Research discussion

Neural network background and our potential ideas

Governing a large country

is like cooking a small dish

治大国若烹小鲜

ancient Chinese philosopher, **Lao-tzu**, said

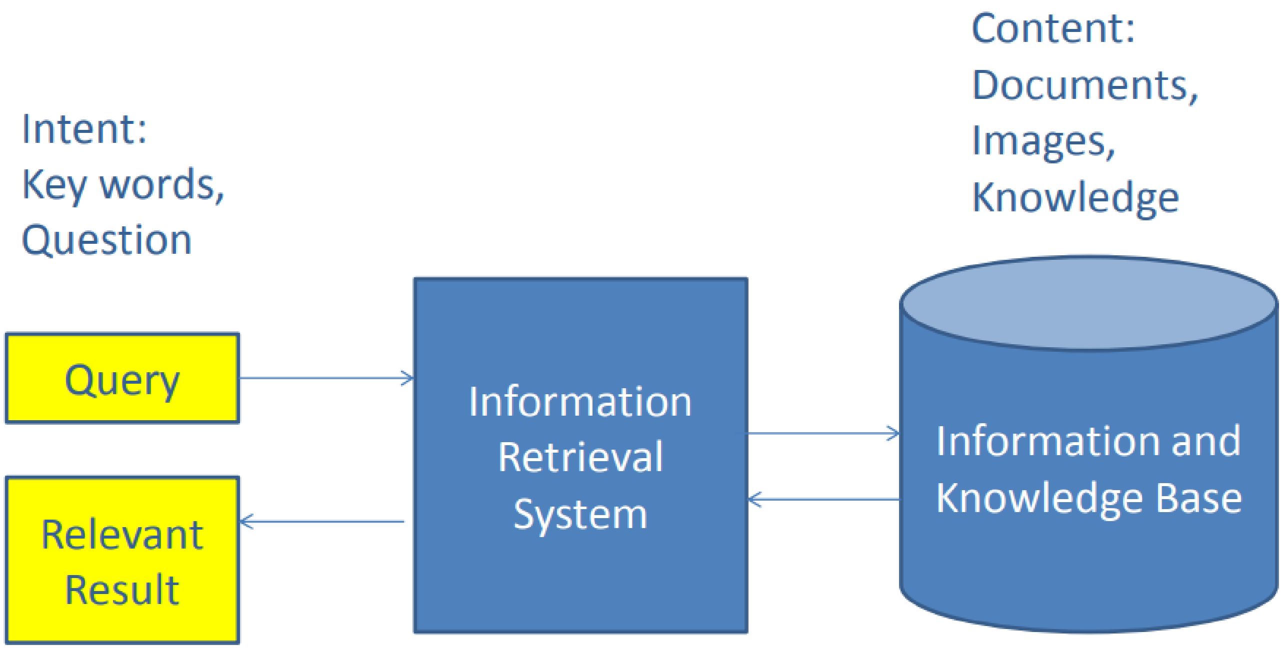
Contents

- General Cooking
  - Material : Word embedding
  - Methods: Neural network
- Quantum-style Cooking

Background of Neural IR

- **Trends of DL for IR**
- Word embedding
- Neural network
- DL for IR/NLP

IR background



Key Questions: How to Represent Intent and Content, How to Match Intent and Content

Traditional IR – Tfidf example

Document:

Star Wars: Episode VII
Three decades after the defeat of the Galactic Empire, a new threat arises.

Query:
star wars the force awakens reviews

$$q = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \xrightarrow{f(q,d)} d = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

$$f_{VSM}(q,d) = \frac{\langle q,d \rangle}{\| q \| \cdot \| d \|}$$

- Representing query and document as word vectors
- calculating cosine similarity between them

Modern IR – Learn to Rank

Query:
star wars the force awakens reviews

Document:

Star Wars: Episode VII
Three decades after the defeat of the Galactic Empire, a new threat arises.

$$
\begin{array}{lccc}
 & q & & d \\
\text{(star wars)} & v_{q1} & \vec{f}(q,d) & v_{d1} \\
\text{(the force awakens)} & \vdots & \longrightarrow & \vdots \\
\text{(reviews)} & & & \\
 & v_{qm} & & v_{dn}
\end{array}
$$

- Conducting query and document understanding
- Representing query and document as multiple feature vectors
- Calculating multiple matching scores between query and document
- Training ranker with matching scores as features using learning to rank

Features + Ranking

Features：
- Language model
- BM25
- Title/Snippet/Document
- Pagerank

Ranking:
- Point-wise
- Pair-wise
- List-wise

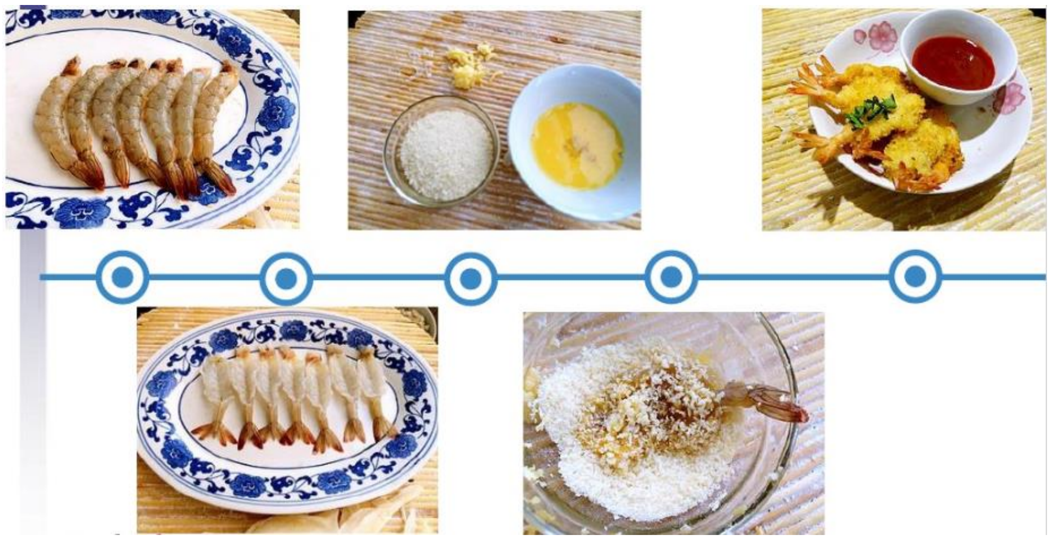
Credited to Prof. Songchun Zhu

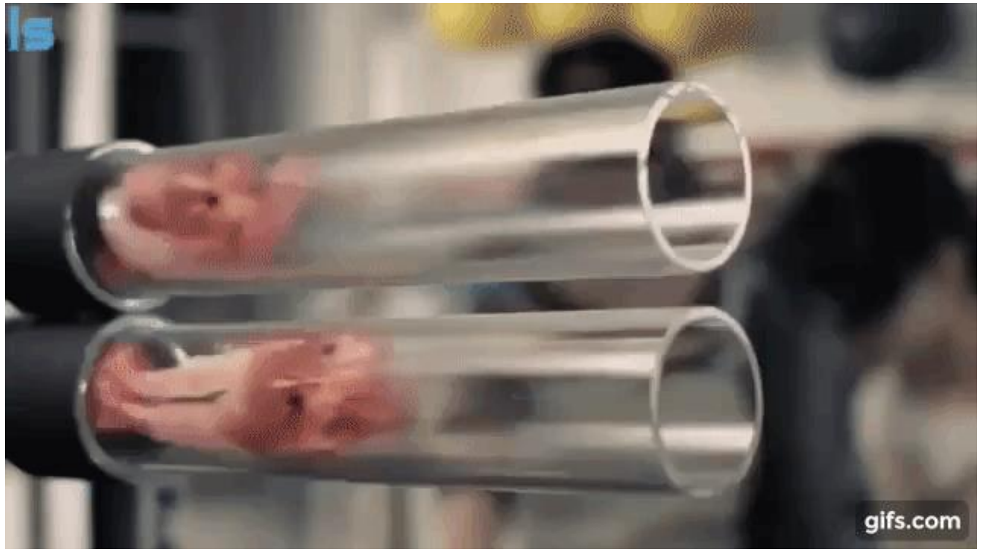
Example of **Mismatch**

Query	Document	Term Matching	Semantic Matching
seattle best hotel	seattle best hotels	no	yes
pool schedule	swimmingpool schedule	no	yes
natural logarithm transformation	logarithm transformation	partial	yes
china kong	china hong kong	partial	no
why are windows so expensive	why are macs so expensive	partial	no

Hang li, http://www.hangli-hl.com/uploads/3/4/4/6/34465961/tsinghua_opportunities_and_challenges_in_deep_learning_for_information_retrieval.pdf

End-to-end





https://www.youtube.com/watch?v=TYpBJ71VW9g

The inputting features are also **learnable/trainable**

Credited to Dr. Naiyan Wang

Trends for Neural IR

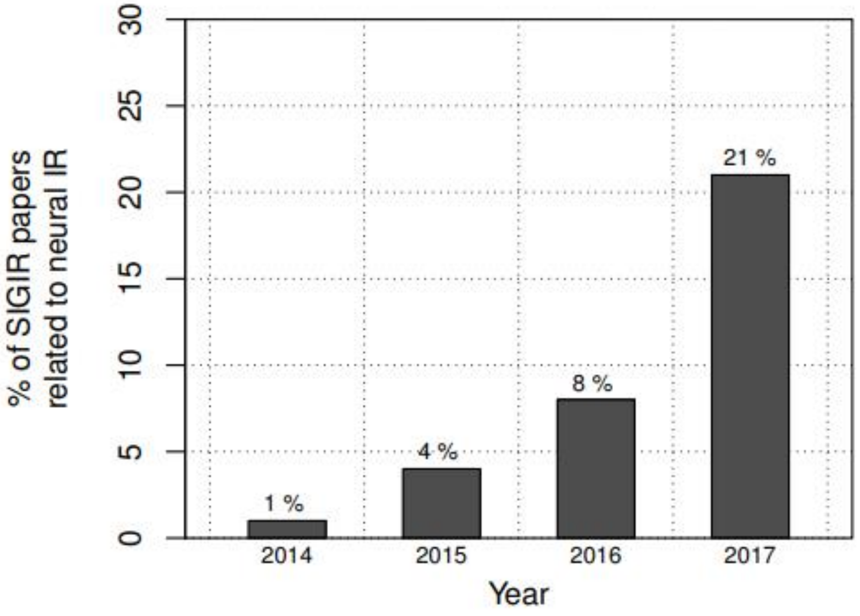


Figure 1: The percentage of neural IR papers at the ACM SIGIR conference—as determined by a manual inspection of the paper titles—shows a clear trend in the growing popularity of the field.

Mitra B, Craswell N. Neural Models for Information Retrieval[J]. arXiv preprint arXiv:1705.01509, 2017.

Background of Neural IR

- Trends of DL for IR
- Word embedding
- Neural network
- DL for IR/NLP

Localist representation

		Size color … unknown
• BMW	[1, 0, 0, 0, 0]	[.3, .7, .2, .1, .5]
• Audi	[0, 0, 0, 1, 0]	[.5, .3, .2, .1, .0]
• Benz	[0, 0, 1, 0, 0]	[.2, .0, .31, .03, .01]
• Polo	[0, 0, 0, 1, 0]	[.1, .1, .5, .5, 0.2]

http://www.cs.toronto.edu/~bonner/courses/2014s/csc321/lectures/lec5.pdf

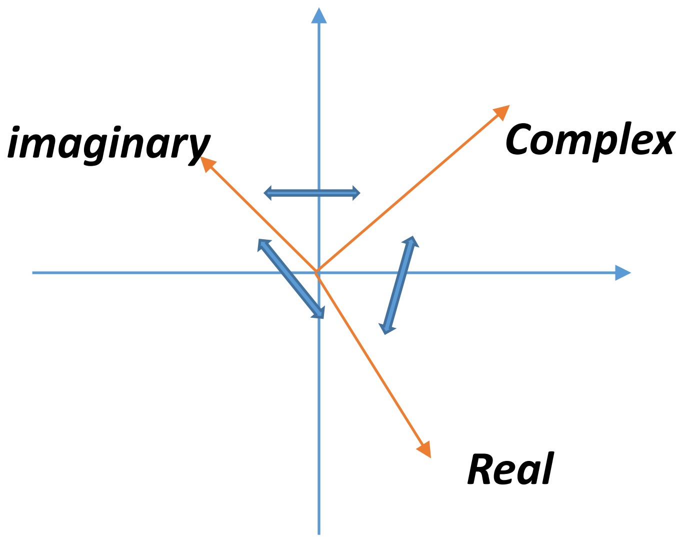
Distributed representation

Size color … unknown

- BMW [1, 0, 0, 0, 0] [.3, .7, .2, .1, .5]
- Audi [0, 0, 0, 1, 0] [.5, .3, .2, .1, .0]
- Benz [0, 0, 1, 0, 0] [.2, .0, .31, .03, .01]
- Polo [0, 0, 0, 1, 0] [.1, .1, .5, .5, 0.2]

Embedding

Distributional hypothesis *linguistic items with similar distributions have similar meanings*



*Life is **complex**. It has both **real** and **imaginary** parts*

https://en.wikipedia.org/wiki/Distributional_semantics

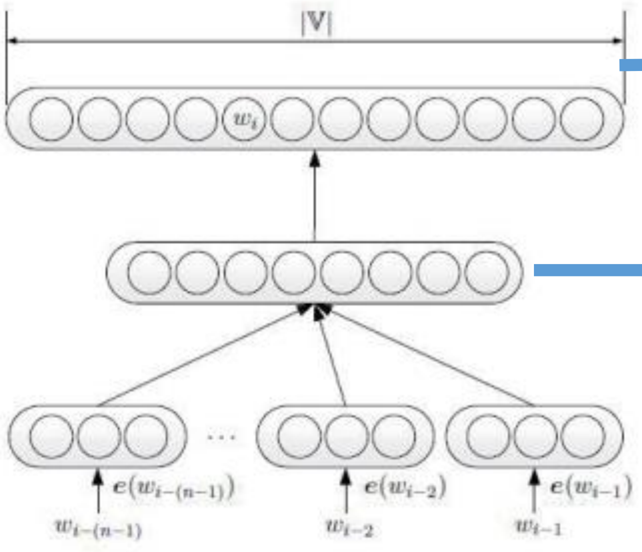
How to get Distributed representation

- Matrix Factorization
  - Word-word Matrix
  - Document-word Matrix
    - PLSA
    - LDA
- Sample-based Prediction
  - NNLM
  - C & W
  - Word2vec

Glove is a combination between these two schools of approaches

Levy, Omer, and Yoav Goldberg. "Neural word embedding as implicit matrix factorization." *Advances in neural information processing systems*. 2014.

NNLM to Word2vec



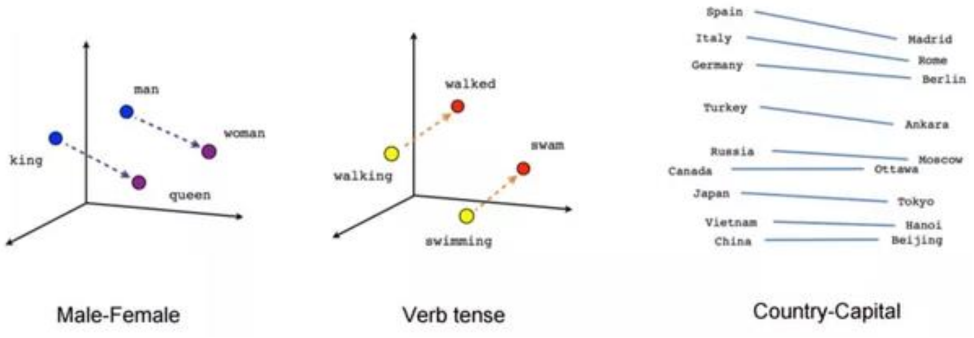
Negative sampling
Hierarchical softmax

Concat -> mean

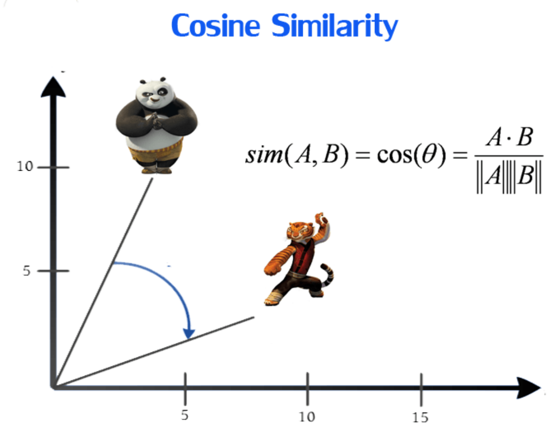
Bengio Y, Ducharme R, Vincent P, et al. A neural probabilistic language model[J]. Journal of machine learning research, 2003, 3(Feb): 1137-1155.
Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.

Advantage of word embedding

- Linguistic regulation
  - $\overrightarrow{king} - \overrightarrow{man} = \overrightarrow{queen} - \overrightarrow{woman}$



- Semantic matching
  - As the initial input Feature/**Weight** for NN



Only Word Embedding ?

Which is the most similar word of “Italy” ?

Maybe “Germany” or “Pasta” ?

You cannot **guarantee** that each similar word pair could help your matching ?

Nie Jianyun said in SIGIR 2016 Chinese-Author Workshop, Tsinghua University, Beijing

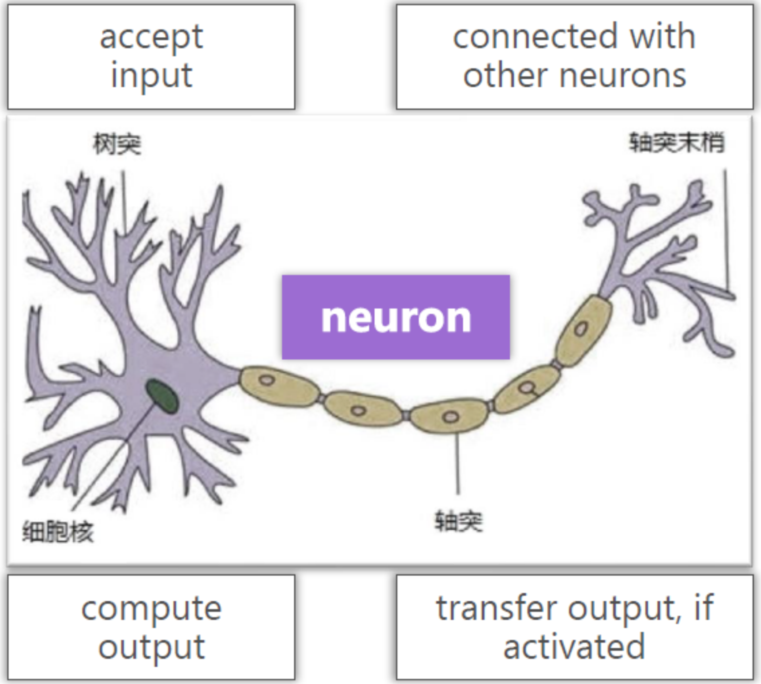
Background of Neural IR

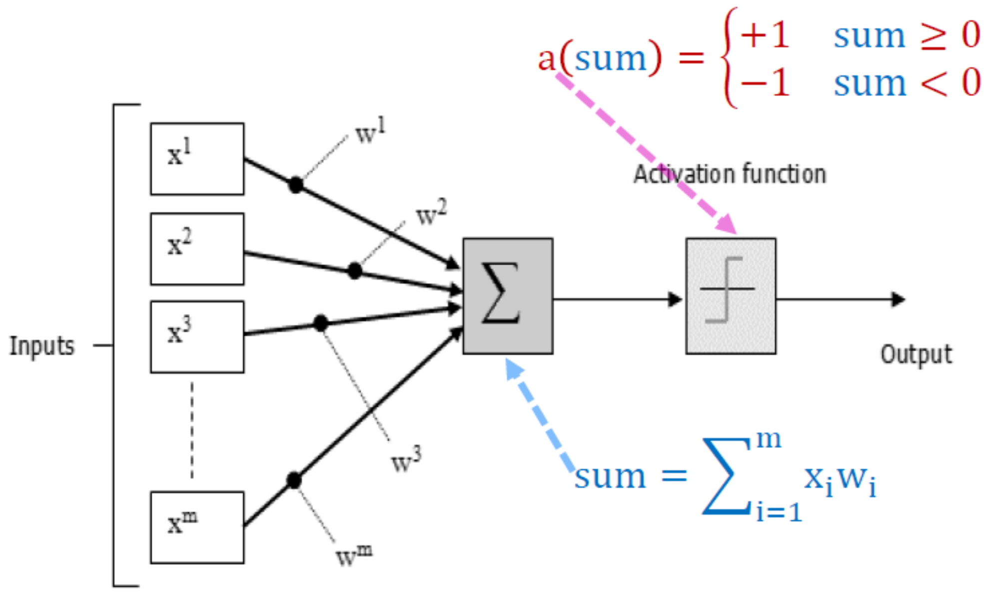
- Trends of DL for IR
- Word embedding
- **Neural network**
- DL for IR/NLP

Neural Network

- MLP
- CNN
  - **Shift/Space invariant**
- Recurrent NN -   [LSTM/GUR]
  - **Time-sensitive**
- Recursive NN
  - **Structure-sensitive**
- Special Case
  - Seq2seq
  - GAN
  - Reinforced Learning

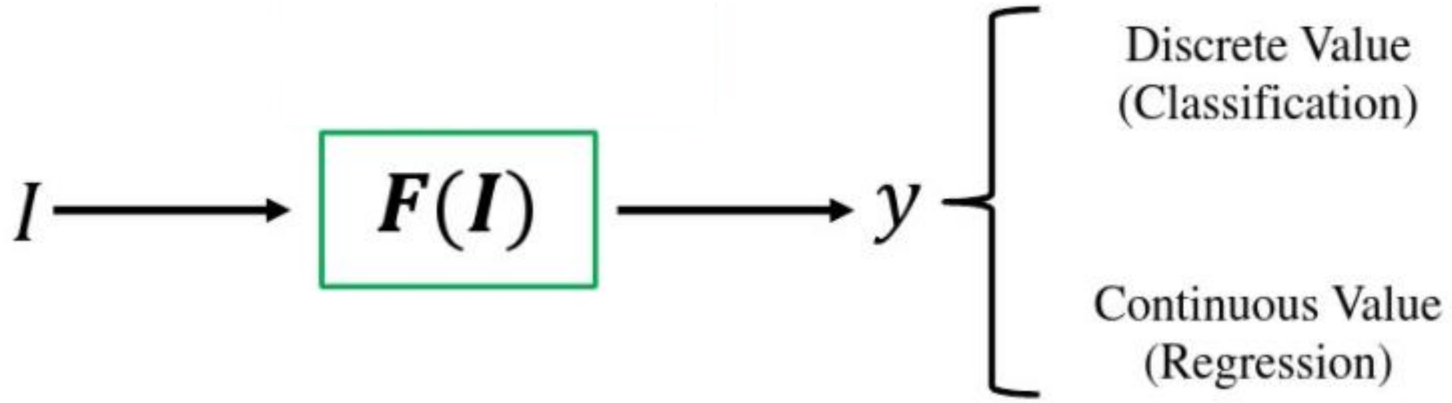
MLP





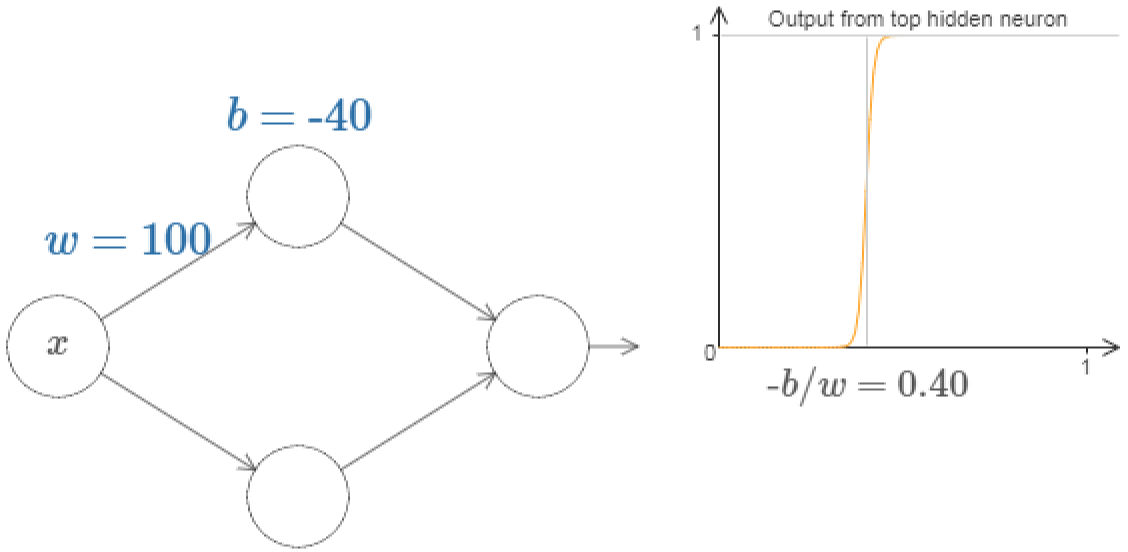
Perception (感知机)

UAT in MLP



Multi-layer Non-linear Mapping - > **U**niversal **A**pproximation **T**heorem

A sample of θ(wx+b)

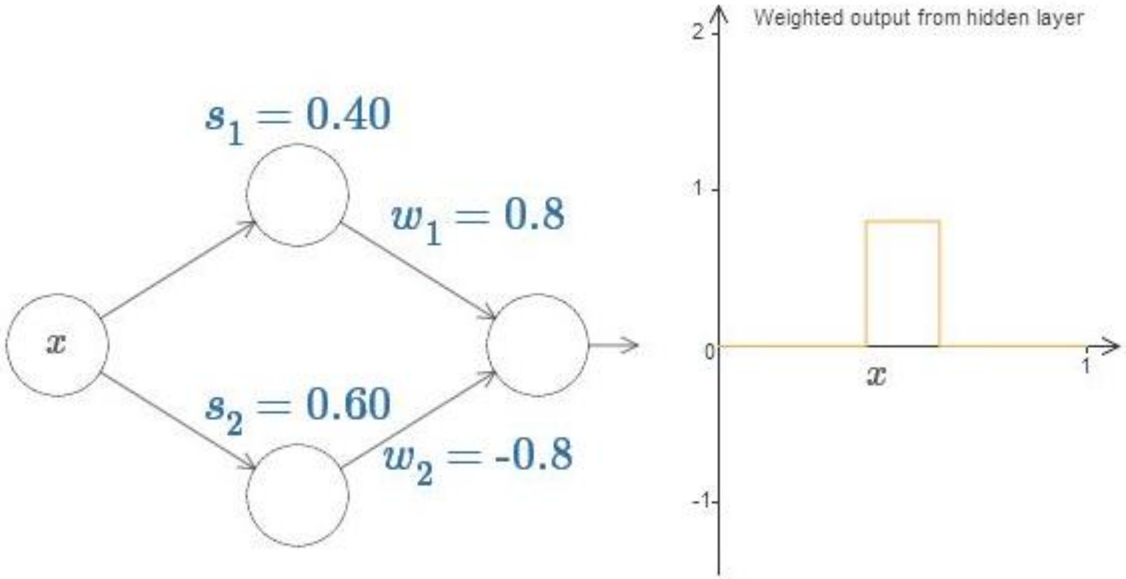


$$s = -b/w.$$

$$\sigma(wx+b), \text{ where } \sigma(z) \equiv 1/(1+e^{-z})$$

http://neuralnetworksanddeeplearning.com/chap4.html

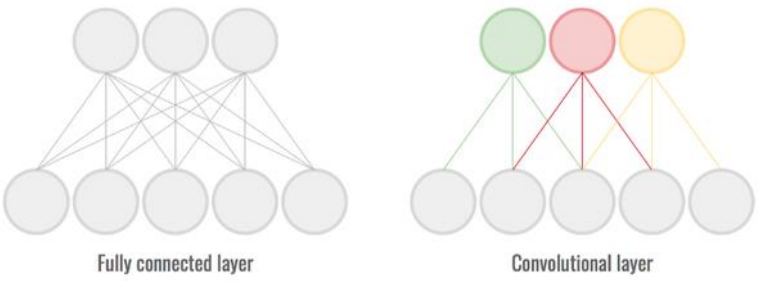
An another sample

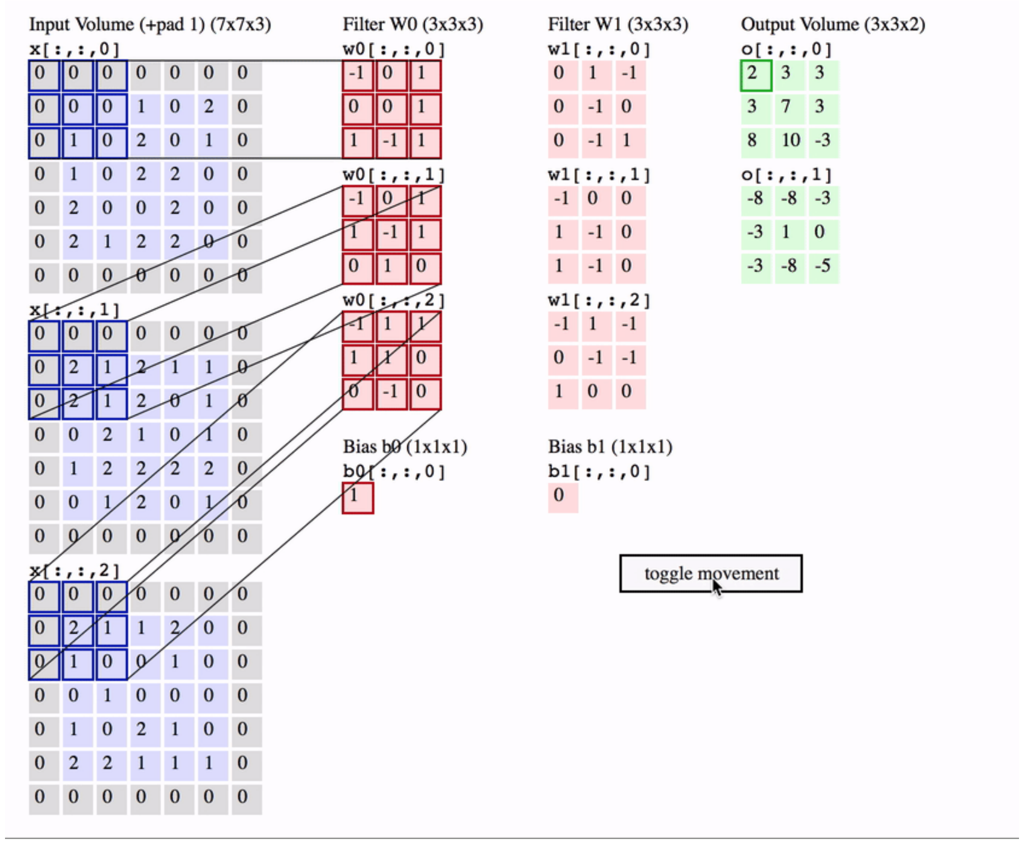


$$\sigma(wx + b), \text{ where } \sigma(z) \equiv 1/(1 + e^{-z})$$

From MLP to CNN

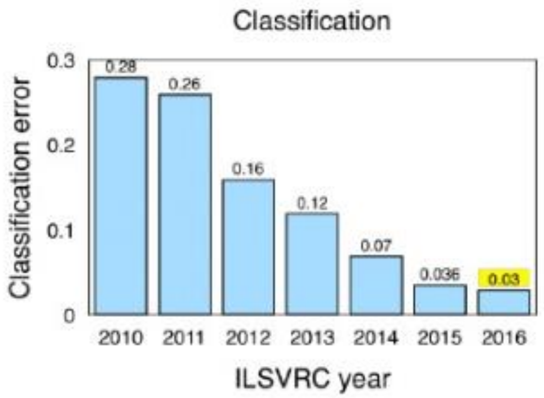
- Local connection
- Shared weight
- Pooling strategy



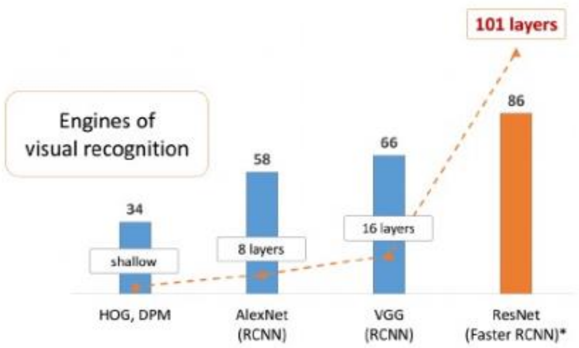


Deep NN in CV

Top 5 error in ImageNet classification

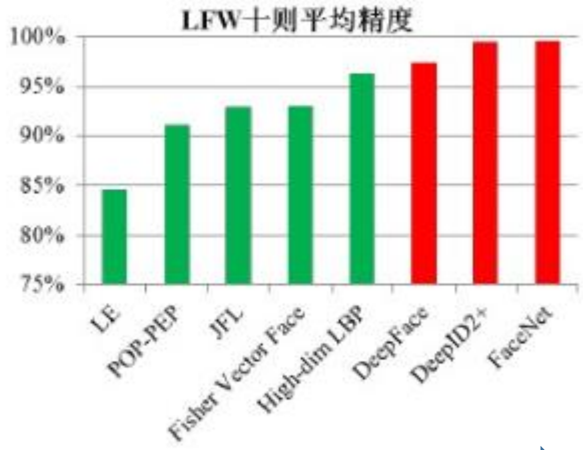


Deep NN



MAP in Pascal VOC visual recognition

10-fold mean precision Face recognition LFW dataset



Credited to Prof. Shiguang Shan with modified

End-2-end in CV

- Tradition CV

- Modern CV： Unsupervised mid-representation

- DNN CV： end-2-end

CNN

- Basic CNN
- Kalchbrenner N, Grefenstette E, Blunsom P. A convolutional neural network for modelling sentences[J]. arXiv preprint arXiv:1404.2188, 2014
- Kim CNN
- VDCNN

CNN [kim EMNLP 2014]

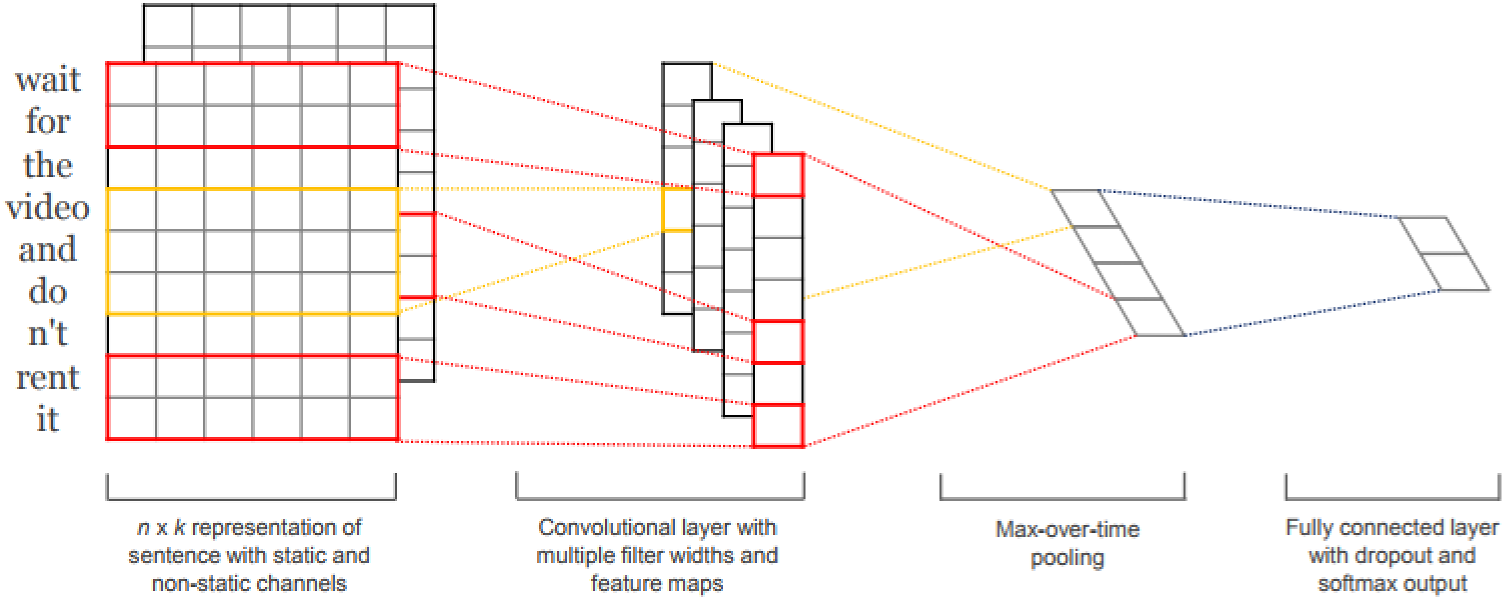


Figure 1: Model architecture with two channels for an example sentence.

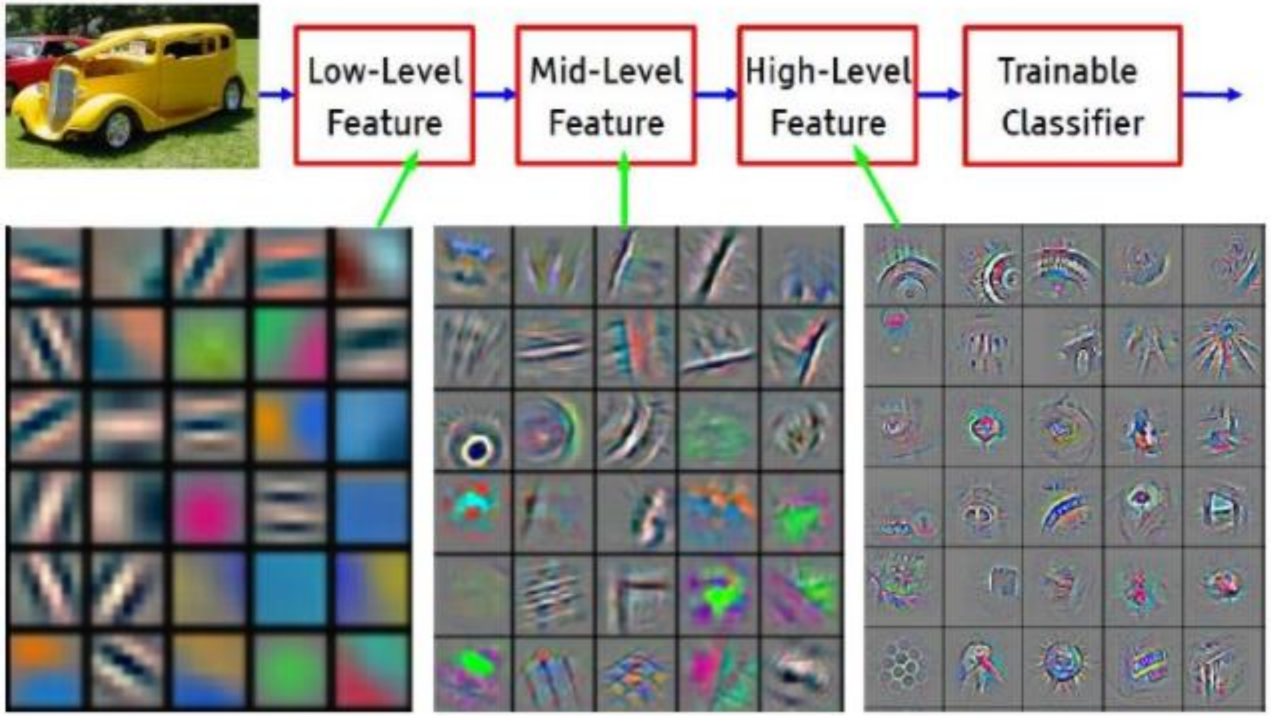
Model	MR	SST-1	SST-2	Subj	TREC	CR	MPQA
CNN-rand	76.1	45.0	82.7	89.6	91.2	79.8	83.4
CNN-static	81.0	45.5	86.8	93.0	92.8	84.7	**89.6**
CNN-non-static	**81.5**	48.0	87.2	93.4	93.6	84.3	89.5
CNN-multichannel	81.1	47.4	**88.1**	93.2	92.2	**85.0**	89.4
RAE (Socher et al., 2011)	77.7	43.2	82.4	–	–	–	86.4
MV-RNN (Socher et al., 2012)	79.0	44.4	82.9	–	–	–	–
RNTN (Socher et al., 2013)	–	45.7	85.4	–	–	–	–
DCNN (Kalchbrenner et al., 2014)	–	48.5	86.8	–	93.0	–	–
Paragraph-Vec (Le and Mikolov, 2014)	–	**48.7**	87.8	–	–	–	–
CCAE (Hermann and Blunsom, 2013)	77.8	–	–	–	–	–	87.2
Sent-Parser (Dong et al., 2014)	79.5	–	–	–	–	–	86.3
NBSVM (Wang and Manning, 2012)	79.4	–	–	93.2	–	81.8	86.3
MNB (Wang and Manning, 2012)	79.0	–	–	**93.6**	–	80.0	86.3
G-Dropout (Wang and Manning, 2013)	79.0	–	–	93.4	–	82.1	86.1
F-Dropout (Wang and Manning, 2013)	79.1	–	–	**93.6**	–	81.9	86.3
Tree-CRF (Nakagawa et al., 2010)	77.3	–	–	–	–	81.4	86.1
CRF-PR (Yang and Cardie, 2014)	–	–	–	–	–	82.7	–
SVM_S (Silva et al., 2011)	–	–	–	–	**95.0**	–	–

Go deeper or not?

- DEEP
  - Slower
  - Overfitting
    - More Parameters, more data need to feed
  - Hard for convergence
    - Highway network
    - Residual Block
    - Inception
- Shallow: one-layer
  - Fast
  - Less data, es. Fastext.

Go deeper or not?

- Image recognition: Pixel → edge → texton → motif → part → object
- Text: Character → word → word group → clause → sentence → story
- Speech: Sample → spectral band → sound → ... → phone → phoneme → word



Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

Modified from Prof. LeCun and Prof. Bengio

Very Large CNN [Conneau EACL]

Corpus:	AG	Sogou	DBP.	Yelp P.	Yelp F.	Yah. A.	Amz. F.	Amz. P.
Method	n-TFIDF	n-TFIDF	n-TFIDF	ngrams	Conv	Conv+RNN	Conv	Conv
Author	[Zhang]	[Zhang]	[Zhang]	[Zhang]	[Zhang]	[Xiao]	[Zhang]	[Zhang]
Error	7.64	2.81	1.31	4.36	37.95*	28.26	40.43*	4.93*
[Yang]	-	-	-	-	-	24.2	36.4	-

Table 4: Best published results from previous work. Zhang et al. (2015) best results use a Thesaurus data augmentation technique (marked with an *). Yang et al. (2016)'s hierarchical methods is particularly adapted to datasets whose samples contain multiple sentences.

Depth	Pooling	AG	Sogou	DBP.	Yelp P.	Yelp F.	Yah. A.	Amz. F.	Amz. P.
9	Convolution	10.17	4.22	1.64	5.01	37.63	28.10	38.52	4.94
9	KMaxPooling	9.83	3.58	1.56	5.27	38.04	28.24	39.19	5.69
9	MaxPooling	9.17	3.70	1.35	4.88	36.73	27.60	37.95	4.70
17	Convolution	9.29	3.94	1.42	4.96	36.10	27.35	37.50	4.53
17	KMaxPooling	9.39	3.51	1.61	5.05	37.41	28.25	38.81	5.43
17	MaxPooling	8.88	3.54	1.40	4.50	36.07	27.51	37.39	4.41
29	Convolution	9.36	3.61	1.36	4.35	**35.28**	27.17	37.58	**4.28**
29	KMaxPooling	**8.67**	**3.18**	1.41	4.63	37.00	27.16	38.39	4.94
29	MaxPooling	8.73	3.36	**1.29**	**4.28**	35.74	**26.57**	**37.00**	4.31

Table 5: Testing error of our models on the 8 data sets. No data preprocessing or augmentation is used.

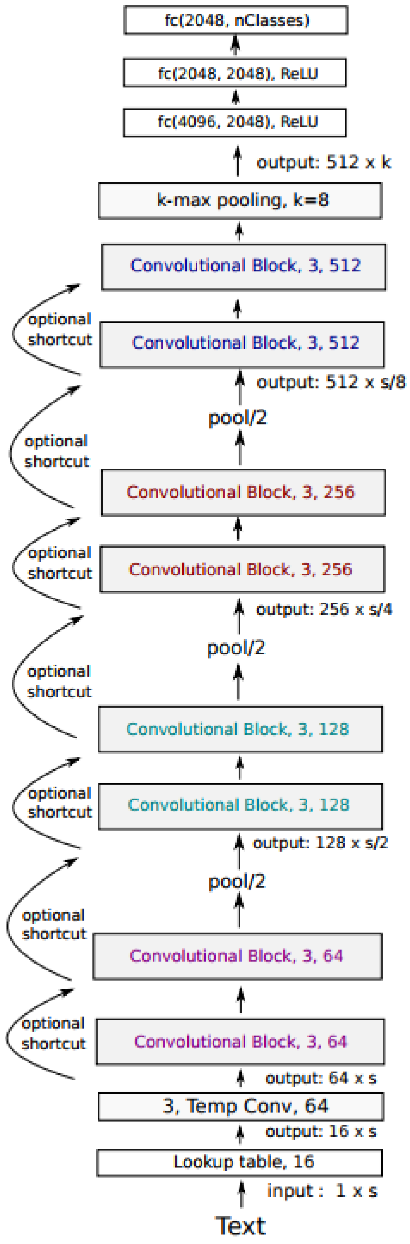


Figure 1: VDCNN architecture.

FASTEX [EACL 2017]

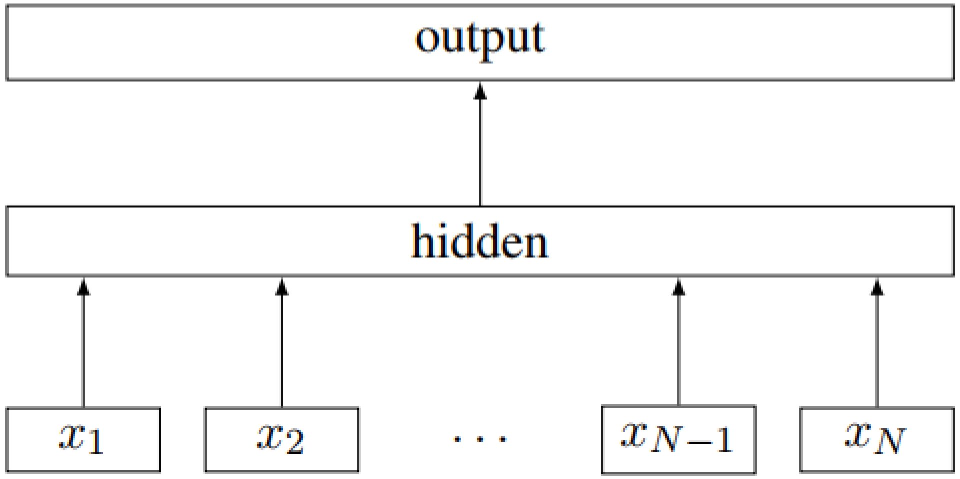
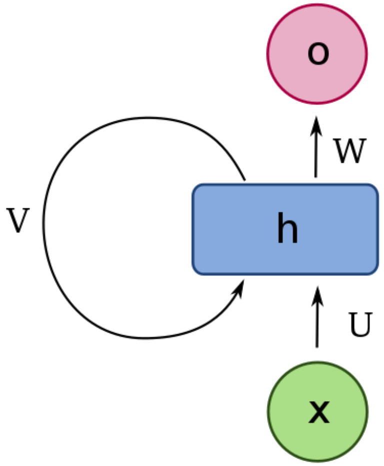


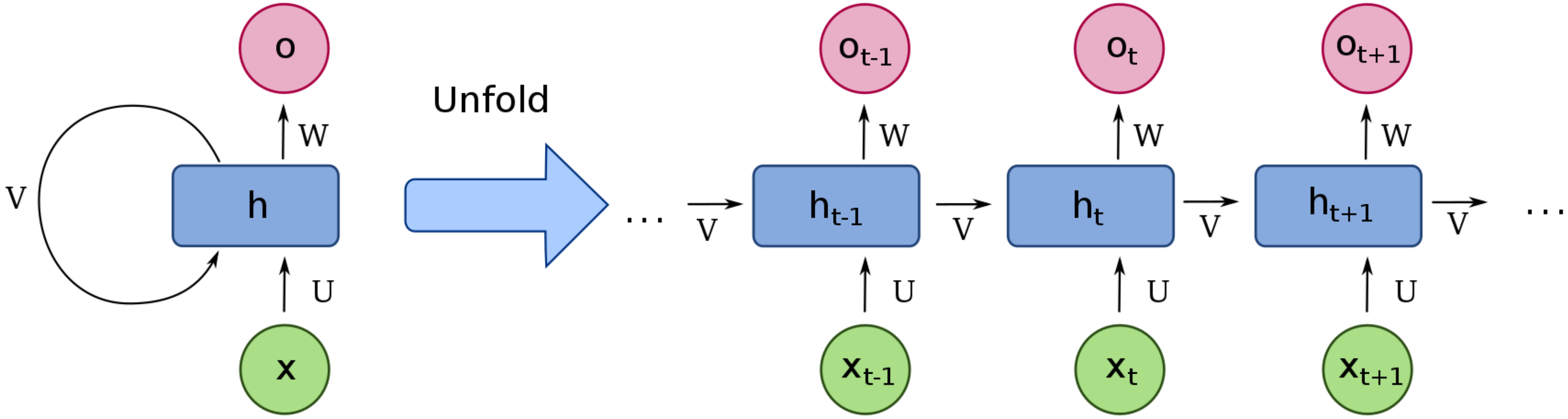
Figure 1: Model architecture of `fastText` for a sentence with N ngram features $x_1, \ldots, x_N$. The features are embedded and averaged to form the hidden variable.

Model	Yelp'13	Yelp'14	Yelp'15	IMDB
SVM+TF	59.8	61.8	62.4	40.5
CNN	59.7	61.0	61.5	37.5
Conv-GRNN	63.7	65.5	66.0	42.5
LSTM-GRNN	65.1	67.1	67.6	45.3
`fastText`	64.2	66.2	66.6	45.2

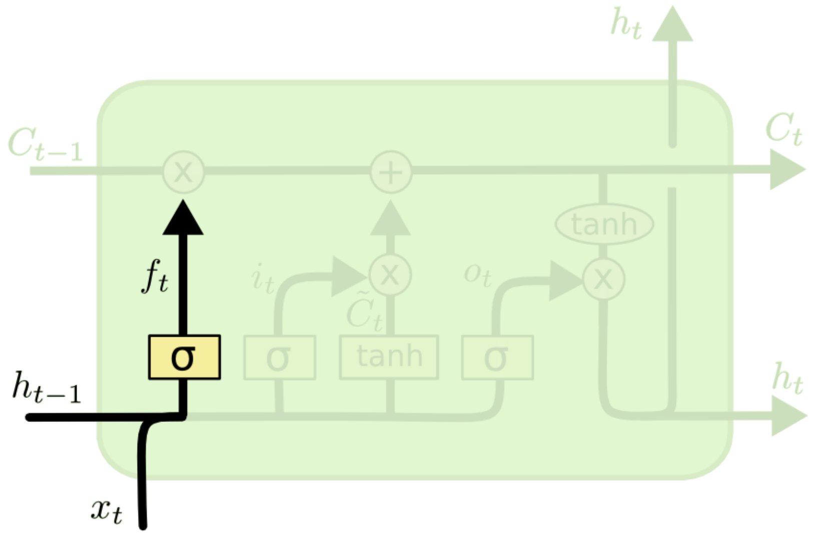
RNN



RNN

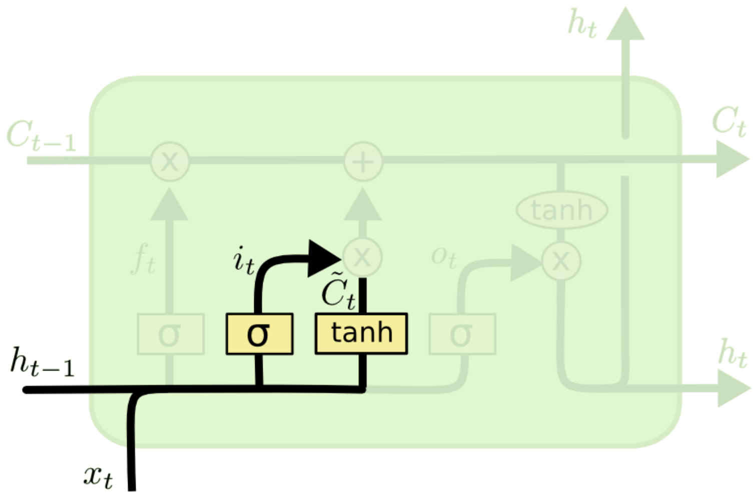


Forget gate



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

Input gate



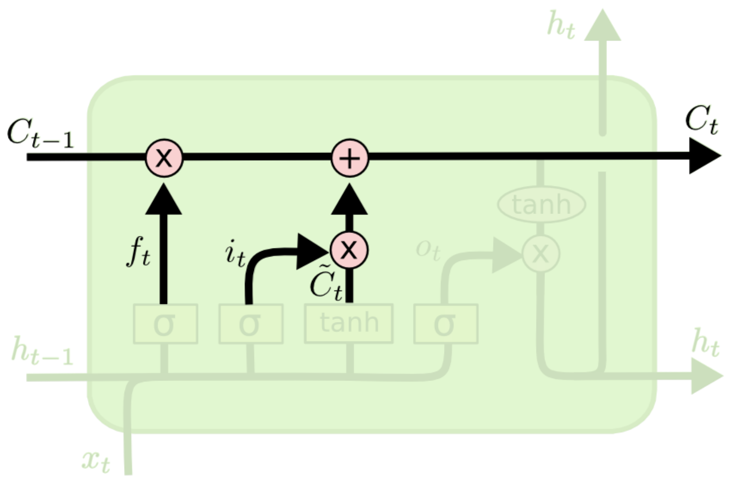
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

replace tanh with **softsign** (not softmax) activation for prevent **overfitting**

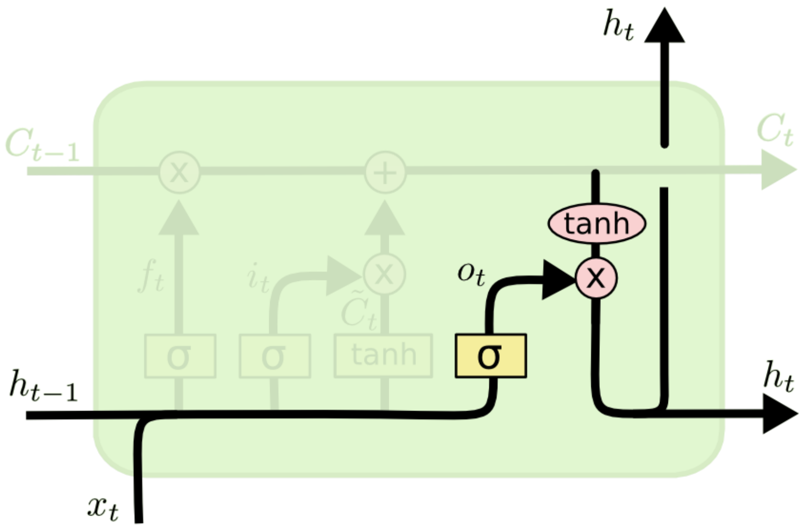
https://zhuanlan.zhihu.com/p/21952042

Forgotten + input



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

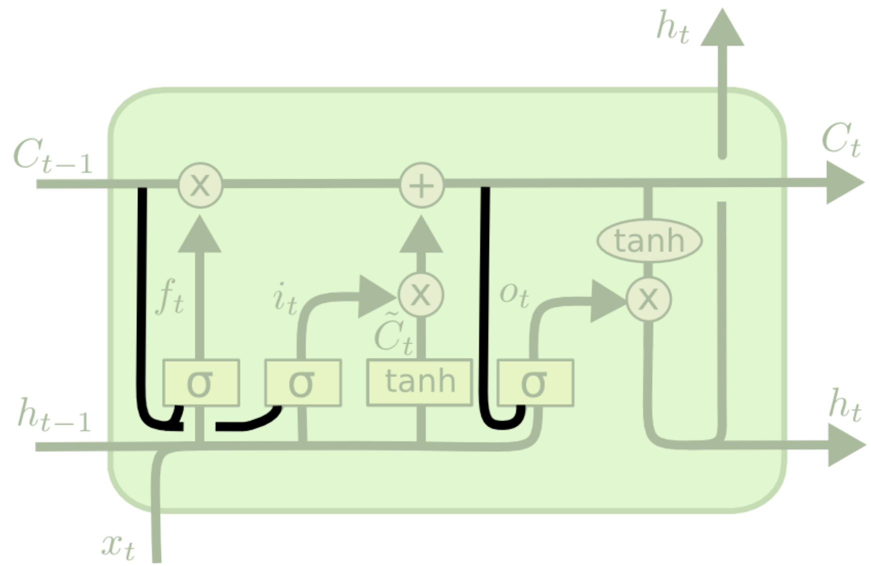
Output Gate



$$o_t = \sigma\left(W_o \left[h_{t-1}, x_t\right] + b_o\right)$$

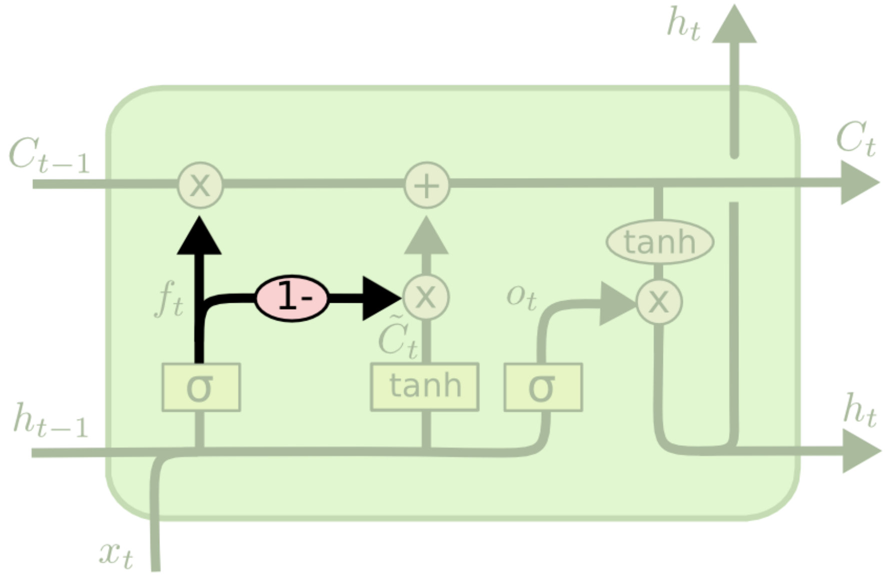
$$h_t = o_t * \tanh\left(C_t\right)$$

LSTM **Variants:** Peephole connections



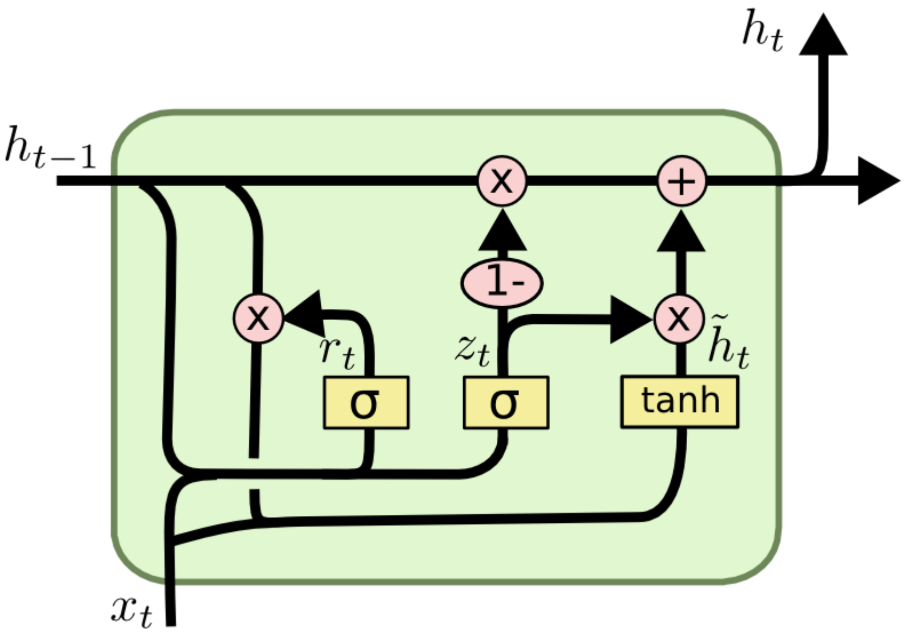
$$f_t = \sigma\left(W_f \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \; + \; b_f\right)$$
$$i_t = \sigma\left(W_i \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \; + \; b_i\right)$$
$$o_t = \sigma\left(W_o \cdot [\boldsymbol{C_t}, h_{t-1}, x_t] \; + \; b_o\right)$$

LSTM **Variants:** coupled forget and input gates



$$C_t = f_t * C_{t-1} + \boldsymbol{(1 - f_t)} * \tilde{C}_t$$

LSTM **Variants:** GRU



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

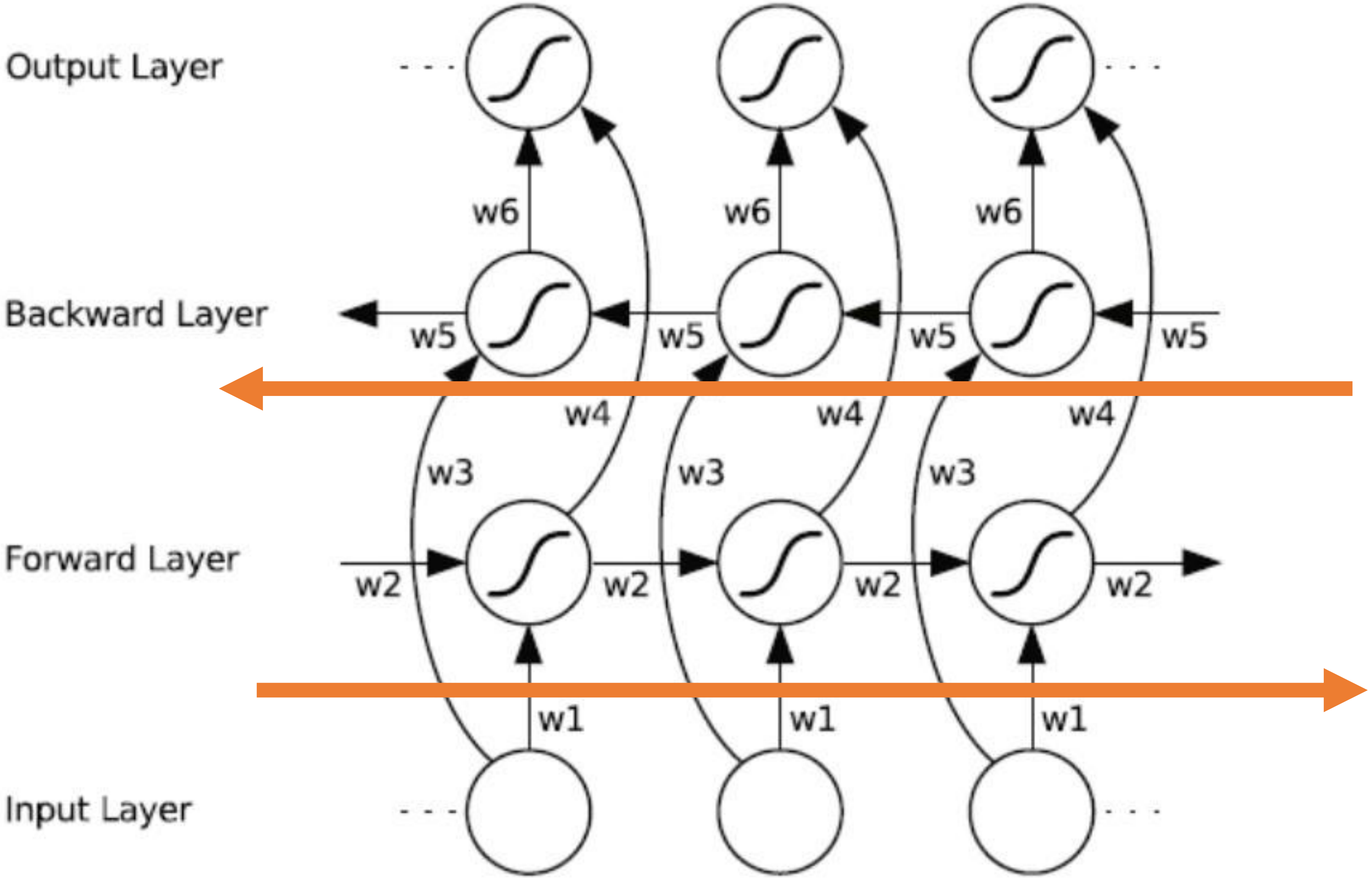
$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

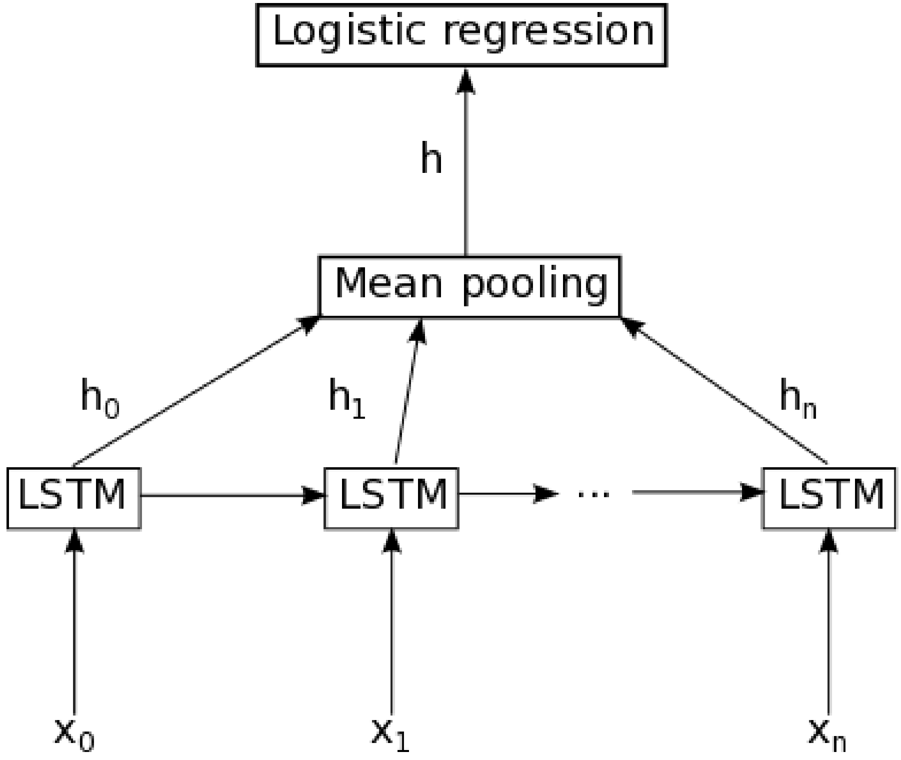
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

- ✓ Hidden = Cell
- ✓ Forget gate + input gate =1

BiLSTM

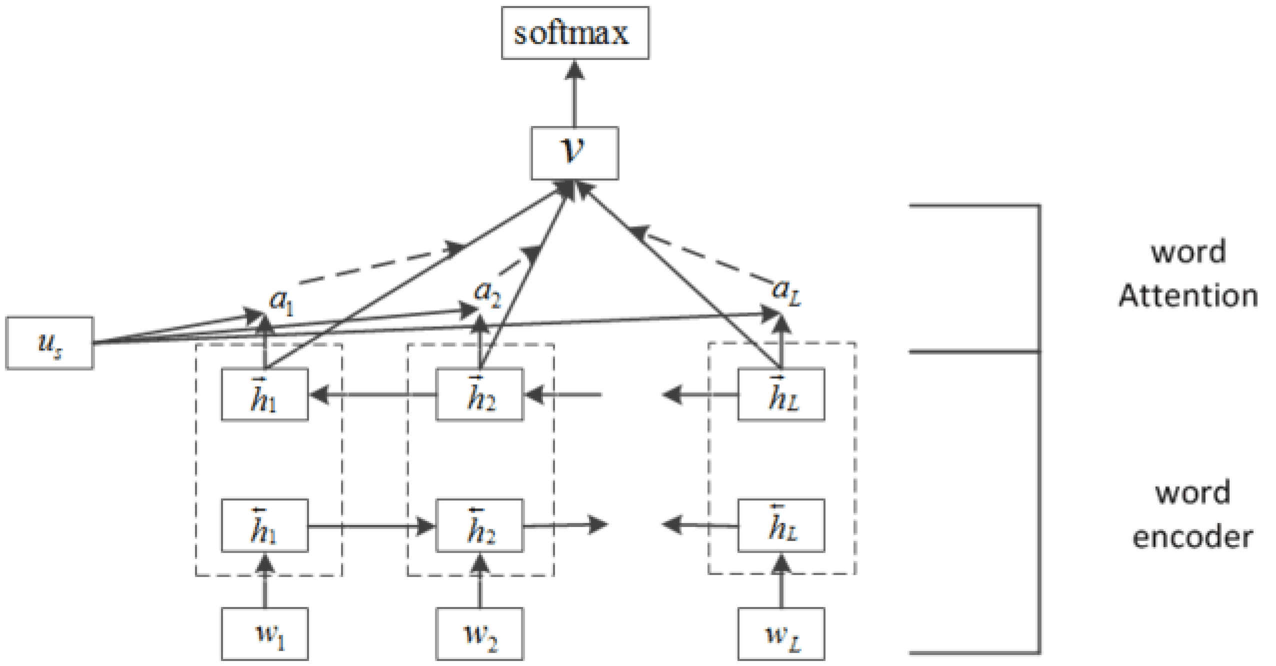

Output Layer
Backward Layer
Forward Layer
Input Layer
w6
w6
w6
w5
w5
w5
w5
w4
w4
w4
w3
w3
w3
w2
w2
w2
w2
w1
w1
w1

Last or Mean?



Logistic regression
h
Mean pooling
h0
h1
hn
LSTM
LSTM
...
LSTM
x0
x1
xn

RNN/LSTM with Attention



BIGRU : 93%

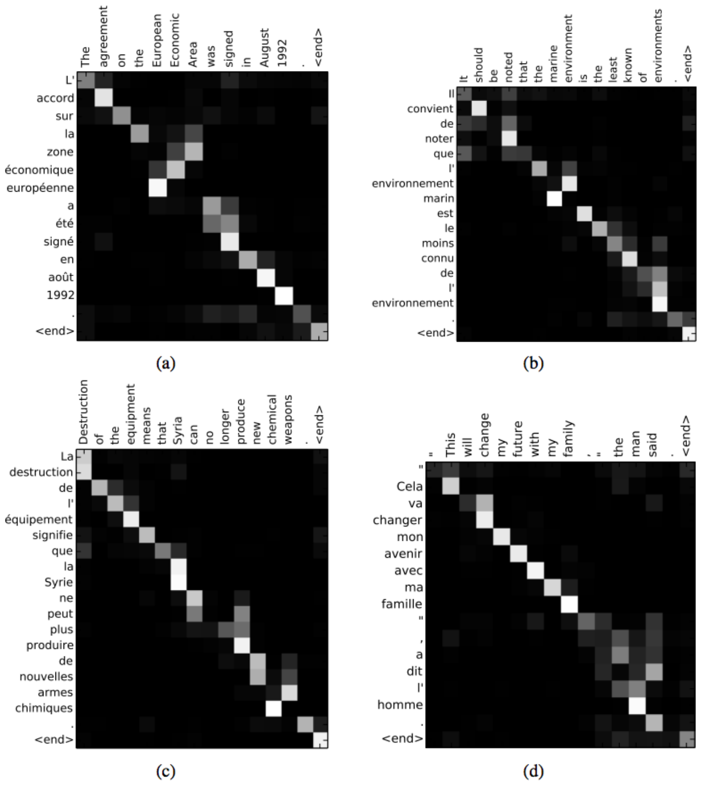
BILSTM : 91.43%

BIGRU_ATTENTION : 95.4%

BILSTM_ATTENTION : 96.2%

https://www.jianshu.com/p/4fbc4939509f

Visualization of Attention in RNN/LSTM



Machine Translation

Figure 5. Examples of mistakes where we can use attention to gain intuition into what the model saw.

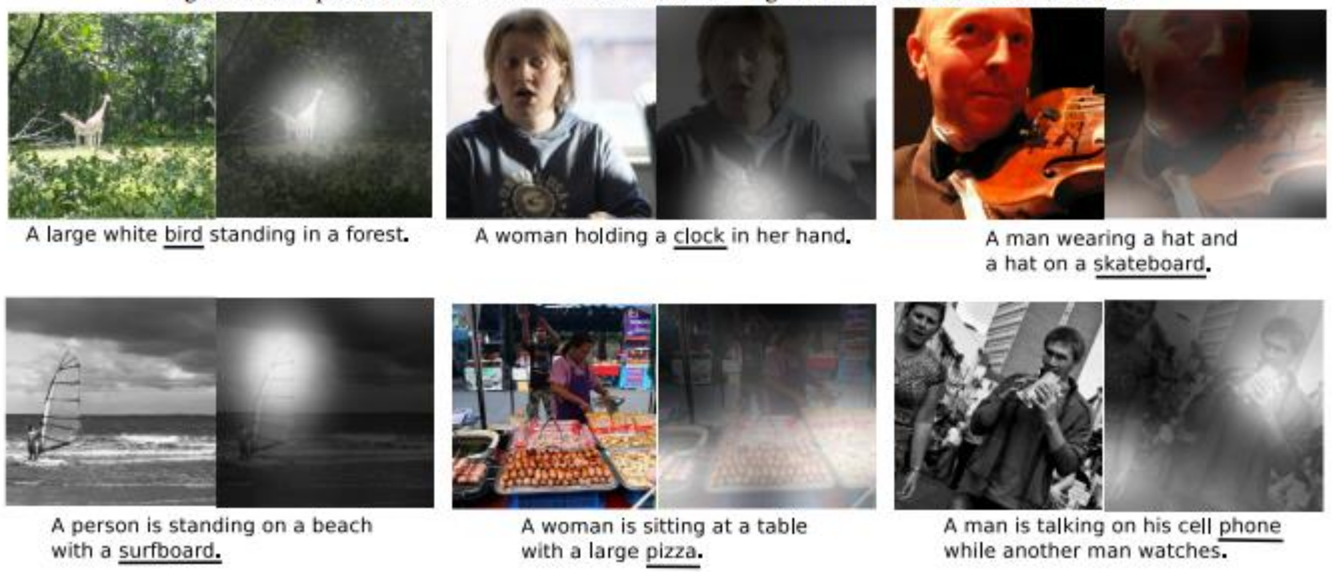
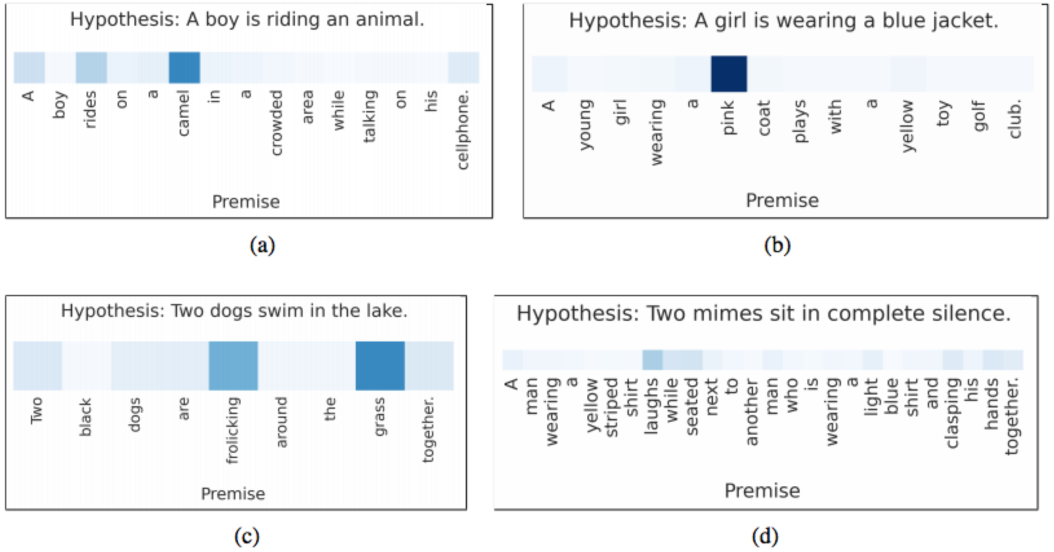
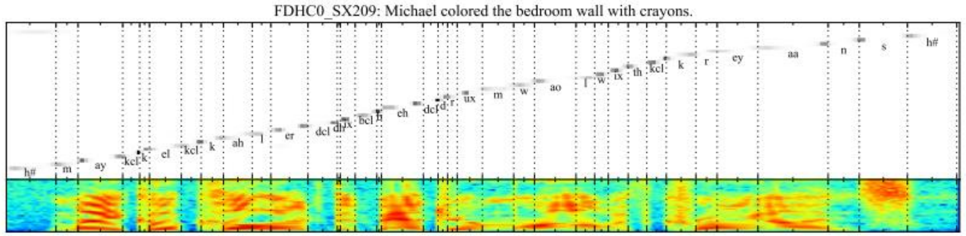


Image Caption

Visualization of Attention in RNN/LSTM

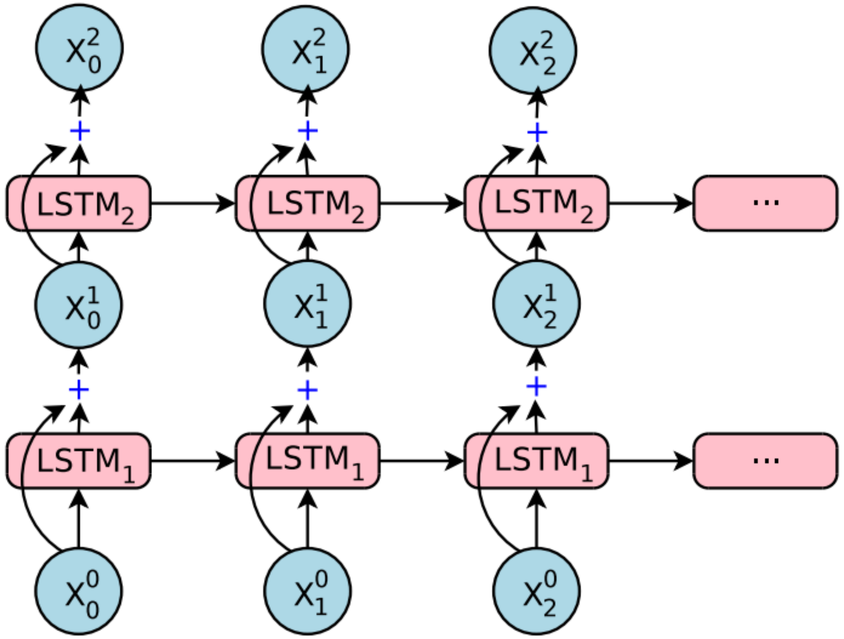


Sematic Entailment



Speech Recognition

Deeper LSTM

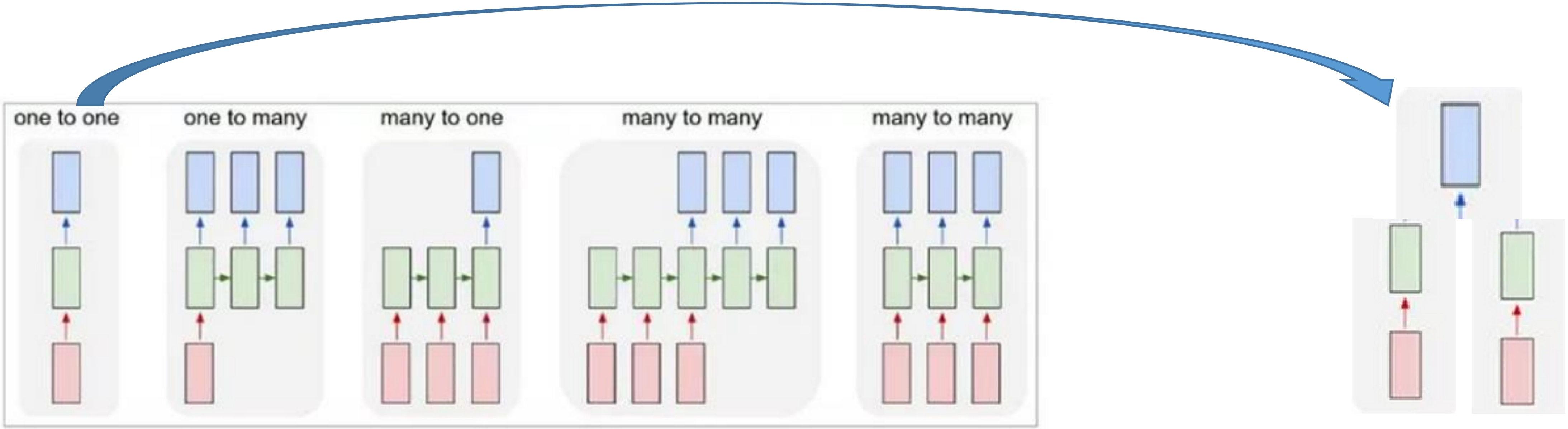


Deep is not necessary, but more feeding data!!!

Background of Neural IR

- Trends of DL for IR
- Word embedding
- Neural network
- **DL for IR/NLP**

Tasks in IR/NLP



- Classification: assigning a label to a string

$$s \rightarrow c$$

- Matching: matching two strings

$$s, t \rightarrow \mathbf{R}^{+}$$

- Translation: transforming one string to another

$$s \rightarrow t$$

- Structured prediction: mapping string to structure

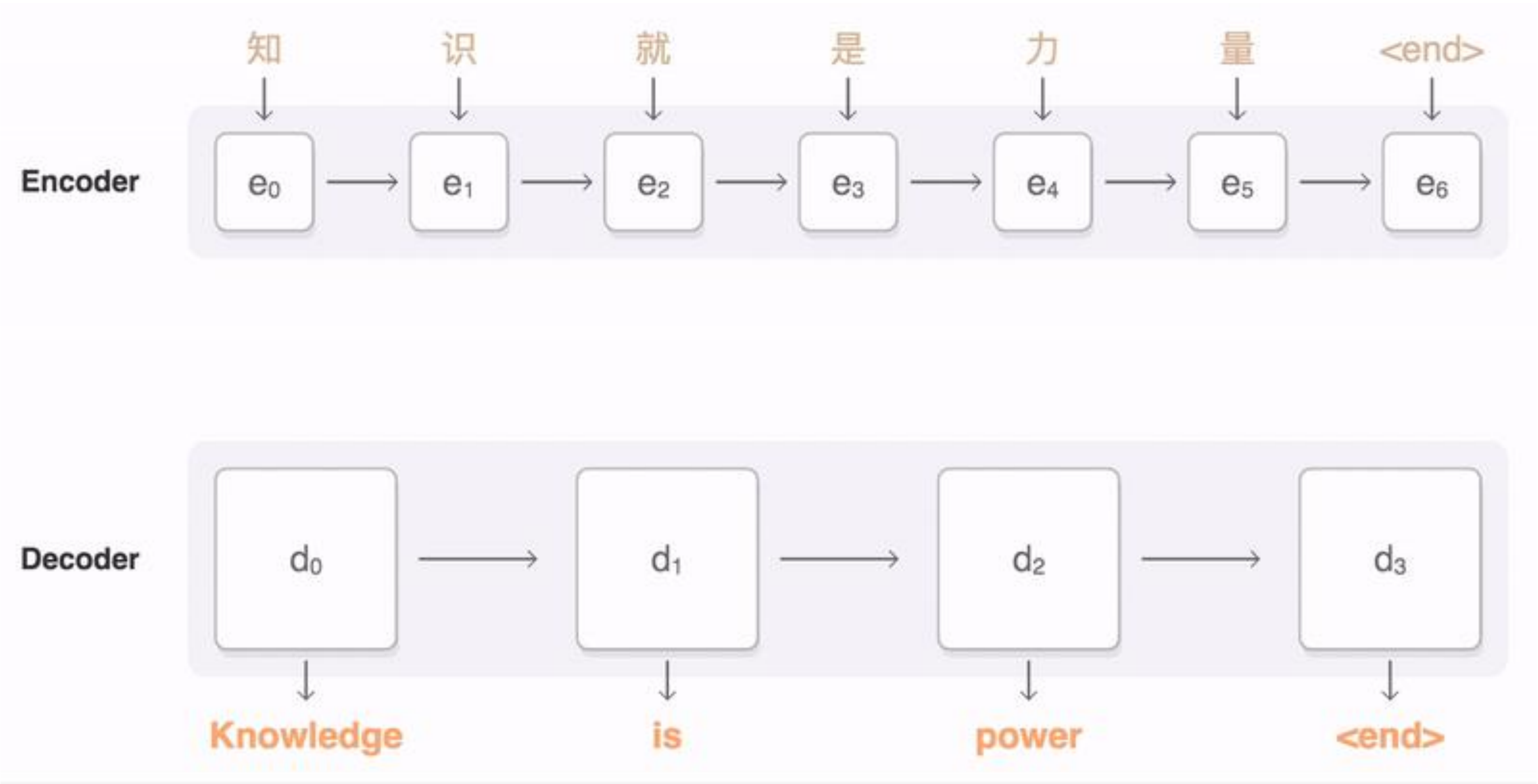
$$s \rightarrow s'$$

Credited by Hang li

Fundamental Demo In Code with PyTorch pseudo code

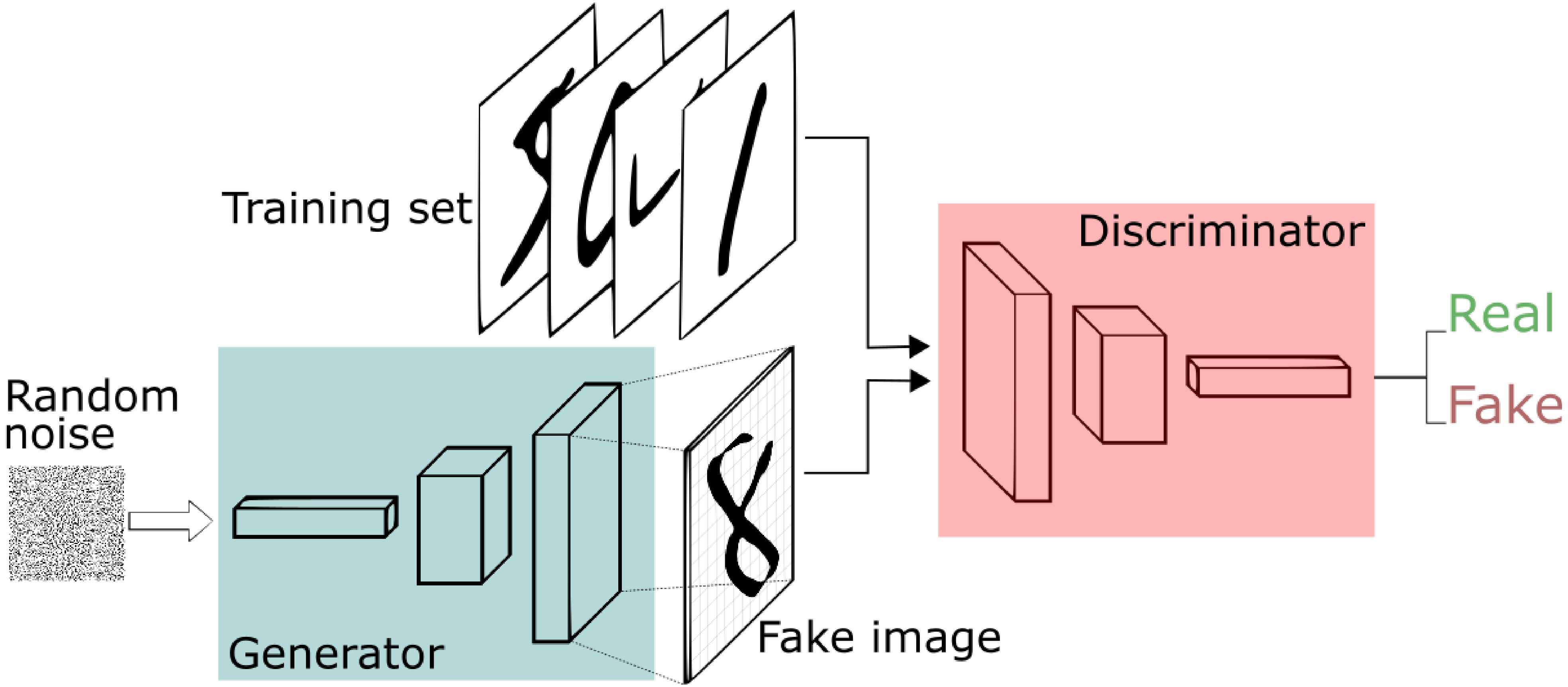
- Model = LSTM/CNN/Capsule/…
- text,lable = Dataset.nextBatch()
- representation = Model(text)
- Classification = FC(representation) FC : Mapping to label size
- Translation = Decode(representation)
- Matching = Cosine(representation1, representation2)
- Sequential_labelling = FCs(representation**s**)

Seq2seq

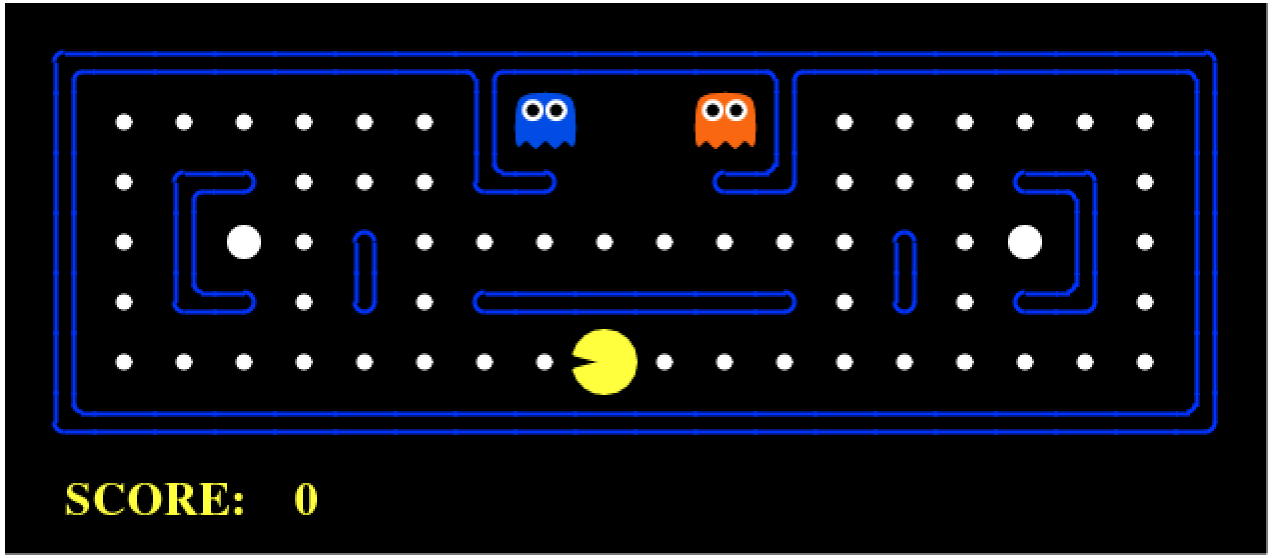


知
识
就
是
力
量
<end>
Encoder
e_0
e_1
e_2
e_3
e_4
e_5
e_6
Decoder
d_0
d_1
d_2
d_3
Knowledge
is
power
<end>

GAN


Training set
Discriminator
Real
Fake
Random noise
Generator
Fake image

Reinforced learning



Compared to the supervised learning:
You can not know the current reward from the current action, namely a delayed reward,
only in the case that the game is finished.

https://www.kdnuggets.com/2018/03/5-things-reinforcement-learning.html

Quantum-style Cooking

- Hilbert semantic space
  - Complex word embedding
  - Hilbert semantic space
  - Application in Text classification
  - Application in Question Answering

- Ideas
  - Dynamics for thematic issues
  - Evolved Density matrix for language model

Complex word-embedding

- Super-linearity superposition with phase

$$
\begin{aligned}
z^* &= z_1 + z_2 = r_1 e^{i\theta_1} + r_2 e^{i\theta_2} \\
&= \sqrt{r_1^2 + r_2^2 + 2r_1 r_2 \cos(\theta_2 - \theta_1)} \times e^{i \arctan\left(\frac{r_1 \sin(\theta_1) + r_2 \sin(\theta_2)}{r_1 \cos(\theta_1) + r_2 \cos(\theta_2)}\right)}
\end{aligned}
$$

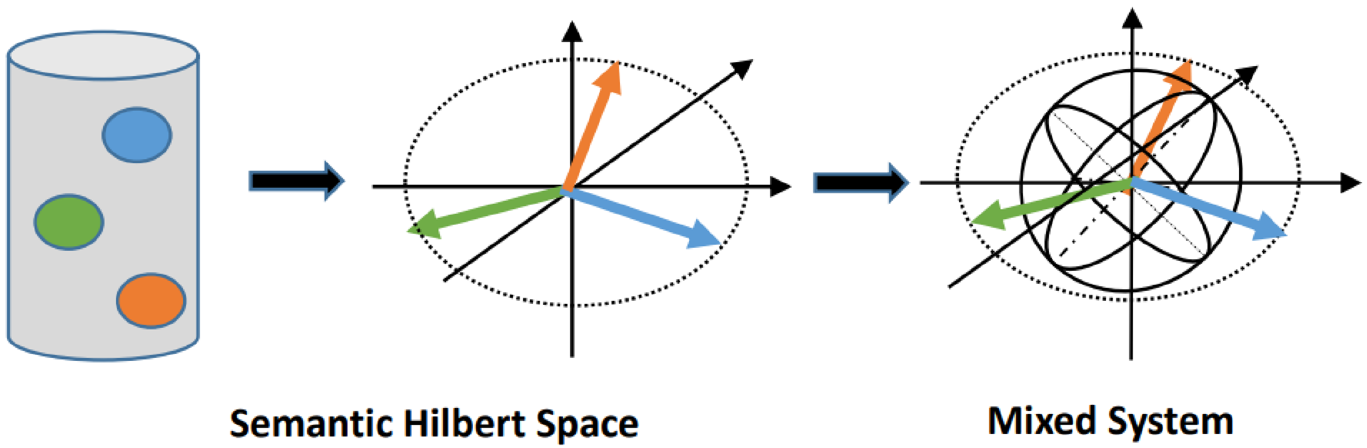
Li Qiuchi, Uprety Sagar, Wang Benyou , Song Dawei [Quantum-inspired Complex Word Embedding, ACL 2018 3rd Workshop on Representation Learning for NLP], **ACL 2018 RepL4NLP**

Hilbert Semantic Space

- **Unify** these four things in a complex-valued space
  - Sememes
  - Word
  - Phrase/Sentence/Documents
  - Topic as measurements

Definition

- Sememes as basic state
- Word as superstition state
- Sentence as mixed system

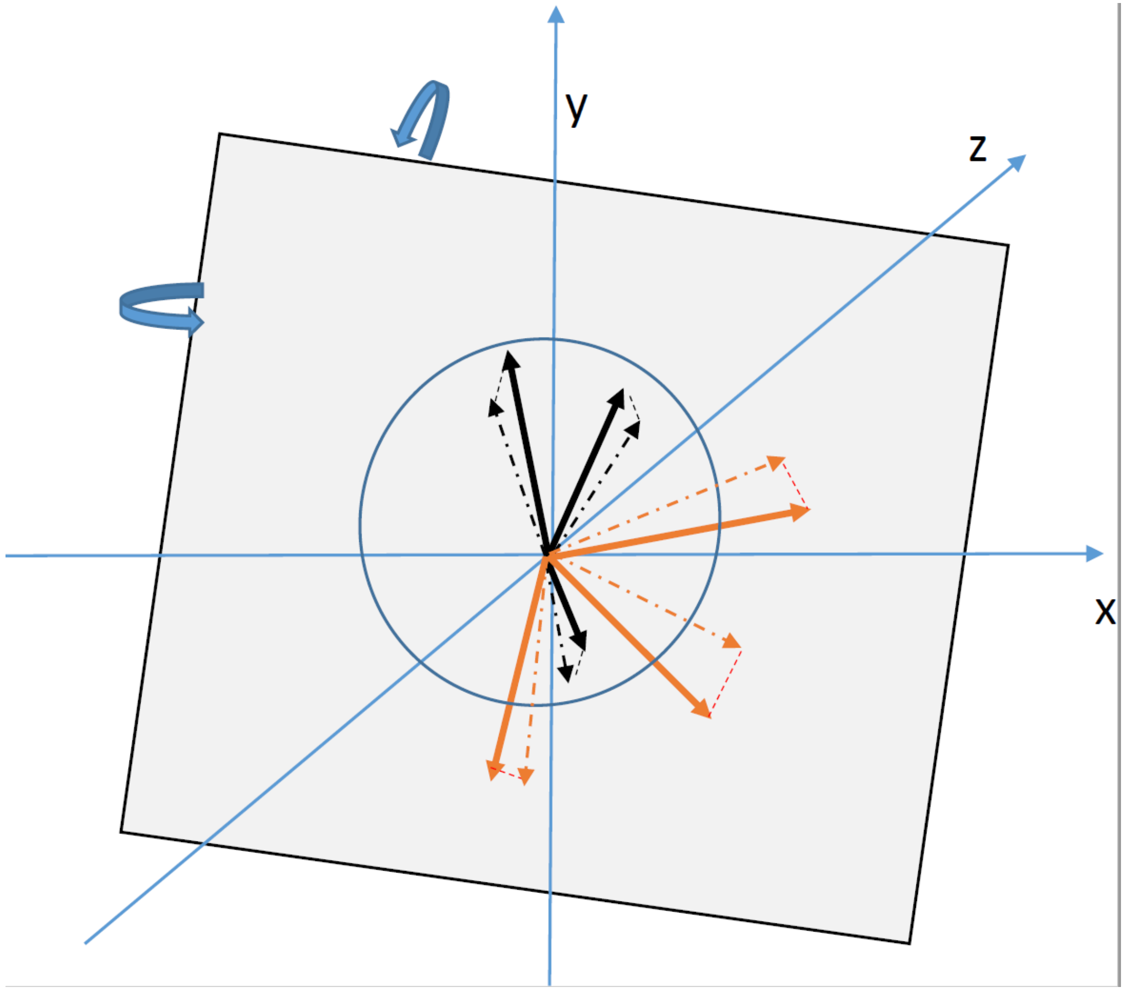


Complex word embedding

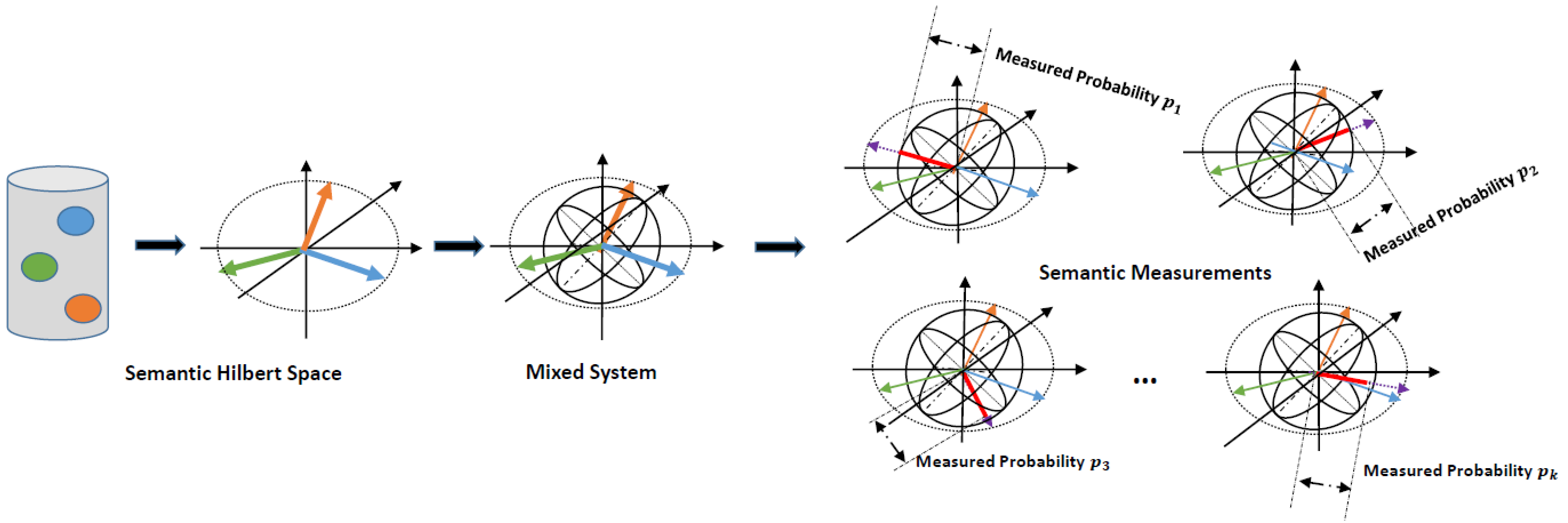
- Dimension: the number of
- Length : weight
- Amplitude part: meaning
- Phase part: polarity ?

- How to infer the overall polarity from the polarity of each words?
  - Is there any quantum phenomena here ?

Trainable Measurements for sentence classification

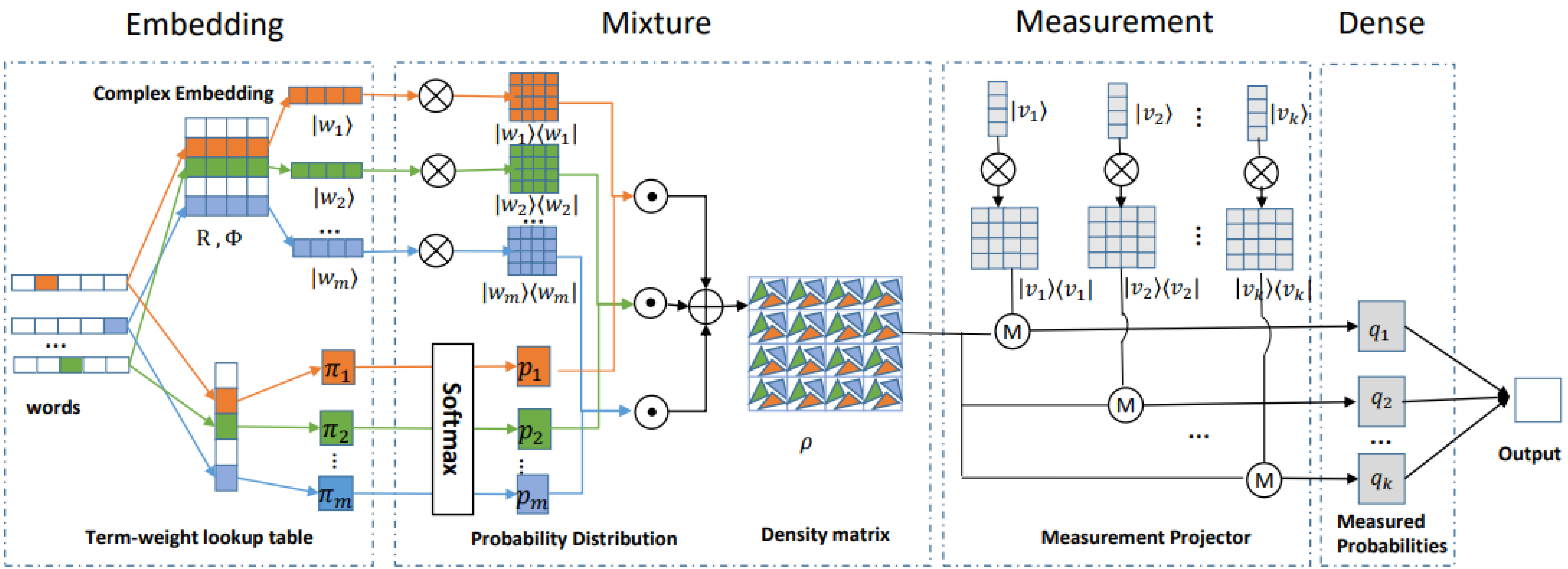


Framework



Semantic Hilbert Space
Mixed System
Measured Probability p_1
Measured Probability p_2
Semantic Measurements
Measured Probability p_3
...
Measured Probability p_k

Implements



Embedding
Mixture
Measurement
Dense
Complex Embedding
$|w_1\rangle$
$|w_2\rangle$
$|w_m\rangle$
R , Φ
words
Term-weight lookup table
π_1
π_2
π_m
Softmax
p_1
p_2
p_m
$|w_1\rangle\langle w_1|$
$|w_2\rangle\langle w_2|$
$|w_m\rangle\langle w_m|$
Probability Distribution
ρ
Density matrix
$|v_1\rangle$
$|v_2\rangle$
$|v_k\rangle$
$|v_1\rangle\langle v_1|$
$|v_2\rangle\langle v_2|$
$|v_k\rangle\langle v_k|$
M
Measurement Projector
q_1
q_2
q_k
Measured Probabilities
Output

Physical meaning for our models

Table 3: Physical meaning and constraint for each component

Components	Traditional DNN	NNQLM [56]	QPDN
Input embedding	arbitrary real vector $(-\infty, \infty)$	arbitrary real vector $(-\infty, \infty)$	unit complex vector, corresponding to superposition state $\{w \mid w \in C^n, \|\|w\|\|_2 = 1\}$
Low-level representation	arbitrary real vector $(-\infty, \infty)$	fake, real-valued density matrix $\{\rho \mid \rho \in \mathcal{R}^{n*n}\}$,	density matrix, corresponding to mixed state $\{\rho \mid \rho = \rho^*, tr(\rho) = 1, \mu\rho\mu^T > 0\ \forall \mu \neq \overrightarrow{0}, \rho \in C^{n*n}\}$,
Abstraction	CNN/RNN/Attention $(-\infty, \infty)$	CNN $(-\infty, \infty)$	measurement vector, corresponding to measurement $\{w \mid w \in C^n, \|\|w\|\|_2 = 1\}$
High-level representation	arbitrary real vector $(-\infty, \infty)$	arbitrary real vector $(-\infty, \infty)$	real-valued probability, corresponding to measurement result $(0, 1)$

Experiments

Table 2: Experiment Results in percentage(%). The best performed value (except for CNN/LSTM) for each dataset is in bold.

Model	CR	MPQA	MR	SST	SUBJ	TREC
Uni-TFIDF	79.2	82.4	73.7	-	90.3	85.0
Word2vec	79.8	**88.3**	77.7	79.7	90.9	83.6
FastText [28]	78.9	87.4	76.5	78.8	91.6	81.8
Sent2Vec [42]	79.1	87.2	76.3	80.2	91.2	85.8
CaptionRep [21]	69.3	70.8	61.9	-	77.4	72.2
DictRep [22]	78.7	87.2	76.7	-	90.7	81.0
Ours: QPDN	**81.0**	87.0	**80.1**	**83.9**	**92.7**	**88.2**
CNN [29]	81.5	89.4	81.1	88.1	93.6	92.4
BiLSTM [16]	81.3	88.7	77.5	80.7	89.6	85.2

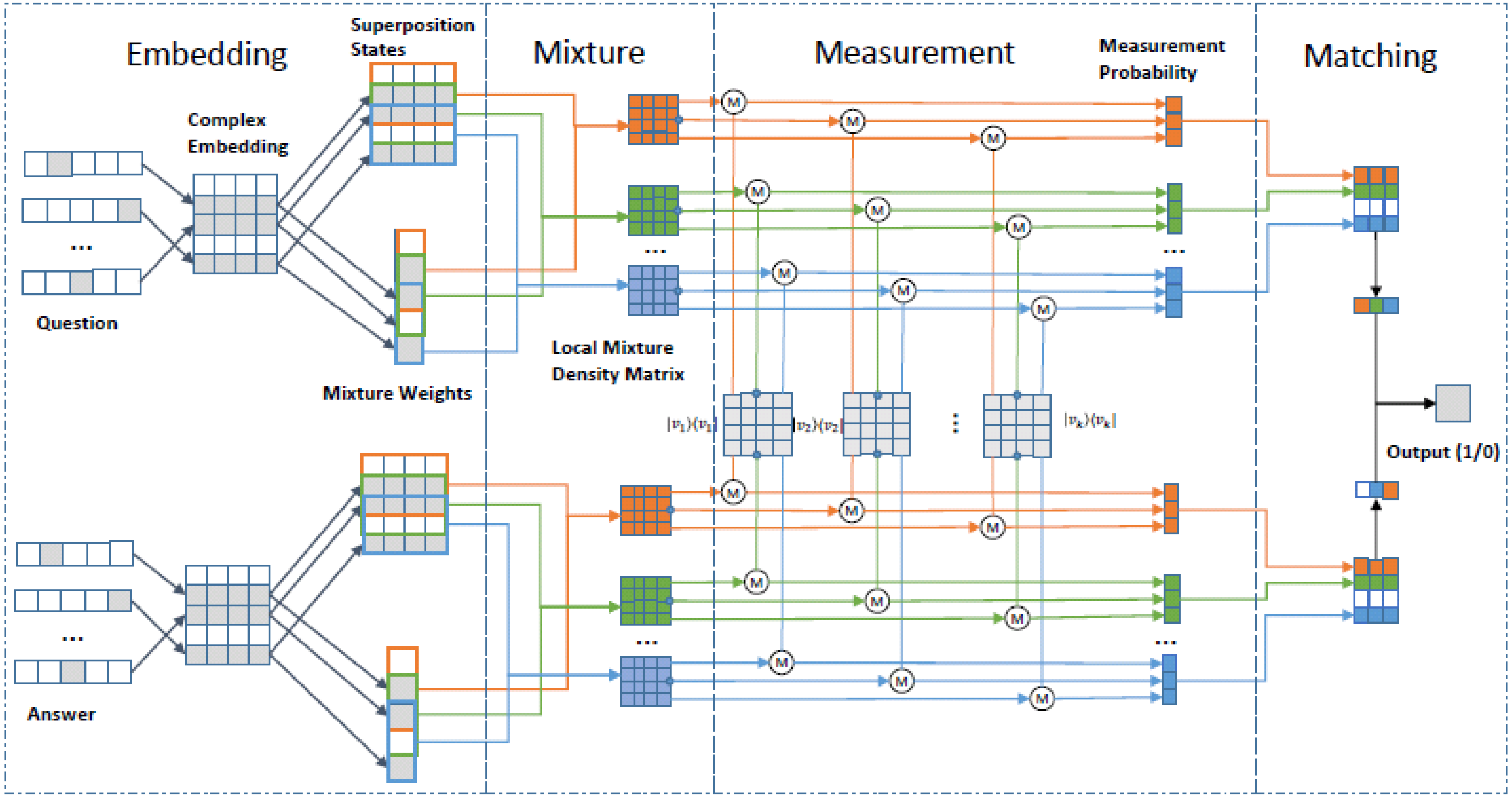
Case study for our measurement

Table 7: The learned measurement for dataset MR. They are selected according to nearest words for a measurement vector in Semantic Hibert Space

Measurement	Selected neighborhood words
1	change, months, upscale, recently, aftermath
2	compelled, promised, conspire, convince, trusting
3	goo, vez, errol, esperanza, ana
4	ice, heal, blessedly, sustains, make
5	continue, warned, preposterousness, adding, falseness

Implements for matching

Figure 1: Architecture of Complex-valued Network for Matching. Ⓜ means a measurement operation according to Eq. 2.



Case study

Table 7: The matching patterns for specific sentence pairs in TREC QA. The darker the color, the bigger weight the word is. The [and] denotes the possible border of the current sliding windows.

Question	Correct Answer
Who is the [president or chief executive of Amtrak] ?	" Long-term success ... " said George Warrington , [Amtrak 's president and chief executive] ."
When [was Florence Nightingale born] ?	,"On May 12 , 1820 , the founder of modern nursing , [Florence Nightingale , was born] in Florence , Italy ."
When [was the IFC established] ?	[IFC was established in] 1956 as a member of the World Bank Group .
[how did women 's role change during the war]	..., the [World Wars started a new era for women 's] opportunities to ....
[Why did the Heaven 's Gate members commit suicide] ?,	This is not just a case of [members of the Heaven 's Gate cult committing suicide] to ...

Experiments

Table 3: Experiment Results on TREC QA Dataset. The best performed values are in bold.

Model	MAP	MRR
Bigram-CNN	0.5476	0.6437
LSTM-3L-BM25	0.7134	0.7913
LSTM-CNN-attn	0.7279	0.8322
aNMM	0.7495	0.8109
MP-CNN	0.7770	0.8360
CNTN	0.7278	0.7831
PWIM	0.7588	0.8219
QLM	0.6780	0.7260
NNQLM-I	0.6791	0.7529
NNQLM-II	0.7589	0.8254
CNM	**0.7701**	**0.8591**
Over NNQLM-II	1.48% ↑	4.08% ↑

Table 4: Experiment Results on Yahoo QA Dataset. The best performed values are in bold.

Model	P@1	MRR
Okapi BM-25	0.2250	0.4927
LSTM	0.4875	0.6829
CNN	0.4125	0.6323
CNTN	0.4654	0.6687
QLM	0.3950	0.6040
NNQLM-I	0.4290	0.6340
NNQLM-II	0.4660	0.6730
CNM	**0.4880**	**0.6845**
Over NNQLM-II	4.72% ↑	1.45% ↑

Table 5: Experiment Results on WikiQA Dataset.The best performed values for each dataset are in bold.

Model	MAP	MRR
Bigram-CNN	0.6190	0.6281
BILSTM	0.6557	0.6695
LSTM-attn	0.6639	**0.6828**
CNN	**0.6701**	0.6822
QLM	0.5120	0.5150
NNQLM-I	0.5462	0.5574
NNQLM-II	0.6496	0.6594
CNM	0.6548	0.6664
Over NNQLM-II	1.01% ↑	1.01% ↑

Weights

Table 6: Selected learned important words in TREC QA. All words are lower.

	Selected words
Important	studio, president, women, philosophy scandinavian, washingtonian, berliner, championship defiance, reporting, adjusted, jarred
Unimportant	71.2, 5.5, 4m, 296036, 3.5 may, be, all, born movements, economists, revenues, computers

Learned measurements

Table 8: Selected learned measurements for TREC QA. They were selected according to nearest words for a measurement vector in Semantic Hilbert Space. All the words are lower.

	Selected neighborhood words for a measurement vector
1	andes, nagoya, inter-american, low-caste, kazakhstan
2	cools, injection, boiling,adrift
3	andrews, paul, manson, bair
4	historically, 19th-century, genetic, hatchback, shipbuilding
5	missile, exile, rebellion, darkness

Ablation Test

Table 9: Ablation Test. The values in parenthesis are the performance difference between the model and CNM.

Setting	MAP	MRR
FastText-MaxPool	0.6659 (0.1042↓)	0.7152 (0.1439↓)
CNM-Real	0.7112 (0.0589↓)	0.7922 (0.0659↓)
CNM-Global-Mixture	0.6968 (0.0733↓)	0.7829 (0.0762↓)
CNM-trace-inner-product	0.6952 (0.0749↓)	0.7688 (0.0903↓)
CNM	0.7701	0.8591

Potential ideas

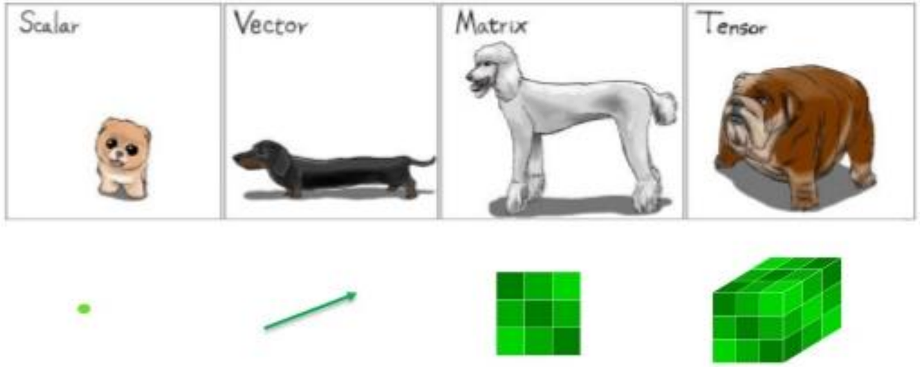
- Representation based on vector space
    - Deep investigation of **complex** vector space
    - Semantic Hilbert vector space for **interpretable** NN like Capsule
    - Overview of word embedding
- Dynamics in vector space
    - Evolved density matrix for language model
    - Dynamic word embedding via tensor decomposition
    - Investigate the dynamics with time-aware multi-turn dialogue

Ideas

- Dynamics for thematic issues
- Evolved Density matrix for language model

Dynamics for thematic issues

- Concatenate the Document-Term or Term-Term Co-occurrence as a Tensor
  - $[M_{t_1}, M_{t_2}, \ldots, M_{t_T}]$ as $T_{t,d,w}$, 3-d Tensor, where M_{t_1} is the D-W matrix.



  - Tensor composition/factorization machine for **time-aware word embedding**
    - *Obtain the neighbor words of "nuclear" in different time stamp.*

Linking embedding with topic/thematic issue

- For a topic, it is usually considered as a distribution of words
  - $p^{(i)} = p(p_{w_1}, p_{w_1}, \dots p_{w_{|v|}})$
- For a word embedding, its neighbor has a well-designed distance, we could also get a distribution as $p_{w_j} = \frac{e^{d_{ij}}}{\sum e^{d_{ij}}}$.
- In a sense, word embedding is considered **lower-level topic**

Text: Character → word → word group → clause → sentence → story

Word embedding

Topic
Thematic issue

Evolved Density matrix for language model

Algorithm 1 Training of Quantum Memory Network

Input: m-dimension word vectors E with size $|V| * m$
A assisted hidden vector h for measurement
A given word sequence $\mathcal{S} = \{\boldsymbol{w_1}, \boldsymbol{w_2}, ...\boldsymbol{w_n}\}$
A initial density matrix ρ_0

```
Initialise ρ = ρ0.
Pretrain embedding and grantee the unit length.
repeat
    for i : n do
        Look up the unit word vector e_{w_i} for word w_i.
        Calculate the bias of the weak measurement by Eq. (1).
        Update the new density matrix by Eq. (2).
        Back propagation by loss shown in Eq. (3).
    end for
until Traversal all the tokens in current sentence.
```

1. $\alpha = \langle h | e_{w_i} \rangle^2$
2. $\rho^{(t)} = U\rho^{(t-1)}U^* * \alpha + |e_{w_i}\rangle\langle e_{w_\mathrm{i}}| * (1 - \alpha)$
3. $p_{w_j} = tr(\rho \left|e_{w_j}\right\rangle \left\langle e_{w_\mathrm{j}}\right|)$