Classifying Relations via Long Short Term Memory Networks along Shortest Dependency Paths

Yan Xu, Lili Mou, Ge Li, Yunchuan Chen, Hao Peng, Zhi Jin

Sep 21, 2015

Institute of Software, School of EECS, Peking University

Outline

1. Overview of Relation Extraction
2. Linguistic Phenomenon Related to Relation Extraction
3. Deep Neural Networks
   - Multi-Channel Long Short Term Memory Networks
   - Regularization: Dropout
4. Experiments
5. Conclusions

Information Extraction

A trillion gallons of water have been poured into an empty region of outer space.

Named Entity Recognition

A trillion gallons of **[water]e1** have been poured into an empty **[region]e2** of outer space.

Relation Classification

Entity-Destination (**[water]e1**, **[region]e2**)

SemEval 2010 Task 8 - Dataset

- Evaluation Exercises on Semantic Evaluation - ACL SigLex event
- Tasks 8 – Multi-Way Classification of Semantic Relations Between Pairs of Nominals
- Training data
  - 8000 training samples
    - "The system as described above has its greatest application in an arrayed <e1>configuration</e1> of antenna <e2>elements</e2>."
    - Component-Whole(e2, e1)
- Testing data
  - 2717 testing samples

SemEval 2010 Task 8 - Relations

- (1) Cause-Effect
- (2) Instrument-Agency
- (3) Product-Producer
- (4) Content-Container
- (5) Entity-Origin
- (6) Entity-Destination
- (7) Component-Whole
- (8) Member-Collection
- (9) Message-Topic
- (10) Other

Outline

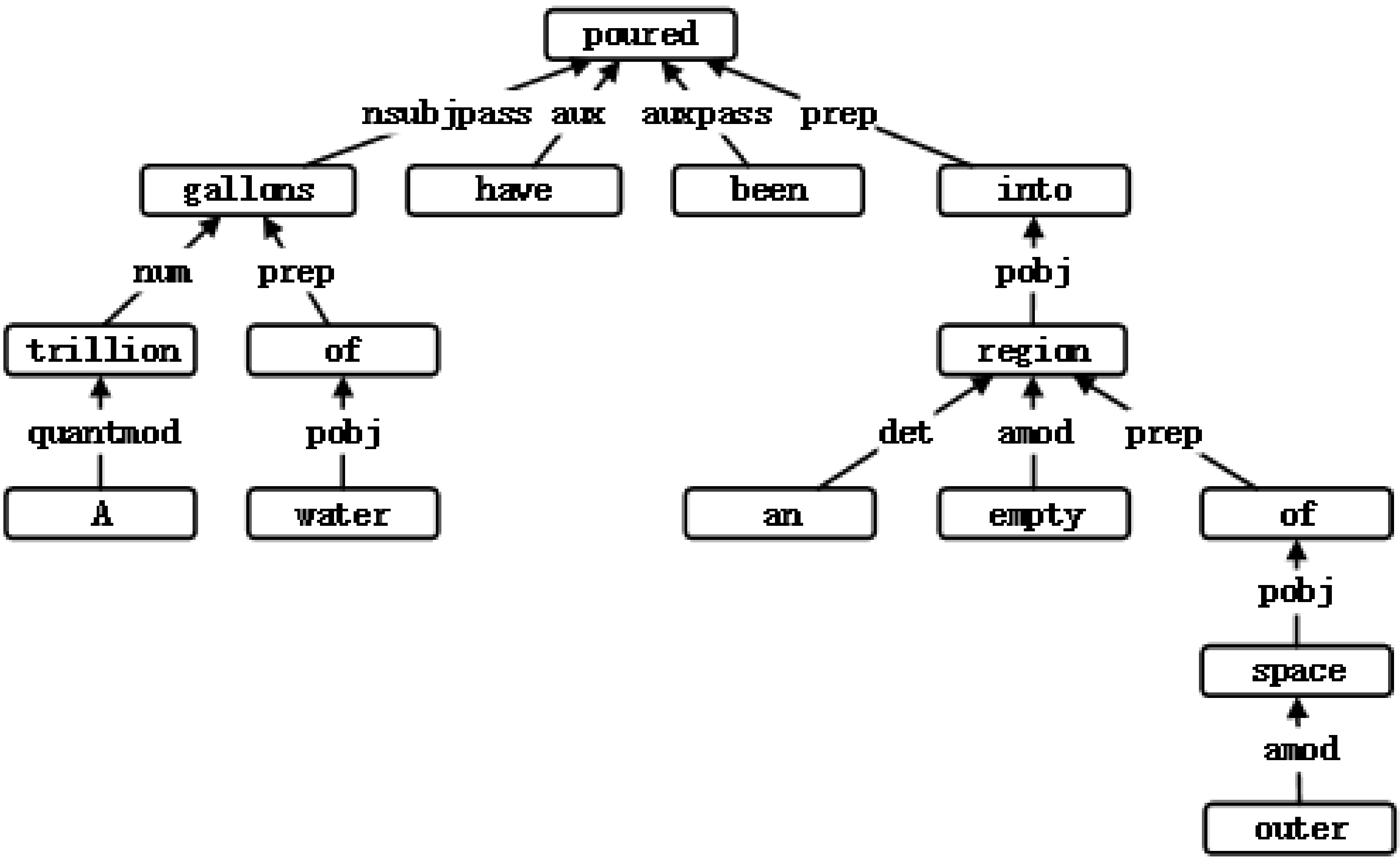
1. Overview of Relation Extraction
2. Linguistic Phenomenon Related to Relation Extraction
3. Deep Neural Networks
   - Multi-Channel Long Short Term Memory Networks
   - Regularization: Dropout
4. Experiments
5. Conclusions

Motivation

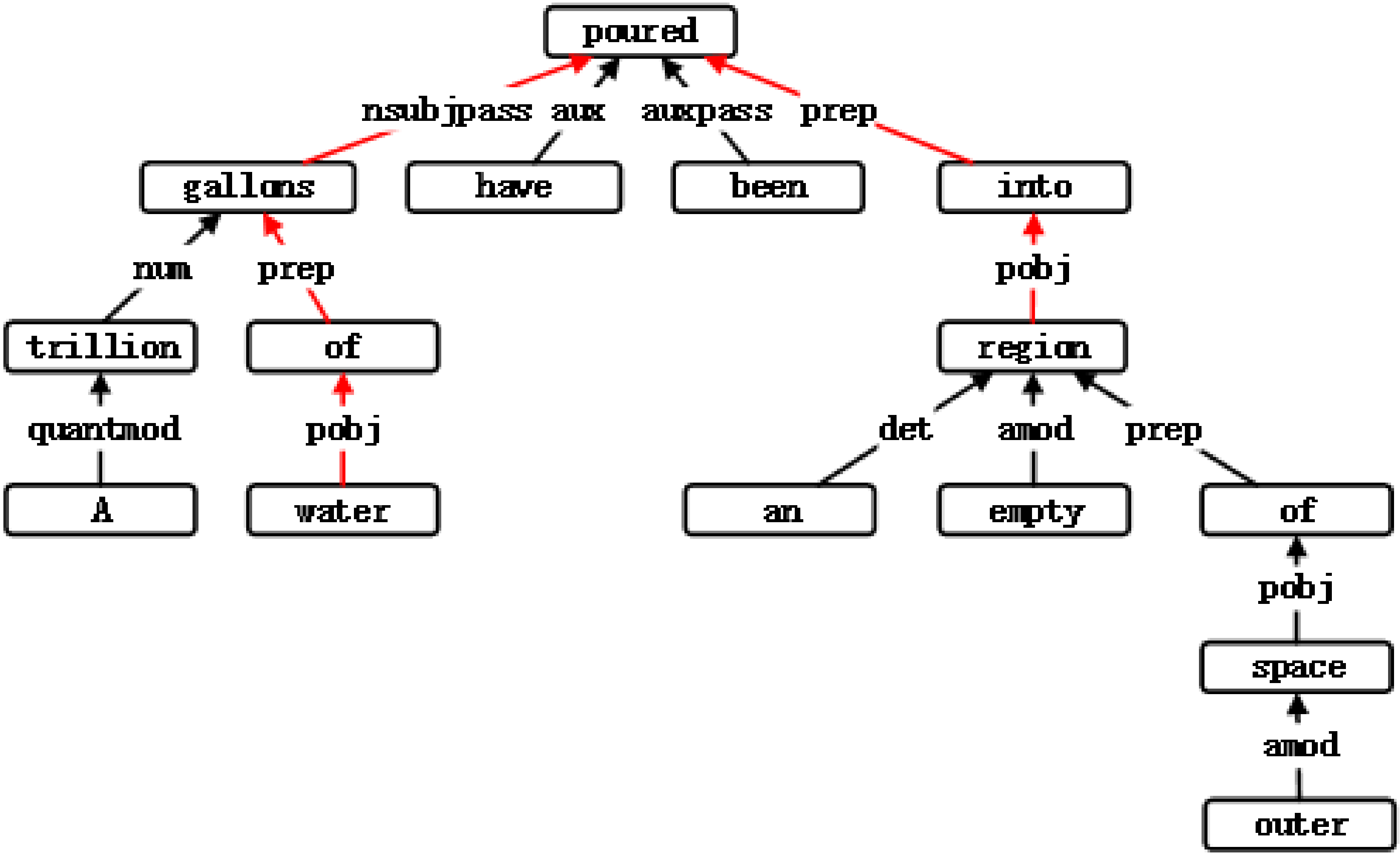
- Which type of **sentence structures** can be more appropriate ?

- Which type of **linguistic information** can be incorporated ?

Dependency Parse Tree



Shortest Dependency Path (SDP)



Directionality

- Dependency trees are a kind of directed graph

- The entities' relation distinguishes its directionality

Subpath1: **[water]e1 → of → gallons → poured**

Subpath2: **poured ← into ← [region]e2**

Info 1: Word Representations

- Word Embeddings
  - Word2vec – map a word to a real-valued vector capturing word's syntactic and semantic information (Mikolov, NIPS' 2013)

word2vec

Tool for computing continuous distributed representations of words.

- Toy example
  - Average embeddings + SVM → nearly **79% F1-score**

Info 2: POS tags

- Ally each word in the path with its POS tag
- Take a coarse-grained POS category heuristically

```
[["NN", "NNS", "NNP", "NNPS"], # noun, proper noun, singular, plural
 ["IN"],  # Preposition or subordinating conjunction
 ["VBN"], # verb, past participle
 ["VBD"], # verb, past tense
 ["VBZ"], # verb,present tense, 3rd person singular
 ["VBG"], # verb, present participle or gerund
 ["VBP"], # verb, present tense, not 3rd person singular
 ["VB"],  # verb, base form
 ["TO"],  # to
 ["JJ", "JJR", "JJS"], # adj
 ["RB", "RBR", "RBS"], # adv
 ["CD"],              # cardinal number
 ["DT", "PDT"],       # determiner
 ["PRP"],             # personal pronoun
 ["RP"]]              # particle
```

Info 3: Grammatical Relations

- A grammatical relation expresses the dependency between a governing word and a dependent word

- Some grammatical relations reflect semantic relation strongly. like “***nsubj***”, “***dboj***”, or “***pobj***”, etc

- In our experiments, grammatical relations are grouped into 19 classes, mainly based on the category proposed by De Marneffe in LREC’ 2006

Info 4: Hypernyms

- With the prior knowledge of hypernyms, we know that "**water** is a kind of **substance**."

- This is a hint that the entities, **water** and **region**, are more of "**Entity-Destination**" relations than other relation like "**Communication-Topic**", "**Cause-Effect**", etc.

- With the help of <u>supersensetagger</u> (Ciaramita, EMNLP' 06)

Outline

1. Overview of Relation Extraction
2. Linguistic Phenomenon of Relation Extraction
3. Deep Neural Networks
   - Multi-Channel Long Short Term Memory Networks
   - Regularization: Dropout
4. Experiments
5. Conclusions

Recurrent Neural Network

- The recurrent neural network is suitable for modeling sequential data by nature.

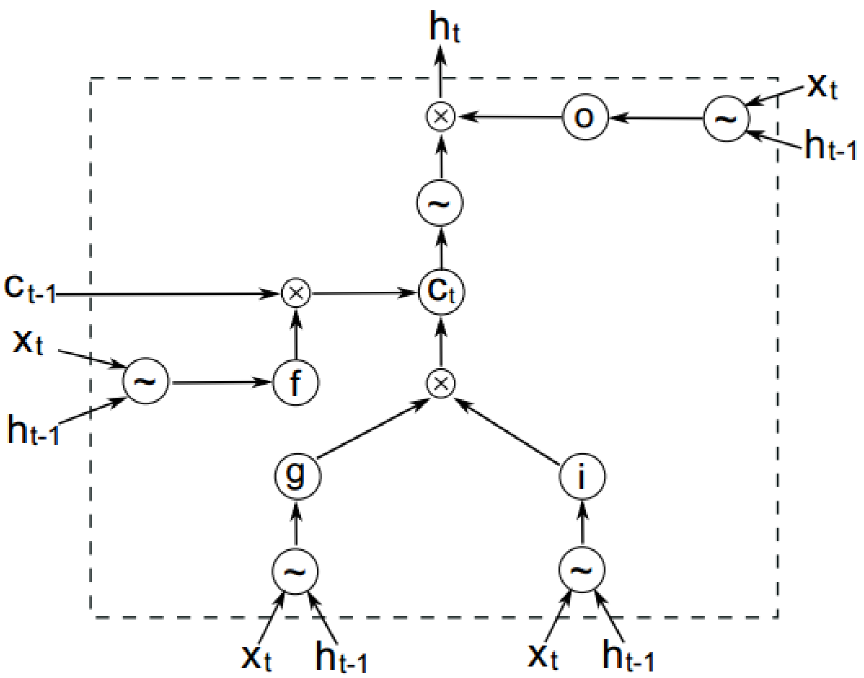
Weight matrices for the input connections

$$\boldsymbol{h}_t = f(W_{in}\boldsymbol{x}_t + W_{rec}\boldsymbol{h}_{t-1} + \boldsymbol{b}_h)$$

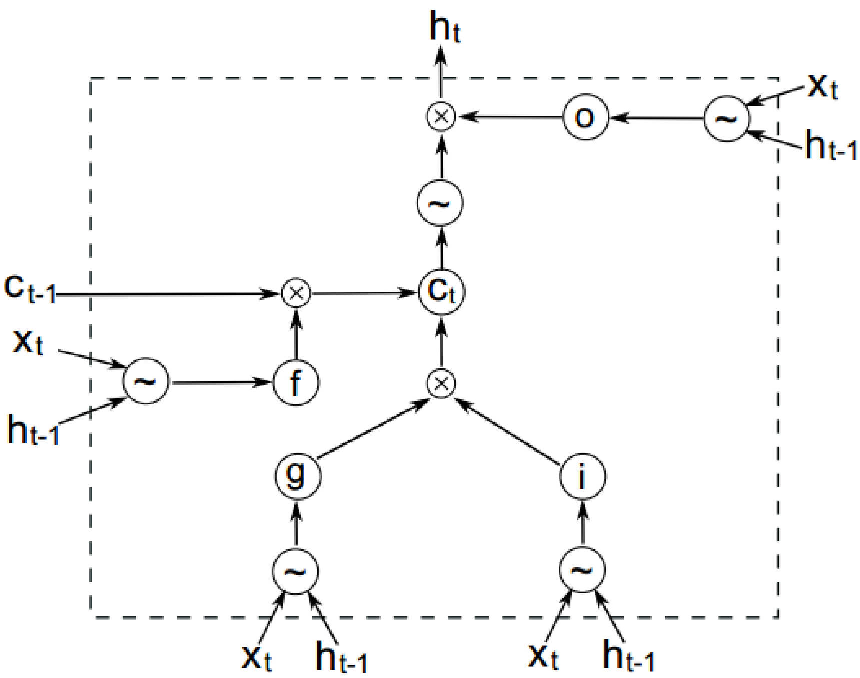
Weight matrices for the recurrent connections

- Gradient vanishing

Long Short Term Memory Networks



Long Short Term Memory Networks



$$\boldsymbol{i}_t = \sigma(W_i \cdot \boldsymbol{x}_t + U_i \cdot \boldsymbol{h}_{t-1} + \boldsymbol{b}_i) \quad (1)$$

$$\boldsymbol{f}_t = \sigma(W_f \cdot \boldsymbol{x}_t + U_f \cdot \boldsymbol{h}_{t-1} + \boldsymbol{b}_f) \quad (2)$$

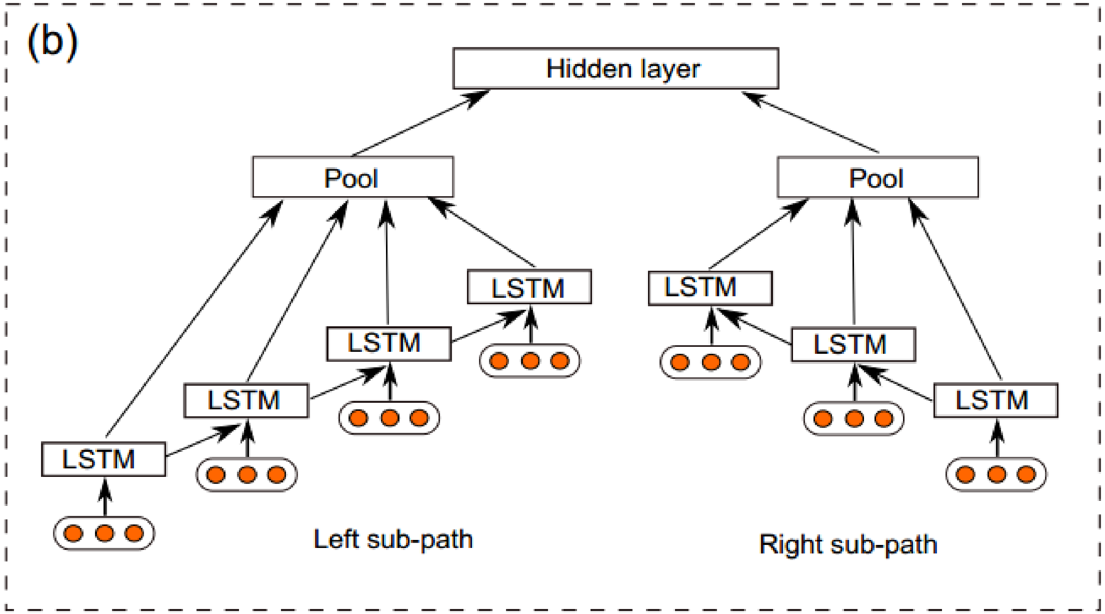
$$\boldsymbol{o}_t = \sigma(W_o \cdot \boldsymbol{x}_t + U_o \cdot \boldsymbol{h}_{t-1} + \boldsymbol{b}_o) \quad (3)$$

$$\boldsymbol{g}_t = \tanh(W_g \cdot \boldsymbol{x}_t + U_g \cdot \boldsymbol{h}_{t-1} + \boldsymbol{b}_g) \quad (4)$$

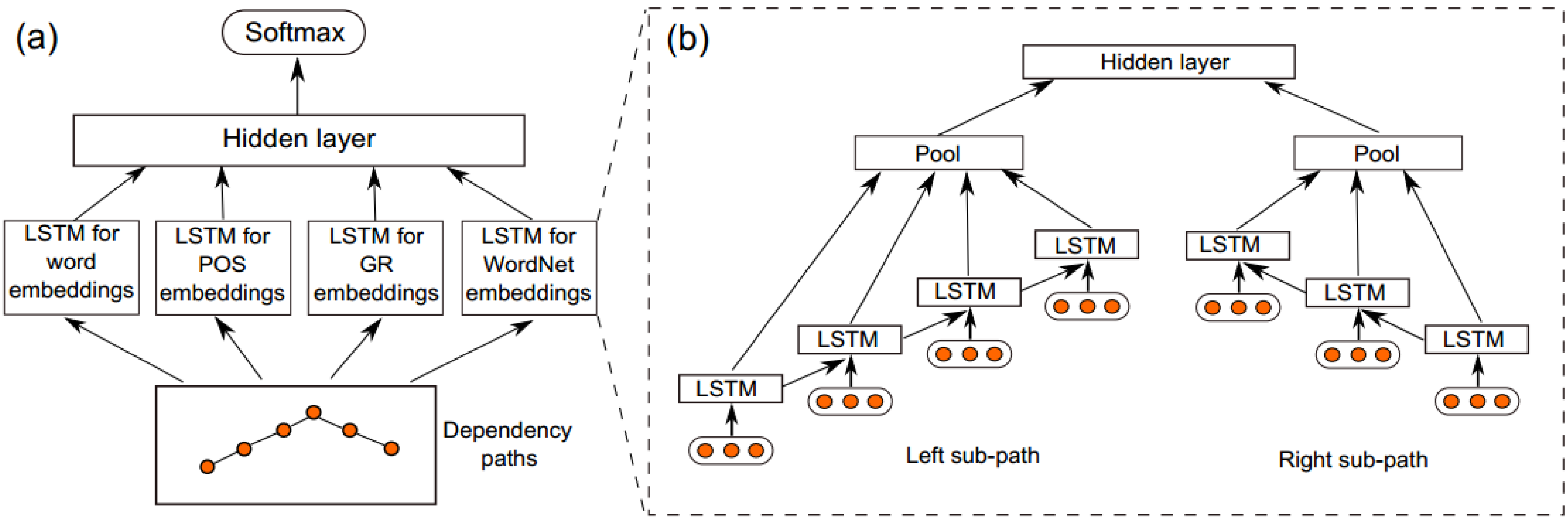
$$\boldsymbol{c}_t = \boldsymbol{i}_t \otimes \boldsymbol{g}_t + \boldsymbol{f}_t \otimes \boldsymbol{c}_{t-1} \quad (5)$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \otimes \tanh(\boldsymbol{c}_t) \quad (6)$$

Framework of SDP-LSTM

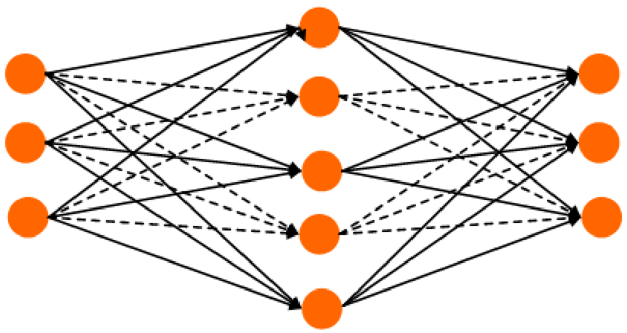


Framework of SDP-LSTM



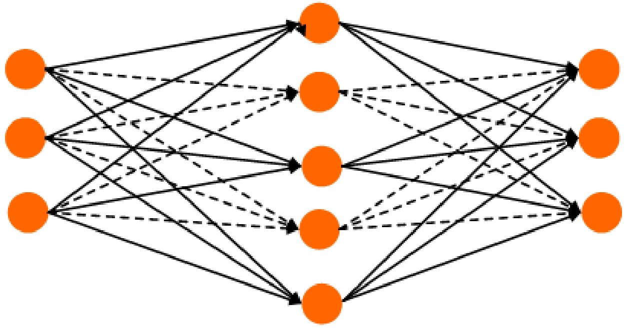
Dropout strategies

- Randomly omitting



Dropout strategies

- Randomly omitting



- Omitting VS Memorizing
  - Dropout different types of neural network layers respectively.

$$\boldsymbol{i}_t = \sigma(W_i \cdot D(\boldsymbol{x}_t) + U_i \cdot \boldsymbol{h}_{t-1} + \boldsymbol{b}_i) \tag{7}$$

$$\boldsymbol{f}_t = \sigma(W_f \cdot D(\boldsymbol{x}_t) + U_f \cdot \boldsymbol{h}_{t-1} + \boldsymbol{b}_f) \tag{8}$$

$$\boldsymbol{o}_t = \sigma(W_o \cdot D(\boldsymbol{x}_t) + U_o \cdot \boldsymbol{h}_{t-1} + \boldsymbol{b}_o) \tag{9}$$

$$\boldsymbol{g}_t = \tanh\Big(W_g \cdot D(\boldsymbol{x}_t) + U_g \cdot \boldsymbol{h}_{t-1} + \boldsymbol{b}_g\Big) \tag{10}$$

Training Objective

- Penalized cross-entropy error

Number of classes

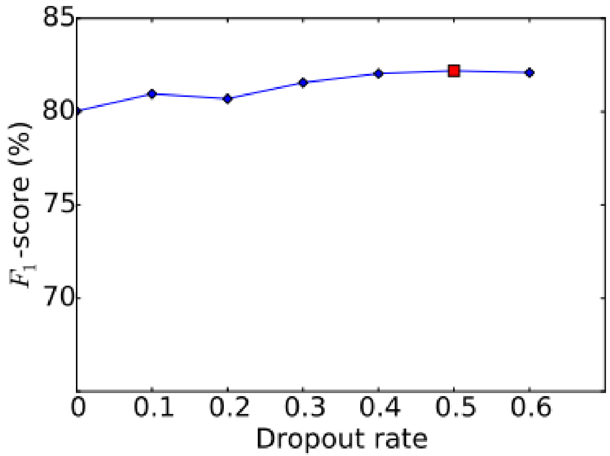
$$J = -\sum_{i=1}^{n_c} t_i \log y_i + \lambda \left(\sum_{i=1}^{\omega} \|W_i\|_F^2 + \sum_{i=1}^{\upsilon} \|U_i\|_F^2 \right)$$

Frobenius norm

Outline

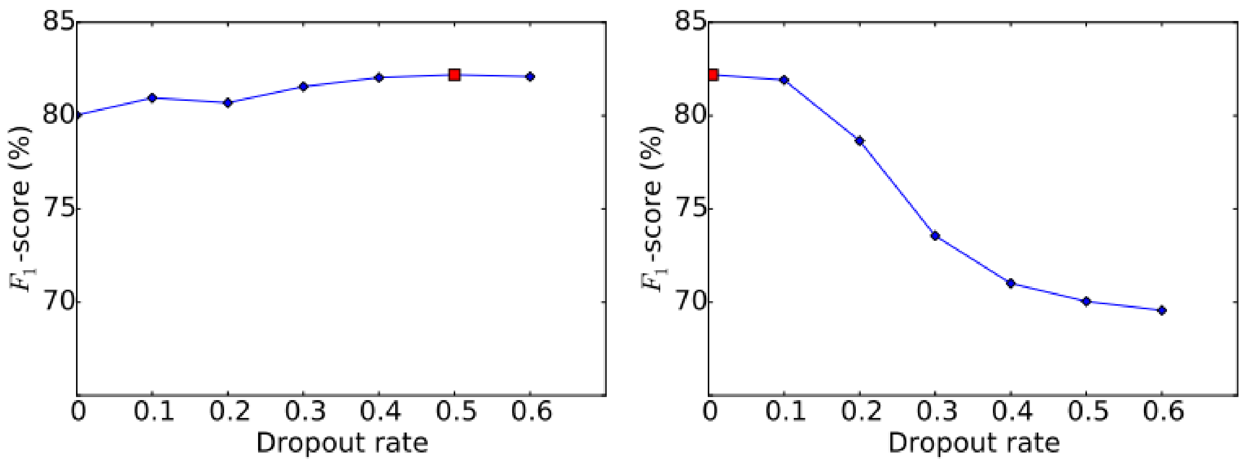
1. Overview of Relation Extraction
2. Linguistic Phenomenon Related to Relation Extraction
3. Deep Neural Networks
   - Multi-Channel Long Short Term Memory Networks
   - Regularization: Dropout
4. Experiments
5. Conclusions

Effect of Dropout Strategies



(a) Dropout word embeddings

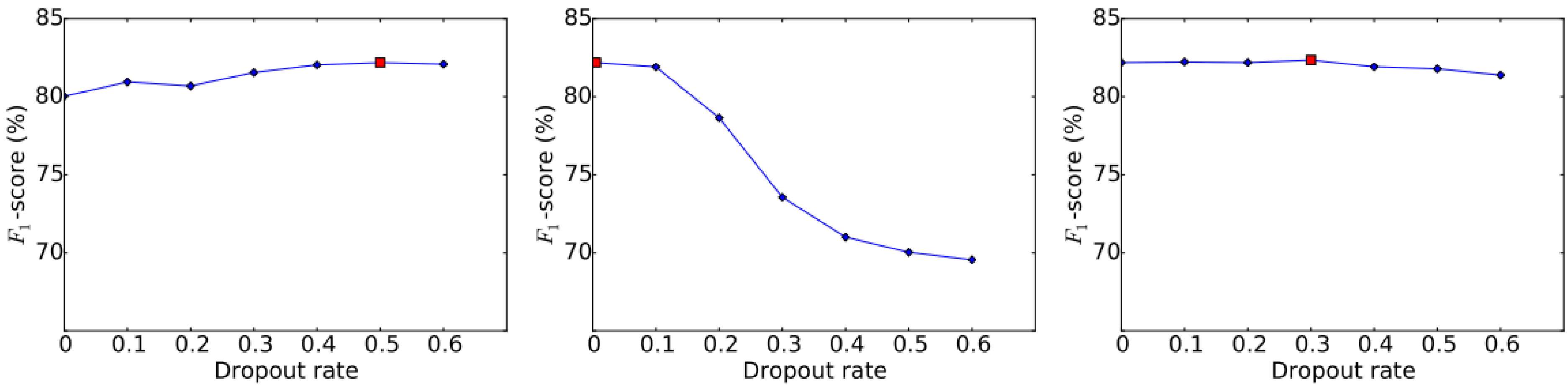
Effect of Dropout Strategies



(a) Dropout word embeddings

(b) Dropout inner cells of memory units

Effect of Dropout Strategies



(a) Dropout word embeddings (b) Dropout inner cells of memory units (c) Dropout the penultimate layer

Effects of Different Channels

- Channel effects

Channels	F_1
Word embeddings	82.35
+ POS embeddings (only)	82.98
+ GR embeddings (only)	83.21
+ WordNet embeddings (only)	83.03
+ POS + GR + WordNet embeddings	83.70

- Traditional recurrent neural network: 82.8%

- LSTM over one path: 82.2%

Comparison

Classifier	Feature set	F_1
SVM	POS, WordNet, prefixes and other morphological features, depdency parse, Levin classes, PropBank, FanmeNet, NomLex-Plus, Google n-gram, paraphrases, TextRunner	82.2
RNN	Word embeddings	74.8
	Word embeddings, POS, NER, WordNet	77.6
MVRNN	Word embeddings	79.1
	Word embeddings, POS, NER, WordNet	82.4
CNN	Word embeddings	69.7
	Word embeddings, word position embeddings, WordNet	82.7
Chain CNN	Word embeddings, POS, NER, WordNet	82.7
FCM	Word embeddings	80.6
	Word embeddings, depedency parsing, NER	83.0
CR-CNN	Word embeddings	82.8†
	Word embeddings, position embeddings	82.7
	Word embeddings, position embeddings	**84.1**†
SDP-LSTM	Word embeddings	82.4
	Word embeddings, POS embeddings, WordNet embeddings, grammar relation embeddings	**83.7**

New Mission Impossible

86 % ?

Outline

1. Overview of Relation Extraction
2. Linguistic Phenomenon Related to Relation Extraction
3. Deep Neural Networks
   - Multi-Channel Long Short Term Memory Networks
   - Regularization: Dropout
4. Experiments
5. Conclusions

Conclusions

- Classifying relation is a challenging task due to the inherent ambiguity of natural languages and the diversity of sentence expression
- The shortest dependency path can be a valuable resource for relation classification
- Treating the shortest dependency path as two sub-paths helps to capture the directionality
- LSTM units are effective in feature detection and propagation along the shortest dependency path

Thanks!

Q&A