linear-chain LSTM can be used to model such problems, **we take a different view point**.

Recursive LSTM

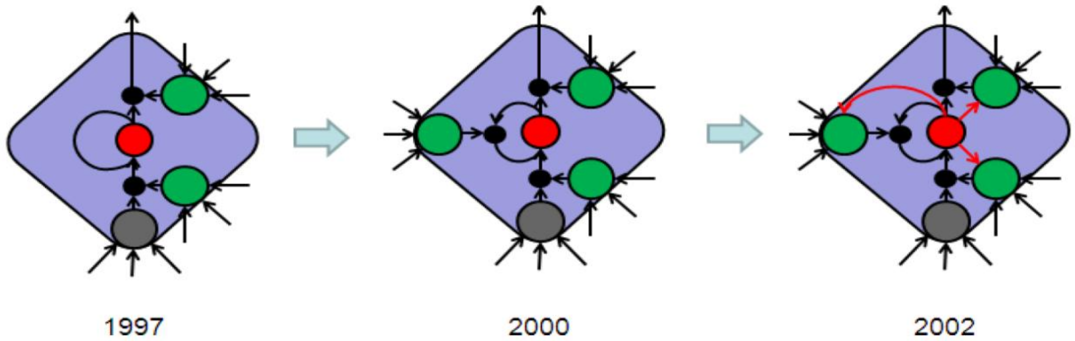
- **We propose a recursive LSTM (tree here).**

- We aim to explore a good way to consider structures (e.g., invariants and long-distance interplays over the structures).
  - E.g., the distance/relationship between n_1 and n_2 are invariant if node p varies (e.g., as a node of noun or a subtree of a longer phrase).
  - Such a model is interesting to us also because it recursively summarizes history over structure constituents.

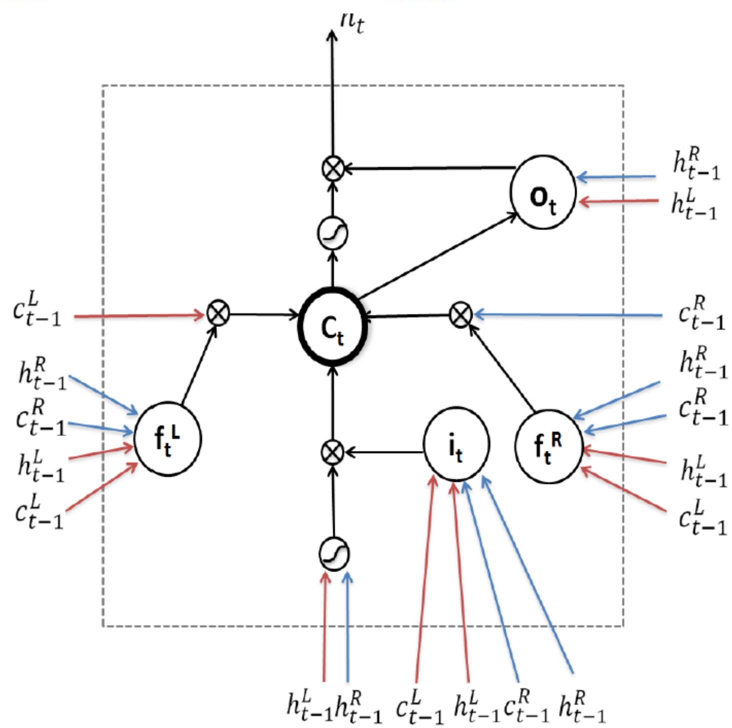


The Memory Blocks

LSTM

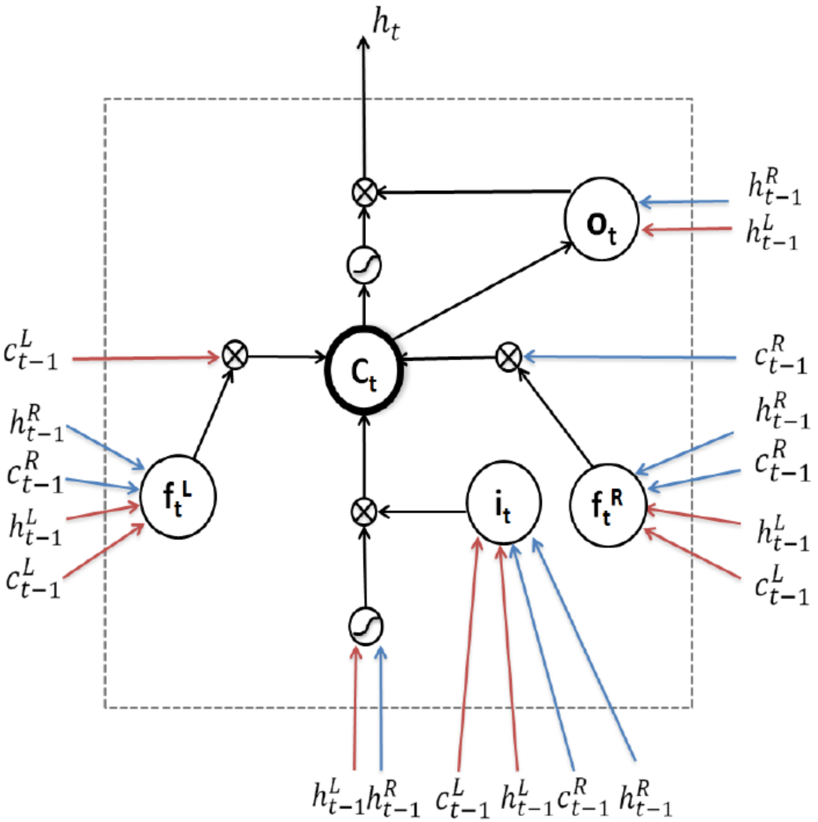


S-LSTM
(Our model)



Xiaodan Zhu, Parinaz Sobhani, Hongyu Guo. 2015. Long Short-Term Memory over Recursive Structures, Proceedings of International Conference on Machine Learning (**ICML**). Lille, France.

S-LSTM: Forward Propagation



$$\begin{aligned} i_t = \sigma(& W^L_{hi} h^L_{t-1} + W^R_{hi} h^R_{t-1} + W^L_{ci} c^L_{t-1} \\ &+ W^R_{ci} c^R_{t-1} + b_i) \end{aligned} \tag{1}$$

$$\begin{aligned} f^L_t = \sigma(& W^L_{hf_l} h^L_{t-1} + W^R_{hf_l} h^R_{t-1} + W^L_{cf_l} c^L_{t-1} \\ &+ W^R_{cf_l} c^R_{t-1} + b_{f_l}) \end{aligned} \tag{2}$$

$$\begin{aligned} f^R_t = \sigma(& W^L_{hf_r} h^L_{t-1} + W^R_{hf_r} h^R_{t-1} + W^L_{cf_r} c^L_{t-1} \\ &+ W^R_{cf_r} c^R_{t-1} + b_{f_r}) \end{aligned} \tag{3}$$

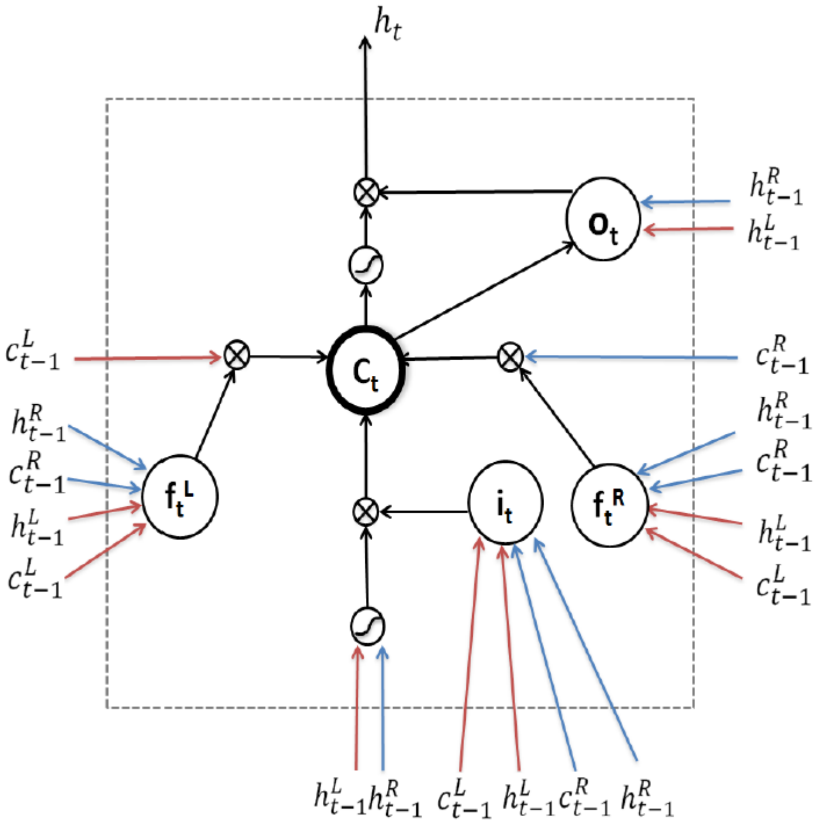
$$x_t = W^L_{hx} h^L_{t-1} + W^R_{hx} h^R_{t-1} + b_x \tag{4}$$

$$c_t = f^L_t c^L_{t-1} + f^R_t c^R_{t-1} + i_t tanh(x_t) \tag{5}$$

$$o_t = \sigma(W^L_{ho} h^L_{t-1} + W^R_{ho} h^R_{t-1} + W_{co} c_t + b_o) \tag{6}$$

$$h_t = o_t tanh(c_t) \tag{7}$$

S-LSTM: Backpropagation



$$\epsilon_t^h = \frac{\partial O}{\partial h_t} \tag{8}$$

$$\delta_t^o = \epsilon_t^h \otimes \tanh(c_t) \otimes f'(o_t) \tag{9}$$

$$\delta_t^{f_l} = \epsilon_t^c \otimes c_{t-1}^L \otimes f'(f_t^L) \tag{10}$$

$$\delta_t^{f_r} = \epsilon_t^c \otimes c_{t-1}^R \otimes f'(f_t^R) \tag{11}$$

$$\delta_t^i = \epsilon_t^c \otimes \tanh(x_t) \otimes f'(i_t) \tag{12}$$

Left child:

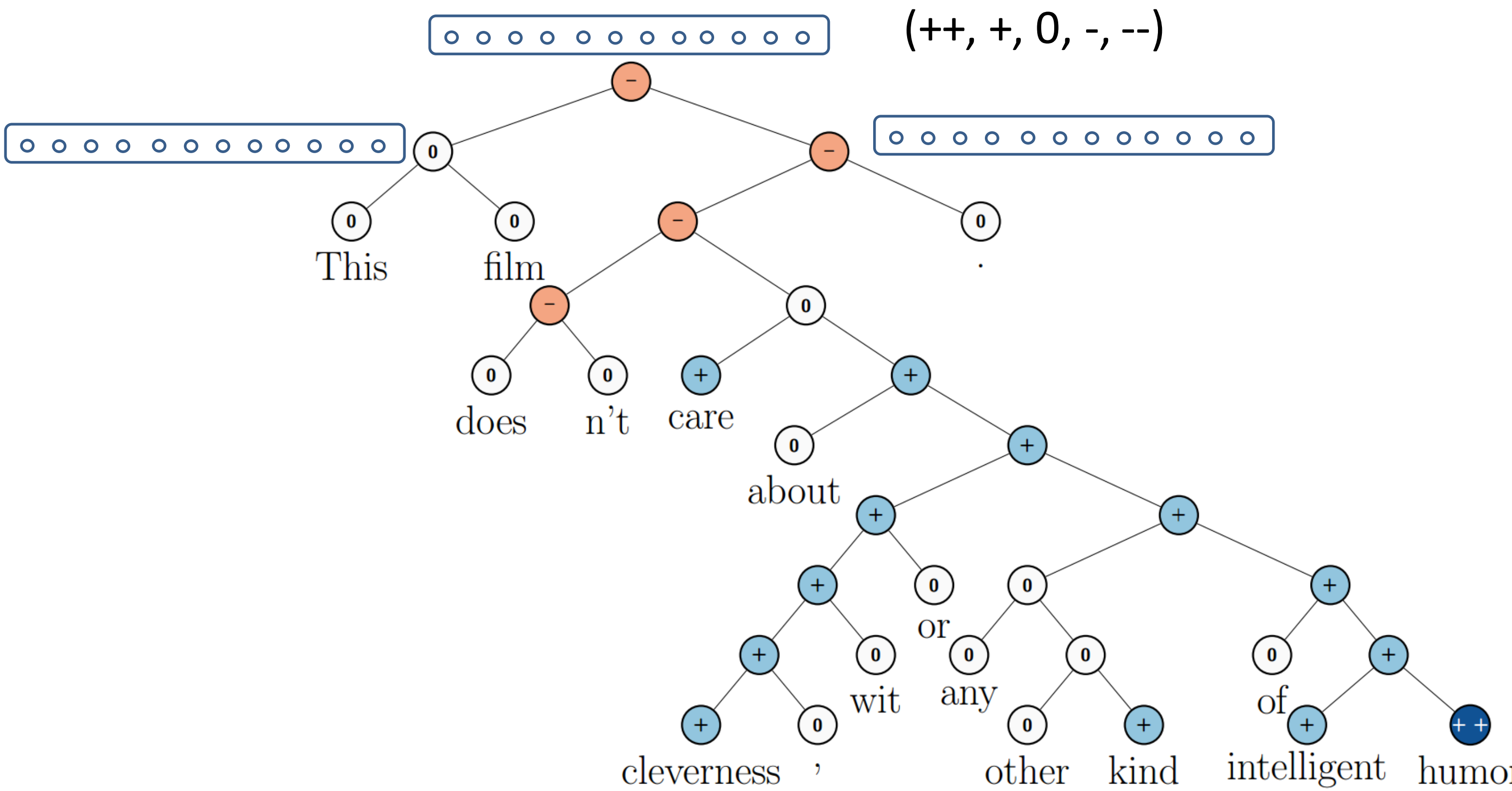
$$\begin{aligned} \epsilon_t^c =& \epsilon_t^h \otimes o_t \otimes g'(tanh(c_t)) + \epsilon_{t+1}^c \otimes f_{t+1}^L + \\ & (W_{ci})^T \delta_{t+1}^i + (W_{cf_l}^L)^T \delta_{t+1}^{f_l} + \\ & (W_{cf_r}^L)^T \delta_{t+1}^{f_r} + (W_{co})^T \delta_t^o \end{aligned} \tag{13}$$

Right child:

$$\begin{aligned} \epsilon_t^c =& \epsilon_t^h \otimes o_t \otimes g'(tanh(c_t)) + \epsilon_{t+1}^c \otimes f_{t+1}^R + \\ & (W_{ci})^T \delta_{t+1}^i + (W_{cf_l}^R)^T \delta_{t+1}^{f_l} + \\ & (W_{cf_r}^R)^T \delta_{t+1}^{f_r} + (W_{co})^T \delta_t^o \end{aligned} \tag{14}$$

Handling non-binary trees?

Experiments
(Sentiment analysis)

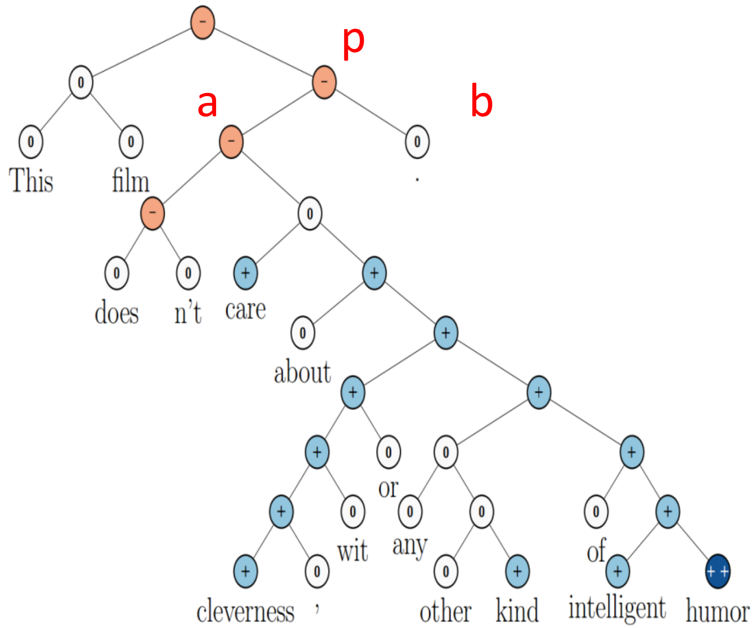


Semantics/sentiment composition

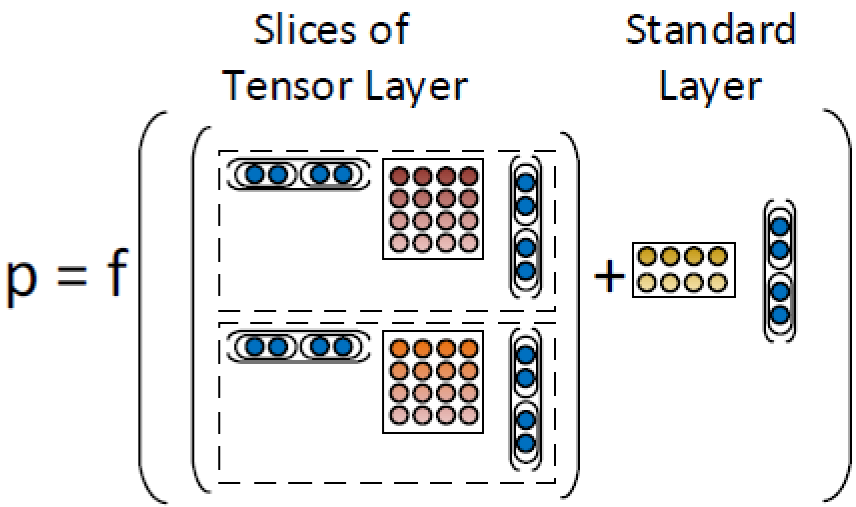
Experiment Set-up

- Data: Stanford Sentiment Treebank
  - Movie reviews
    - # sentences: 8544/1101/2210 (training/dev./test)
    - # phrases: 318582/41447/82600
  - All phrases, including roots (sentences), are manually annotated with sentiment labels.
- Evaluation metric
  - Classification accuracy (5-category)

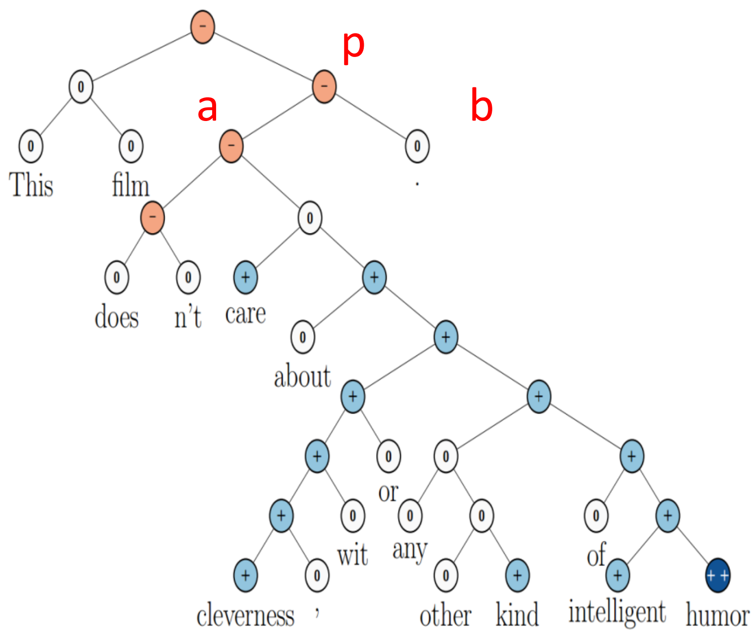
Recursive Neural Tensor Network (RNTN) [Socher et al., '13]



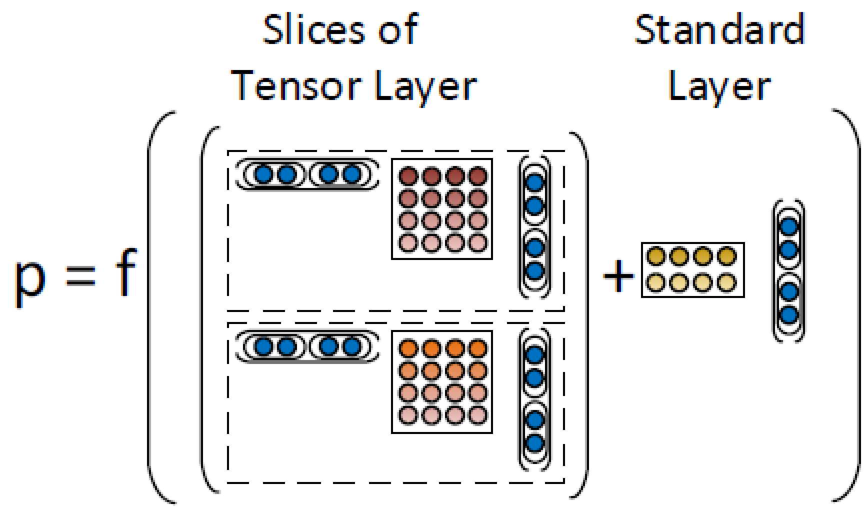
$$p = tanh(\begin{bmatrix} a \\ b \end{bmatrix}^T V^{[1:d]} \begin{bmatrix} a \\ b \end{bmatrix} + W \begin{bmatrix} a \\ b \end{bmatrix})$$

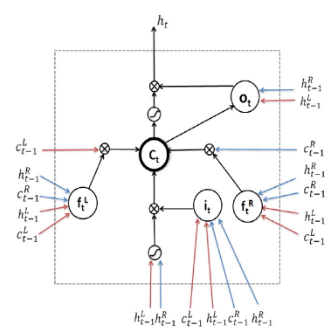


Recursive Neural Tensor Network (RNTN) [Socher et al., ’13]



$$p = tanh(\begin{bmatrix} a \\ b \end{bmatrix}^T V^{[1:d]} \begin{bmatrix} a \\ b \end{bmatrix} + W \begin{bmatrix} a \\ b \end{bmatrix})$$



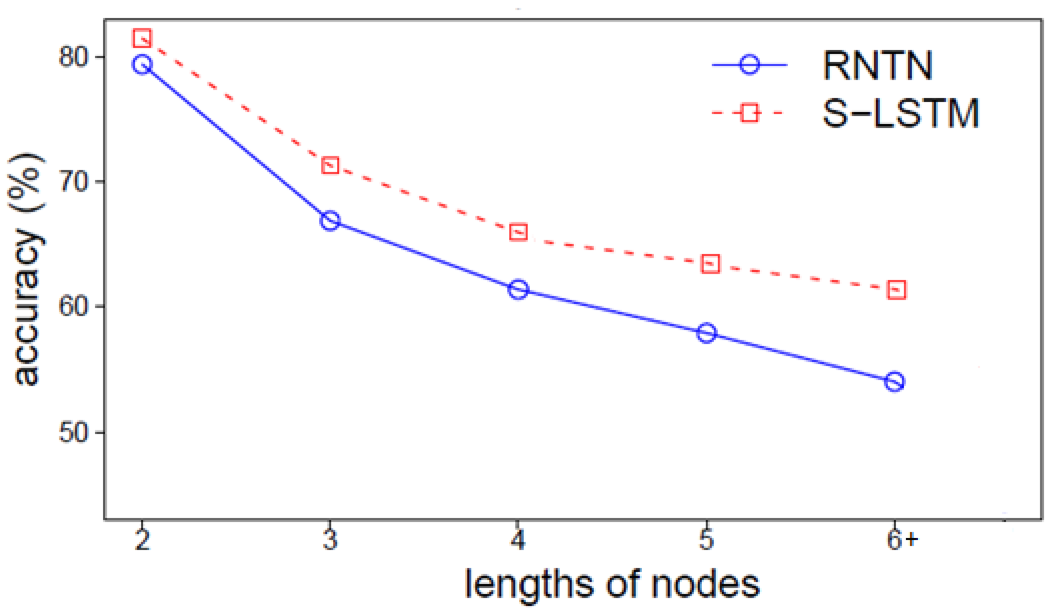


Results (Default setting)

Performances (accuracies) of different models on the test set of Stanford Sentiment Treebank, at the sentence level (roots) and the phrase level. † shows the performance are statistically significantly better ($p < 0.05$) than the corresponding models.

MODELS	ROOTS	PHRASES
NB	41.0	67.2
SVM	40.7	64.3
RVNN	43.2	79.0
RNTN	45.7	80.7
S-LSTM	**48.9**†	**81.9**†

Performances on Phrases of Different Lengths

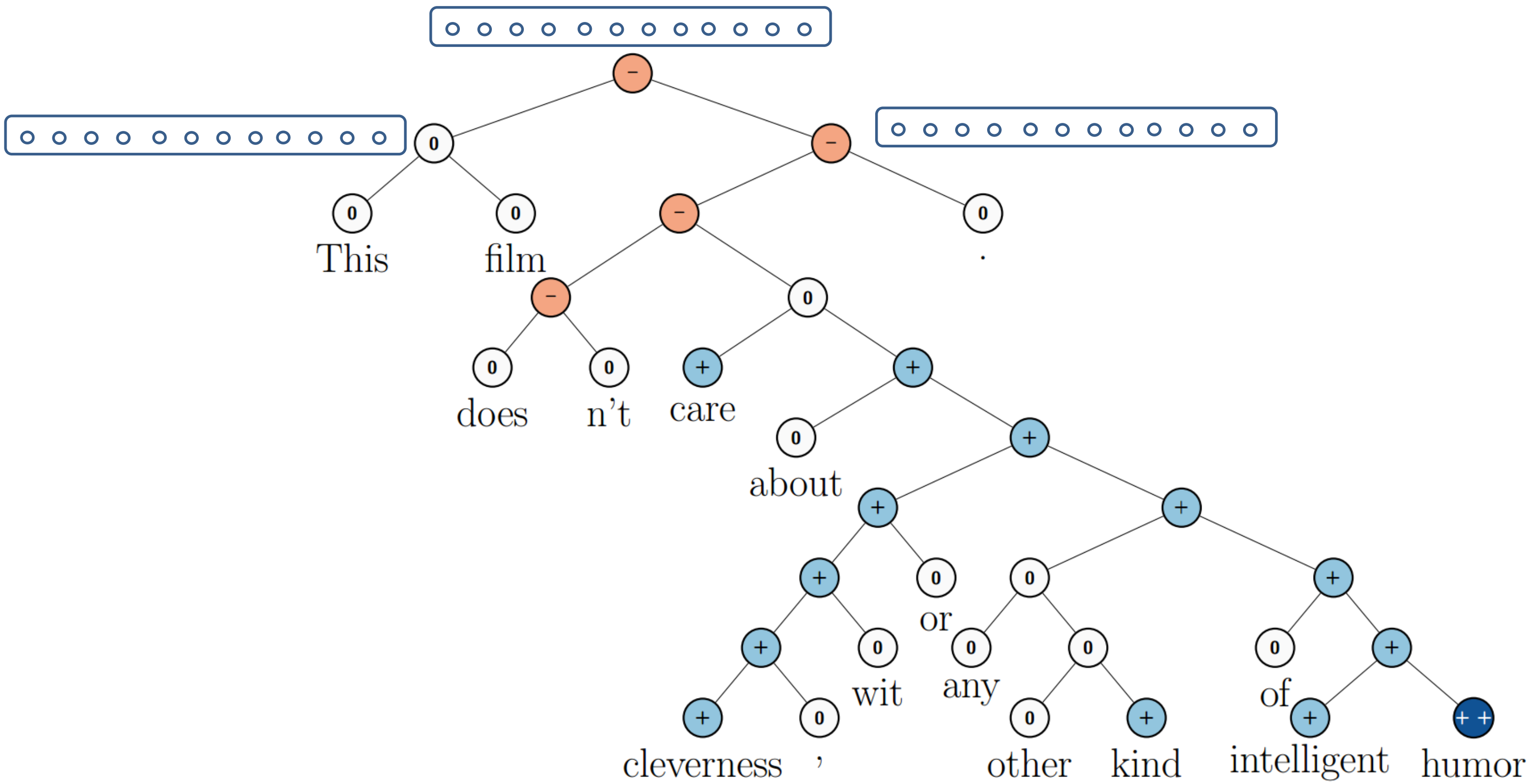


Accuracy on nodes(phrases) of different lengths

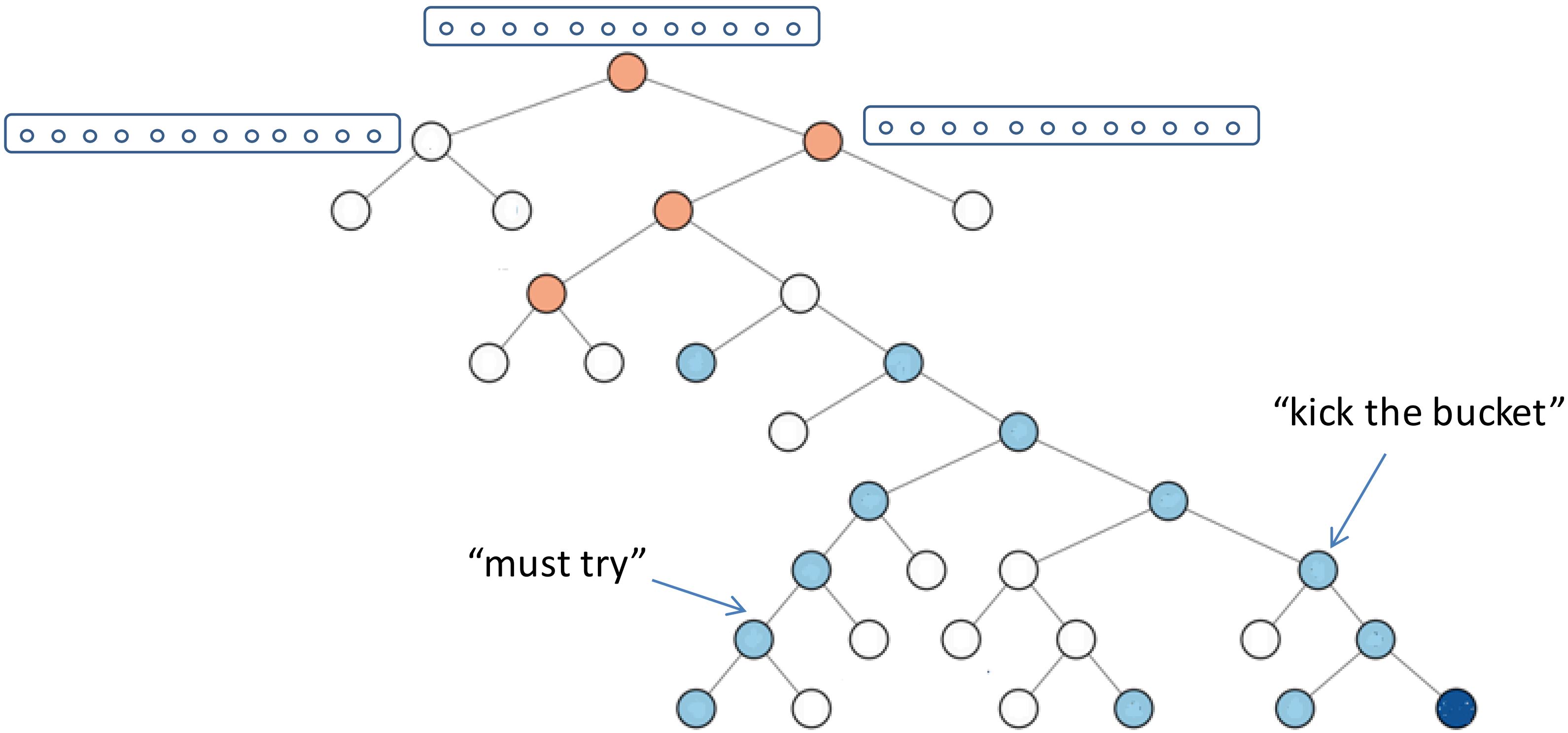
Structures vs. no Structures

Performances of models that do not use the given sentence structures. S-LSTM-LR is a degenerated version of S-LSTM that reads input words from left to right, and S-LSTM-RL reads words from right to left.

MODELS	ROOTS
S-LSTM-LR	40.2
S-LSTM-RL	40.3
S-LSTM	43.5†

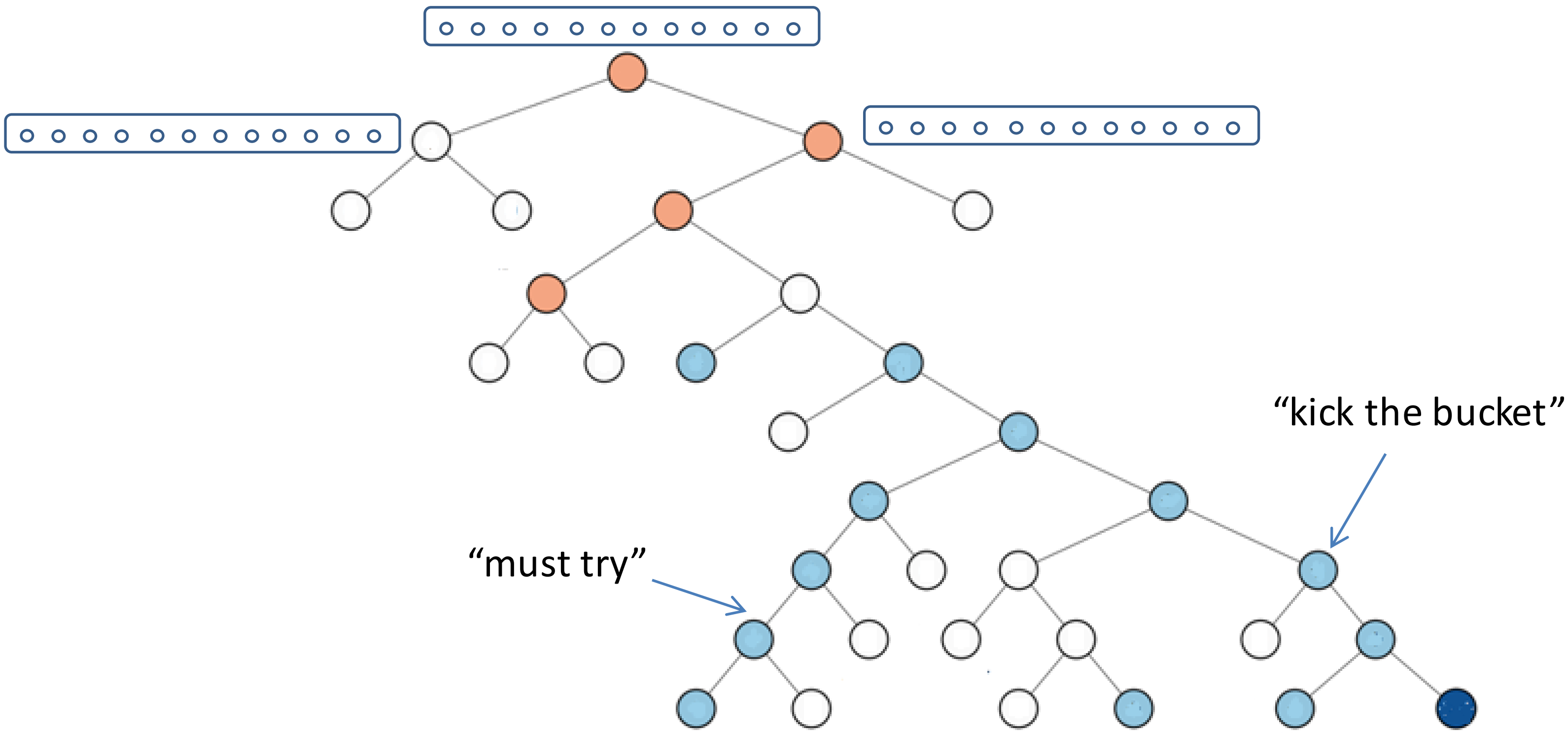


Semantic Composition with Distributed Representation



Semantic Composition with Distributed Representation

Case Study II: Networks for Integrating Compositional and Non-compositional Meaning

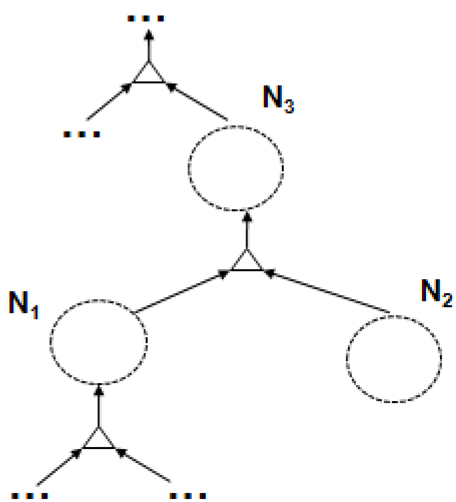


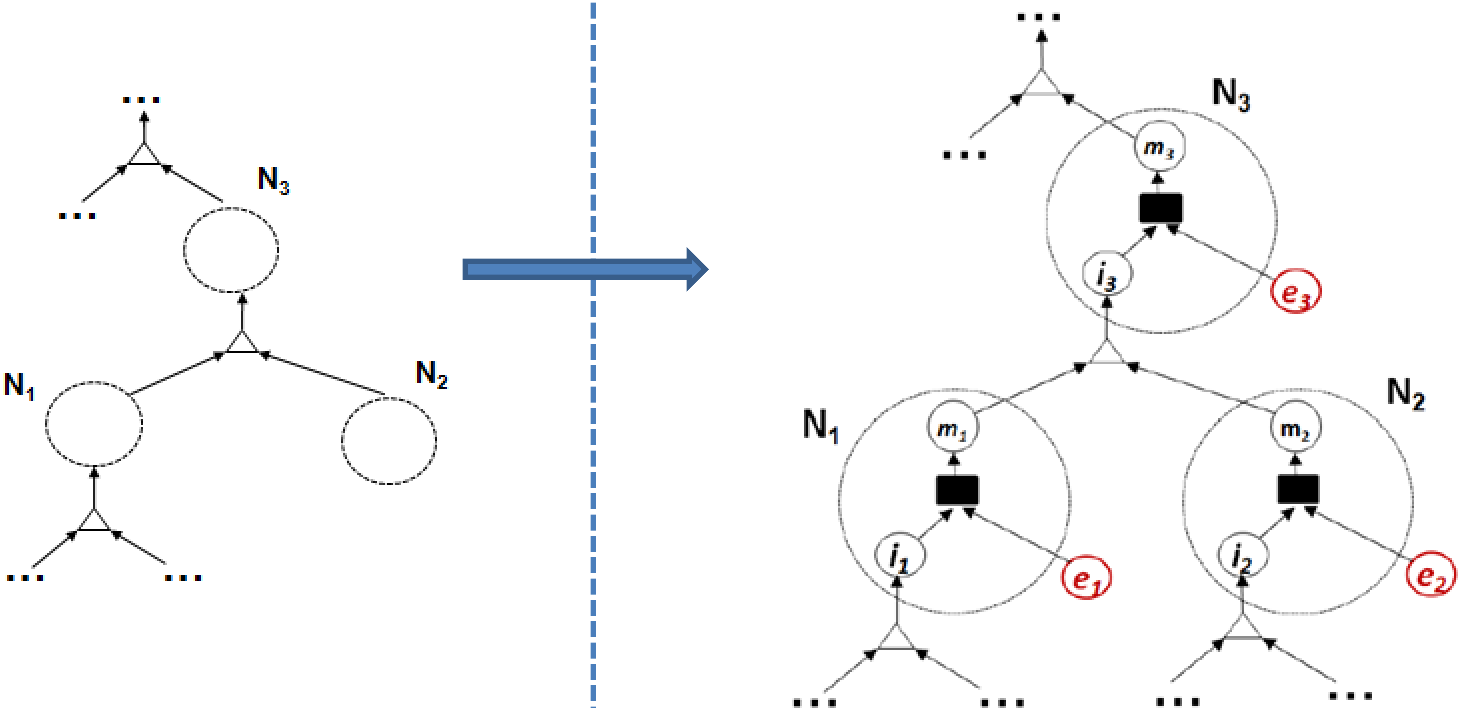
Semantic Composition with Distributed Representation

- A framework that is able to consider both compositionality/non-compositionality is of theoretical interest.

- A pragmatic viewpoint:
  - If one is able to obtain the sentiment/semantics of a text span holistically (e.g., for “must try”), it would be desirable that a composition model has the ability to decide the sources of knowledge it will use, ***softly***.

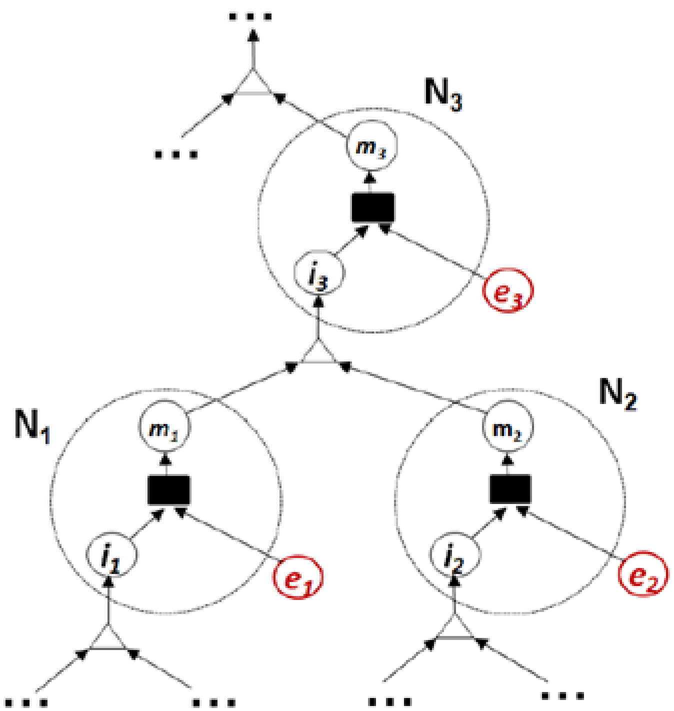
- Integrating compositional and non-compositional sentiment in the process of sentiment composition.
- Idea: Enabling individual composition operations to possess the capability of choosing and merging information from different resources locally, to optimize a global objective.


...
N3
...
N1
N2
...
...



A framework for considering compositionality and non-compositionality in composition.

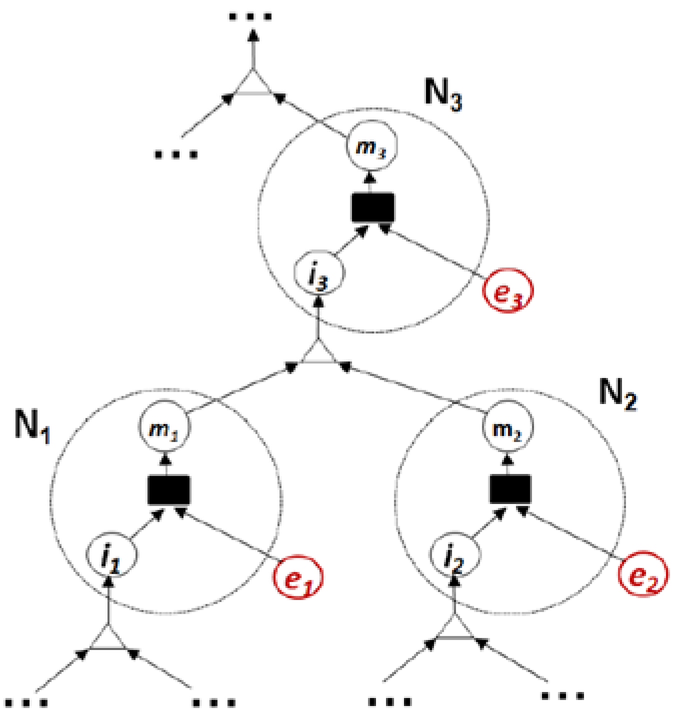
Model 1: Regular bilinear merging



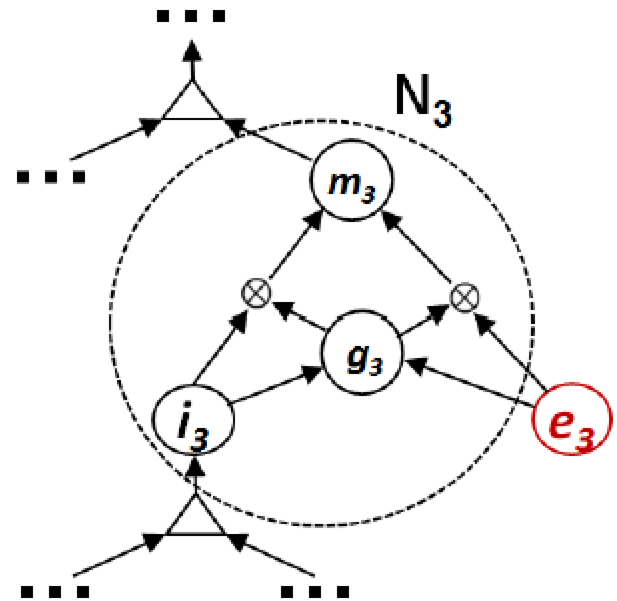
A framework for considering compositionality and non-compositionality in composition.

$$m_3 = tanh(W_m \begin{bmatrix} i_3 \\ e_3 \end{bmatrix} + b_m)$$

Model 2: Explicitly gated merging



A framework for considering compositionality and non-compositionality in composition.



$$g_3 = \sigma(\begin{bmatrix} W_{g_e} e_3 \\ \\ W_{g_i} i_3 \end{bmatrix} + b_g)$$

$$m_3 = tanh(W_m(g_3 \otimes \begin{bmatrix} i_3 \\ e_3 \end{bmatrix}) + b_m)$$

Model 3: Confined-tensor-based merging

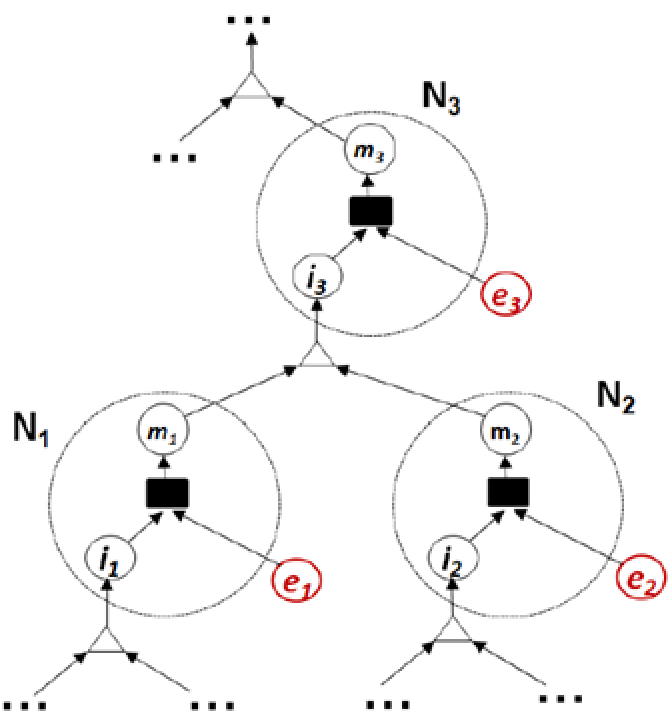
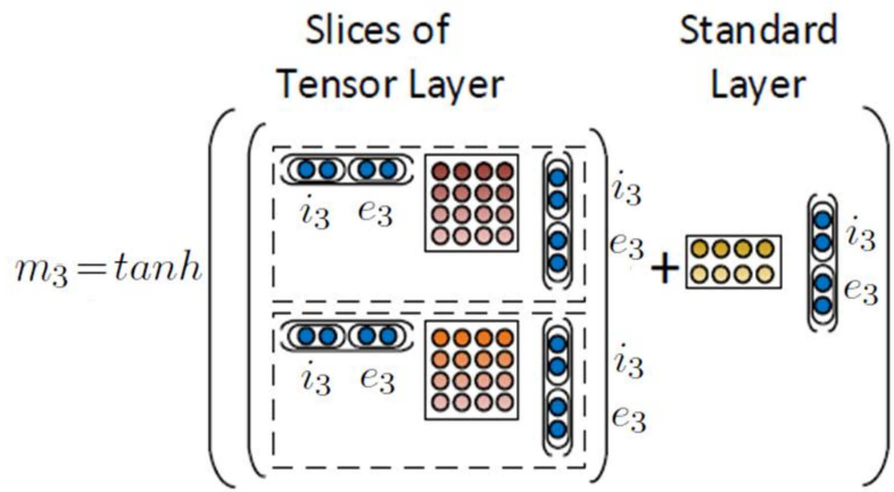


Figure 1: A prior-enriched semantic network (PESN) for sentiment composition.



$$m_3 = tanh(\begin{bmatrix} i_3 \\ e_3 \end{bmatrix}^T V_m^{[1:d]} \begin{bmatrix} i_3 \\ e_3 \end{bmatrix} + W_m \begin{bmatrix} i_3 \\ e_3 \end{bmatrix})$$

Model 3: Confined-tensor-based merging

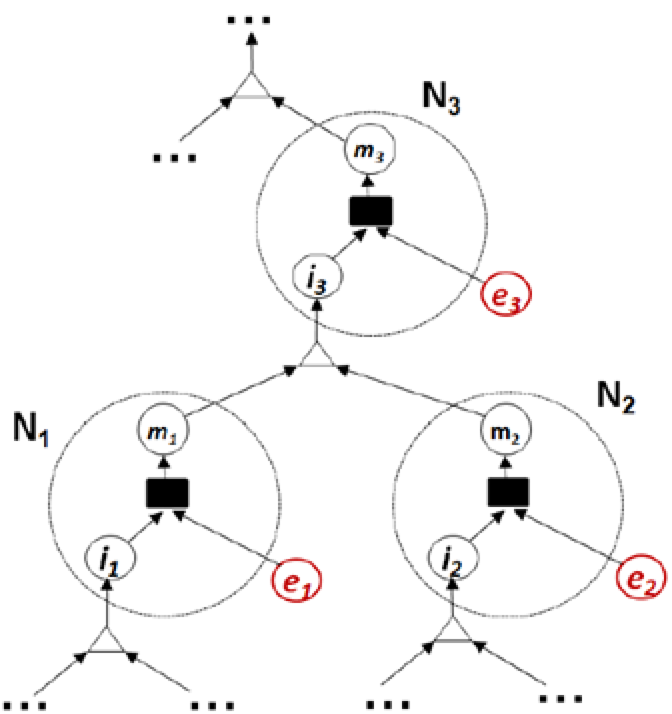
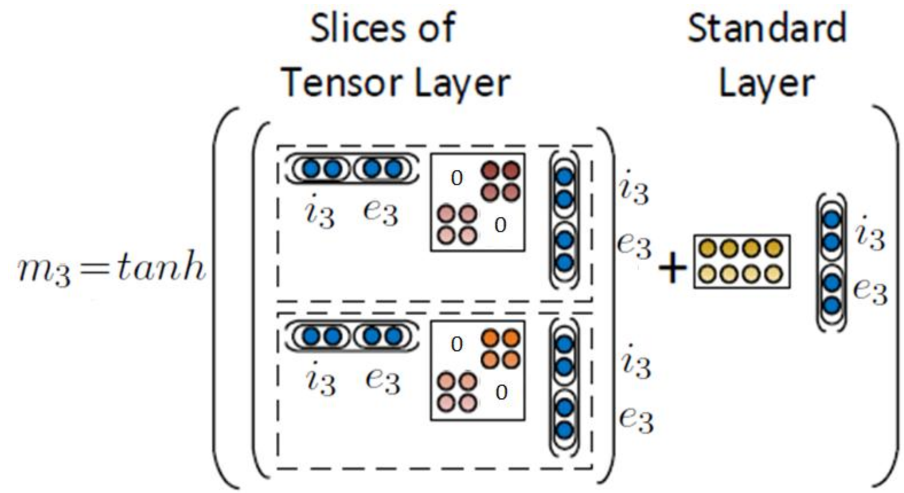


Figure 1: A prior-enriched semantic network (PESN) for sentiment composition.



$$m_3 = tanh(\begin{bmatrix} i_3 \\ e_3 \end{bmatrix}^T V_m^{[1:d]} \begin{bmatrix} i_3 \\ e_3 \end{bmatrix} + W_m \begin{bmatrix} i_3 \\ e_3 \end{bmatrix})$$

Experiment set-up

- Data: Stanford Sentiment Treebank
  - Movie reviews
    - # sentences: 8544/1101/2210 (training/dev./test)
    - # phrases: 318582/41447/82600
  - All phrases, including roots (sentences), are manually annotated with sentiment labels.
- Evaluation metric
  - Classification accuracy

Experiment set-up

- Non-compositional sentiment
  - Using the human annotation coming with Stanford Sentiment Treebank for bigrams and trigrams.
  - Sentiment of ngrams automatically learned from tweets (Mohammad et al., 2013b).
    - Polled the Twitter API every four hours from April to December 2012 in search of tweets with either a positive word hashtag or a negative word hashtag.
    - Using 78 seed hashtags (32 positive and 36 negative) such as #good, #excellent, and #terrible to annotate sentiment.
    - 775,000 tweets that contain at least a positive hashtag or a negative hashtag were used as the learning corpus.

Experiment set-up

- Pointwise mutual information (PMI) is calculated for each bigrams and trigrams.

$$score(w) = PMI(w, positive) - PMI(w, negative)$$

- Each sentiment score is converted to a *one-hot* vector; e.g. a bigram with a score of -1.5 will be assigned a 5-dimensional vector [0, 1, 0, 0, 0] (i.e., the ***e*** vector).

Results: prediction performance

Models	sentence-level (roots)	all phrases (all nodes)
(1) RNTN	42.44	79.95
(2) Regular-bilinear (auto)	42.37	79.97
(3) Regular-bilinear (manu)	42.98	80.14
(4) Explicitly-gated (auto)	42.58	80.06
(5) Explicitly-gated (manu)	43.21	80.21
(6) Confined-tensor (auto)	42.99	80.49
(7) Confined-tensor (manu)	**43.75**†	**80.66**†

Table 1: Model performances (accuracies) on predicting 5-category sentiment at the sentence (root) level and phrase level.

- The results is based on the version 3.3.0 of the Stanford CoreNLP.
- We trained the RNTN models with the default parameters and run the training from 5 different random initializations.
  java -mx8g edu.stanford.nlp.sentiment.SentimentTraining -numHid 25 -trainPath train.txt -devPath dev.txt -train –model model.ser.gz

Remarks

- **Deep Learning** is a set of machine learning algorithms that model high-level abstractions in data by using model architectures (often ***neural networks***).
- It has significantly improved the states of the art on many problems in many fields.
  - **Natural language processing**
  - Speech recognition
  - Image/video processing

Remarks

Two fundamental questions:

- How to represent the meaning of words?

- How to represent the meaning of sentences or larger spans of text?

Remarks

- A recursive LSTM model to consider input structures in composition.
- Achieved the state-of-the-art performance on a semantic composition task.
- Explicitly modeling the structures is helpful.

Remarks

- We are also concerned with integrating compositionality and non-compositionality in the process of composition.
- We discuss how to enable each composition operation to be able to choose and merge information from these two types of sources locally, to optimize a global objective.
  - We showed moderate improvement over a baseline model that does not consider this.

Thank you!