TRANSFER LEARNING FOR SEQUENCE TAGGING WITH HIERARCHICAL RECURRENT NETWORKS

Zhilin Yang, Ruslan Salakhutdinov & William W. Cohen

ICLR 2017

YUAN SHUAI

SMILE LAB

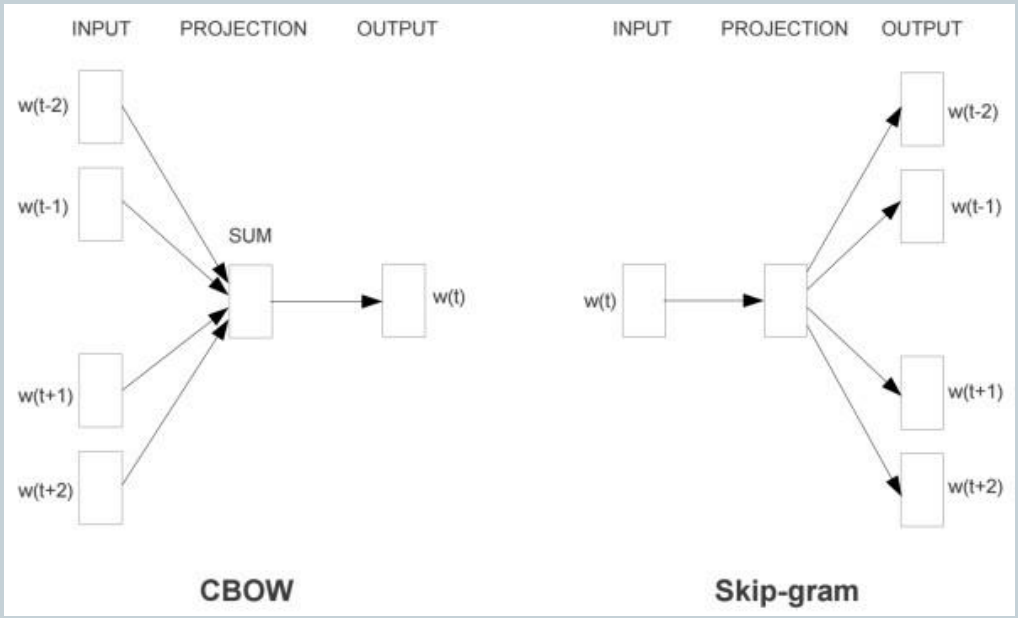
- Background knowledge
- Basic model
- Transfer learning based on the model

Background knowledge

- What is transfer learning?
  - **迁移学习基本概念**
    - **域(Domain)**：由数据特征和特征分布组成，是学习的主体
      - Source domain (源域)：已有知识的域
      - Target domain (目标域)：要进行学习的域
    - **任务(Task)**：由目标函数和学习结果组成，是学习的结果
  - **形式化**
    - 条件：给定一个源域 $\mathcal{D}_S$ 和源域上的学习任务 $\mathcal{T}_S$ ，目标域 $\mathcal{D}_T$ 和目标域上的学习任务 $\mathcal{T}_T$
    - 目标：利用 $\mathcal{D}_S$ 和 $\mathcal{T}_S$ 学习在目标域上的预测函数 $f(\cdot)$ 。
    - 限制条件：$\mathcal{D}_S \neq \mathcal{D}_T$ 或 $\mathcal{T}_S \neq \mathcal{T}_T$

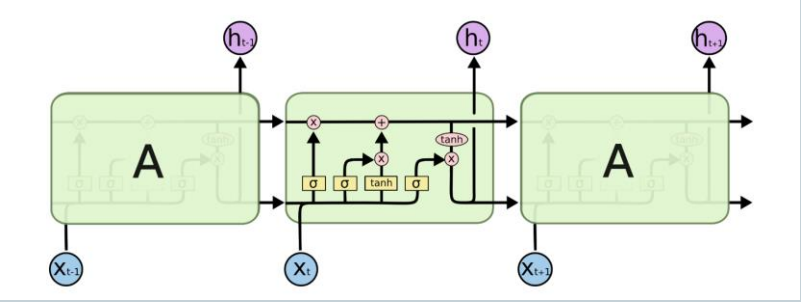
Word Embedding

- Mapping words or phrases from the vocabulary to vectors of real numbers.

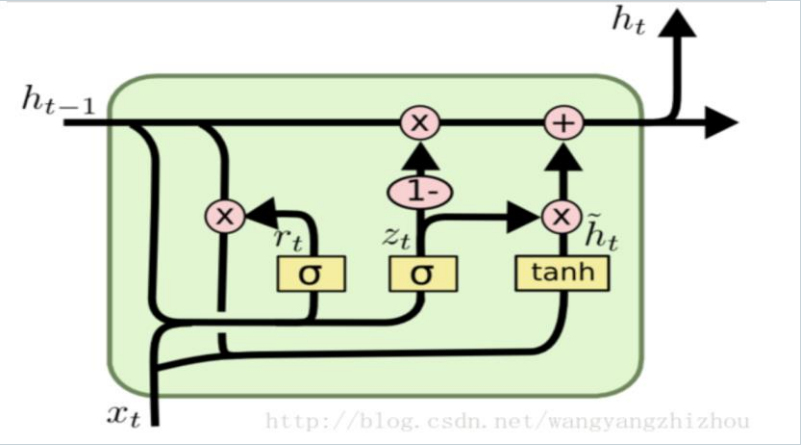


https://link.zhihu.com/?target=http%3A//mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/

LSTM & GRU



Forget gate
Input gate
Output gate



Reset gate
Update gate

http://colah.github.io/posts/2015-08-Understanding-LSTMs/

CRF

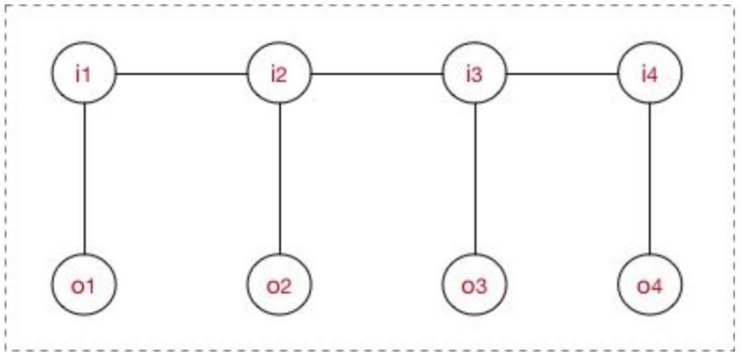
- In this topic, we only focus on Linear Chain CRFs
- A framework for building probabilistic models to segment and label sequence data
- Sequence tagging: POS tagging, text chunking , NER(named entity recognition)

Conditional random fields: Probabilistic models for segmenting and labeling sequence data

http://blog.echen.me/2012/01/03/introduction-to-conditional-random-fields/

CRF

Definition. *Let* $G = (V, E)$ *be a graph such that* $\mathbf{Y} = (\mathbf{Y}_v)_{v \in V}$, *so that* $\mathbf{Y}$ *is indexed by the vertices of* G. *Then* $(\mathbf{X}, \mathbf{Y})$ *is a conditional random field in case, when conditioned on* $\mathbf{X}$, *the random variables* $\mathbf{Y}_v$ *obey the Markov property with respect to the graph:* $p(\mathbf{Y}_v \mid \mathbf{X}, \mathbf{Y}_w, w \neq v) = p(\mathbf{Y}_v \mid \mathbf{X}, \mathbf{Y}_w, w \sim v)$, *where* $w \sim v$ *means that* w *and* v *are neighbors in* G.



Hammersley & Clifford Theorem:

$$P(Y) = \frac{1}{Z} \prod_C \Psi_C(Y_C)$$

$$Z = \sum_Y \prod_C \Psi_C(Y_C)$$

Linear Chain CRF:

$$P(I|O) = \frac{1}{Z(O)} e^{\sum_i^T \sum_k^M \lambda_k f_k(O, I_{i-1}, I_i, i)} = \frac{1}{Z(O)} e^{[\sum_i^T \sum_j^J \lambda_j t_j(O, I_{i-1}, I_i, i) + \sum_i^T \sum_l^L \mu_l s_l(O, I_i, i)]}$$

CRF

- Part-of-Speech Tagging:

In POS tagging, the goal is to label a sentence (a sequence of words or tokens) with tags like ADJECTIVE, NOUN, PREPOSITION, VERB, ADVERB, ARTICLE.

For example

"Bob drank coffee at Starbucks" -> "Bob (n.) drank (v.) coffee (n.) at (prep.) Starbucks (n.)"

[n. v. n. prep. n.] is called label list l.

Another one may be [n. v. v. prep. n.]

We only want the best one, so we need feature functions to score each label list.

Feature Functions in a CRF

In a CRF, each **feature function** is a function that takes in as input:

- a sentence s
- the position i of a word in the sentence
- the label l_i of the current word
- the label l_{i-1} of the previous word

and outputs a real-valued number (though the numbers are often just either 0 or 1).

Some examples:

$f_1(s, i, l_i, l_{i-1}) = 1$ if l_i = ADVERB and the ith word ends in "-ly"; 0 otherwise.

$f_2(s, i, l_i, l_{i-1}) = 1$ if $i = 1$, l_i = VERB, and the sentence ends in a question mark;

$f_4(s, i, l_i, l_{i-1}) = 1$ if l_{i-1} = PREPOSITION and l_i = PREPOSITION.

Features to Probabilities

Next, assign each feature function f_j a weight λ_j (I'll talk below about how to learn these weights from the data). Given a sentence s, we can now score a labeling l of s by adding up the weighted features over all words in the sentence:

$$score(l|s) = \sum_{j=1}^{m} \sum_{i=1}^{n} \lambda_j f_j(s, i, l_i, l_{i-1})$$

(The first sum runs over each feature function j, and the inner sum runs over each position i of the sentence.)

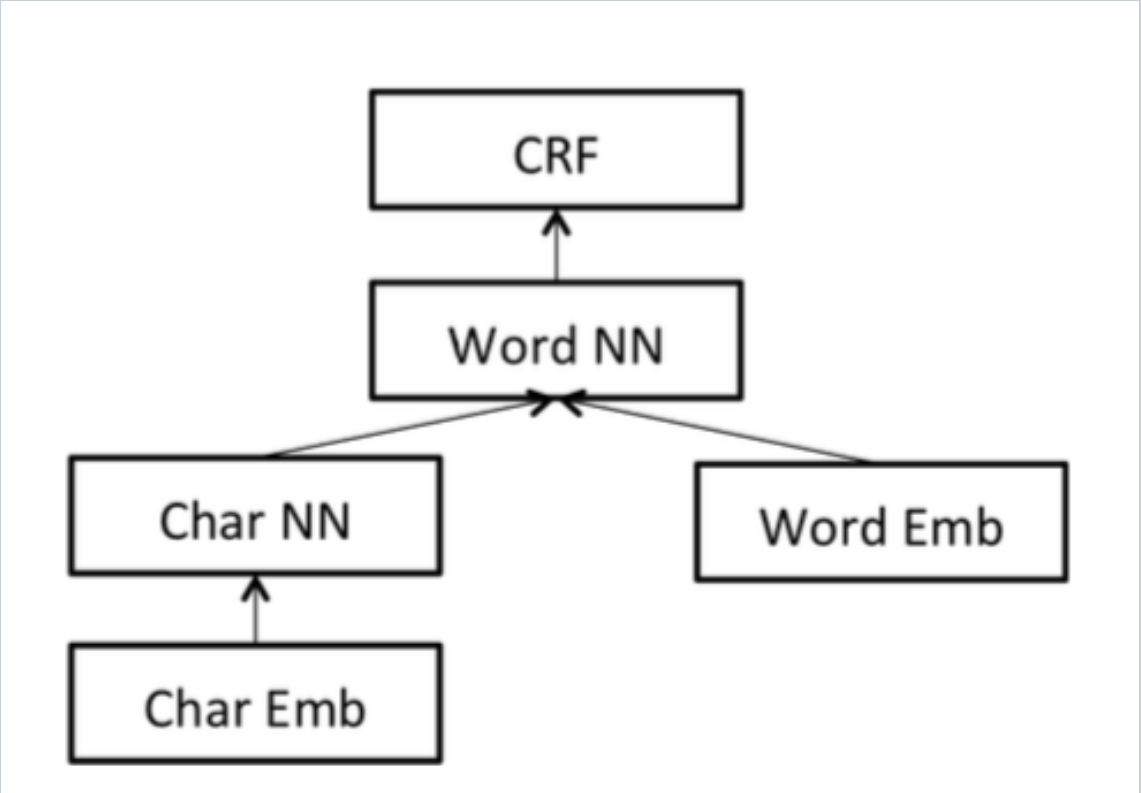
Finally, we can transform these scores into probabilities $p(l|s)$ between 0 and 1 by exponentiating and normalizing:

$$p(l|s) = \frac{exp[score(l|s)]}{\sum_{l'} exp[score(l'|s)]} = \frac{exp[\sum_{j=1}^{m}\sum_{i=1}^{n}\lambda_j f_j(s,i,l_i,l_{i-1})]}{\sum_{l'} exp[\sum_{j=1}^{m}\sum_{i=1}^{n}\lambda_j f_j(s,i,l'_i,l'_{i-1})]}$$

Smells like Logistic Regression?

Basic model

Embedding + Bi-LSTM + CRF

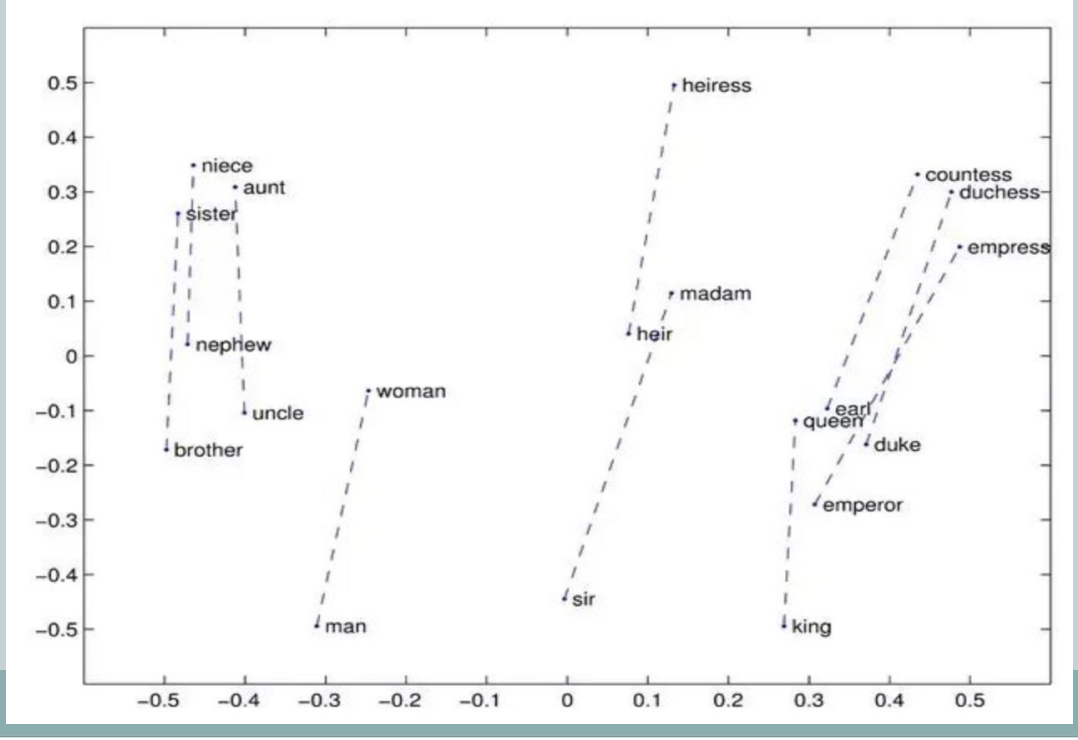


This model is so prevailing.

Why embedding?

- To replace traditional one-hot inputs of CRF with word vectors which include more sematic information

e.g. v(queen)-v(king)= v(woman)-v(man)



Why embedding in character level?

- Some proper nouns have no trained word embeddings.

- Introducing morphological information

  e.g. capital letter - ‘Apple’

- Extracting more sematic information

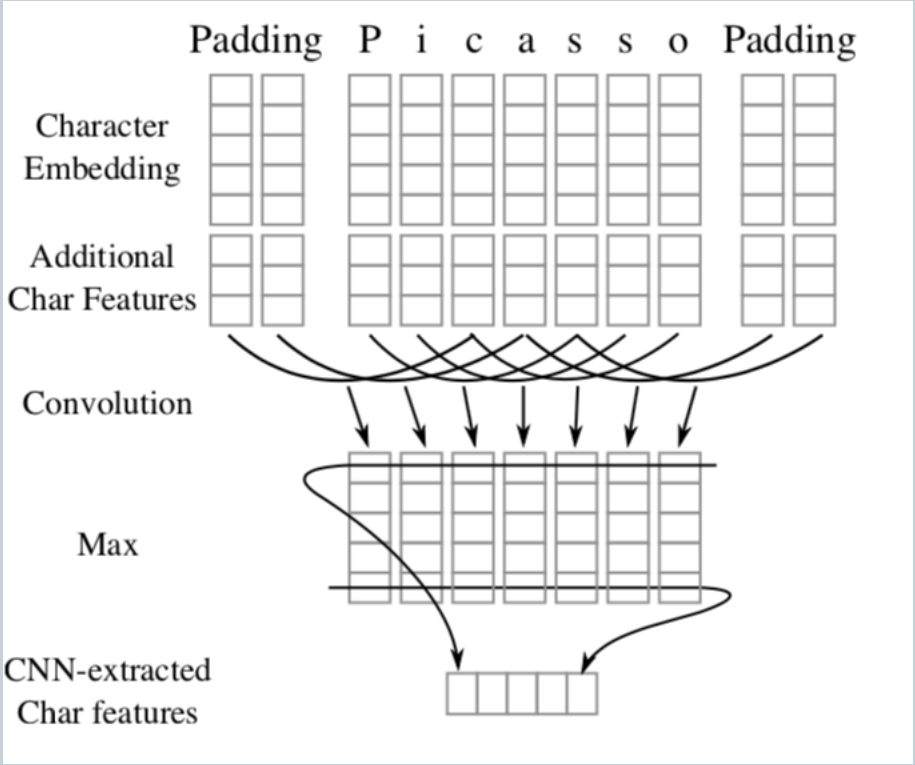
  e.g. Prefix – ‘re-’-> ‘back’

  Suffix – ‘-ed’ -> past tenses

  Root

How embedding？

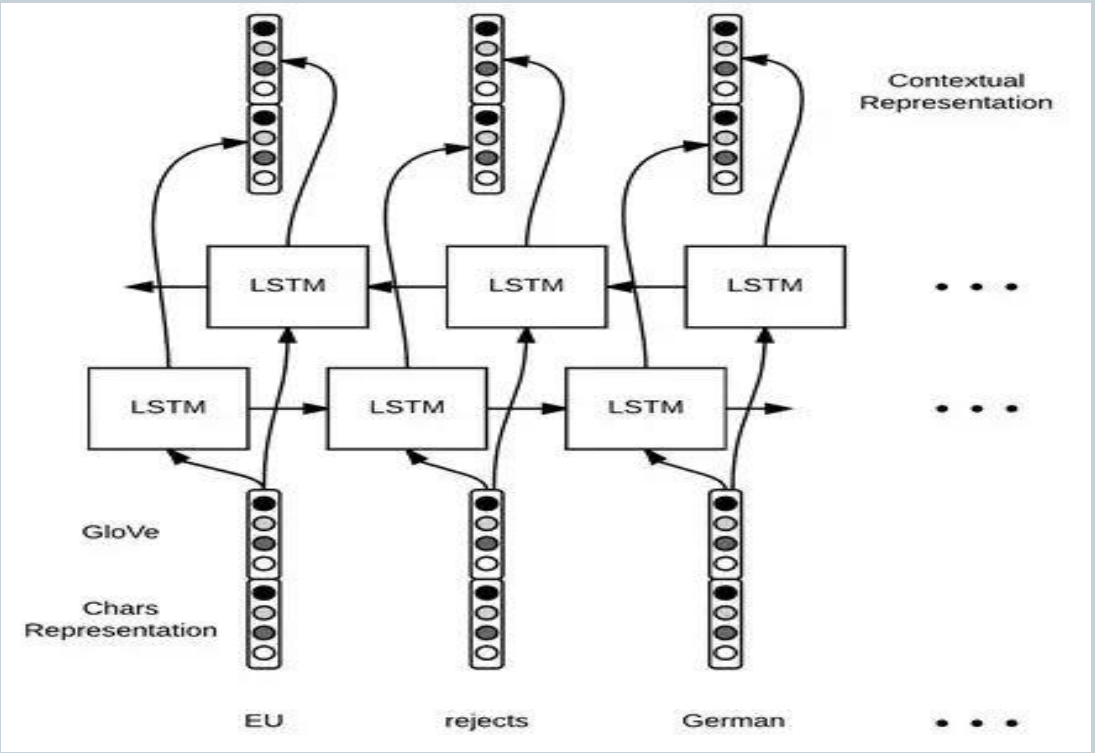
- Both of the word-level layer and the character-level layer can be implemented as convolutional neural networks (CNNs) or recurrent neural networks (RNNs)(e.g. GRU/LSTM)

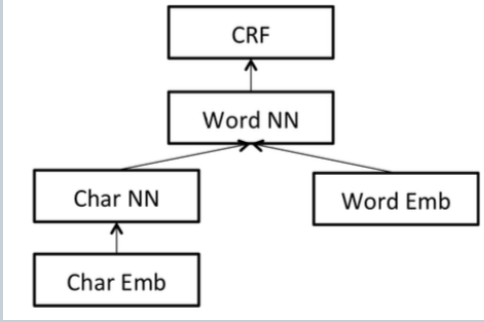


"Named Entity Recognition with Bidirectional LSTM-CNNs"

Word NN and Char NN

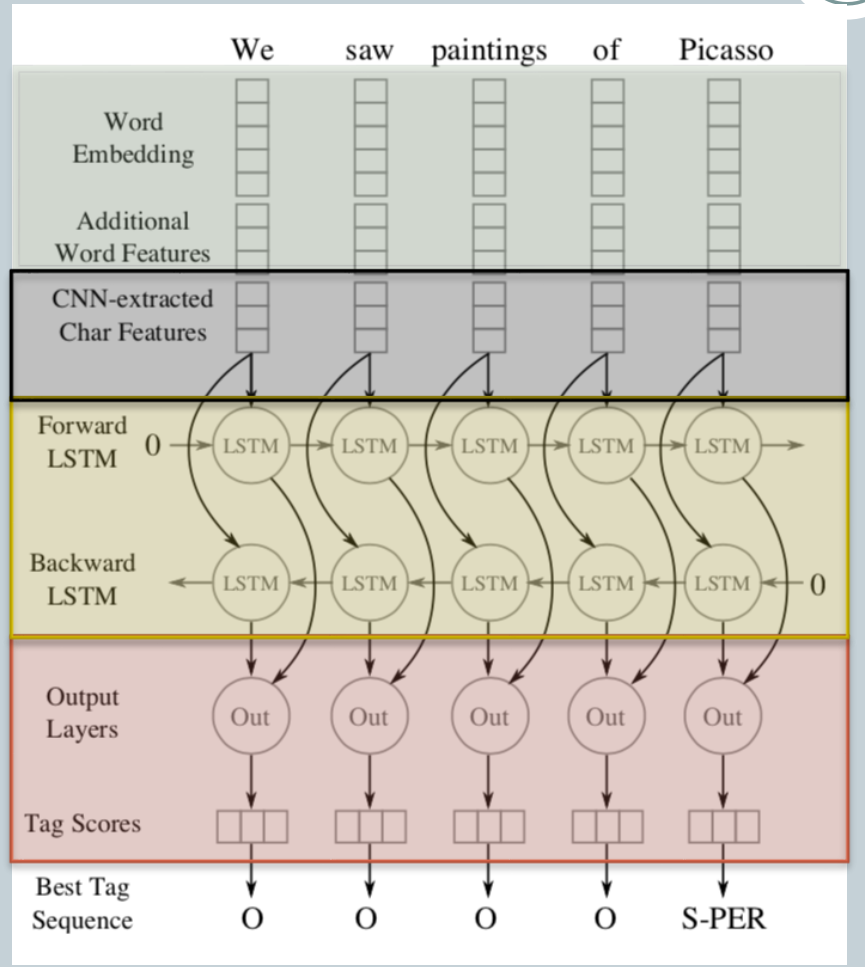
- Aiming to gain contextual information
- Usually using Bi-LSTM(why?)

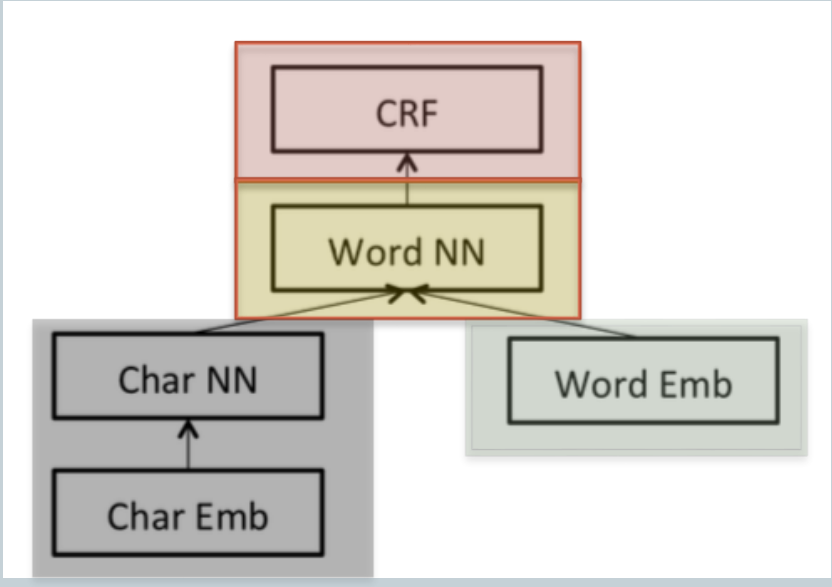




Input: embedding vector
Output: contextual representation

Details

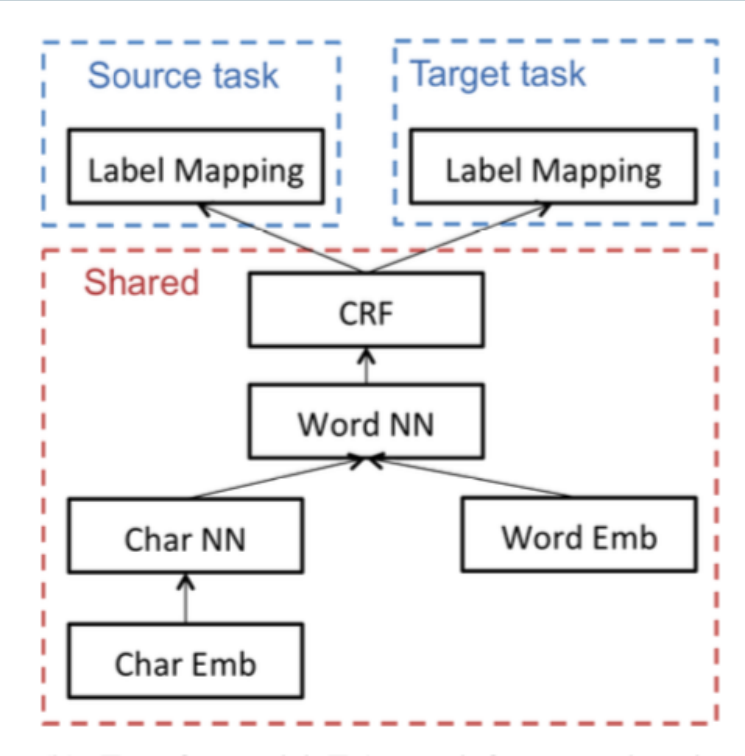

We
saw
paintings
of
Picasso
Word Embedding
Additional Word Features
CNN-extracted Char Features
Forward LSTM
0
LSTM
Backward LSTM
0
Output Layers
Out
Tag Scores
Best Tag Sequence
O
O
O
O
S-PER


CRF
Word NN
Char NN
Char Emb
Word Emb

Transfer models

T-A:used for cross-domain transfer where label mapping is possible.

e.g. POS tags in the Genia biomedical corpus & Penn Treebank tags (Barrett & Weber-Jahnke, 2014)



In this situation, we share all the model parameters and feature representation in the neural networks, including the word and character embedding.

Transfer models

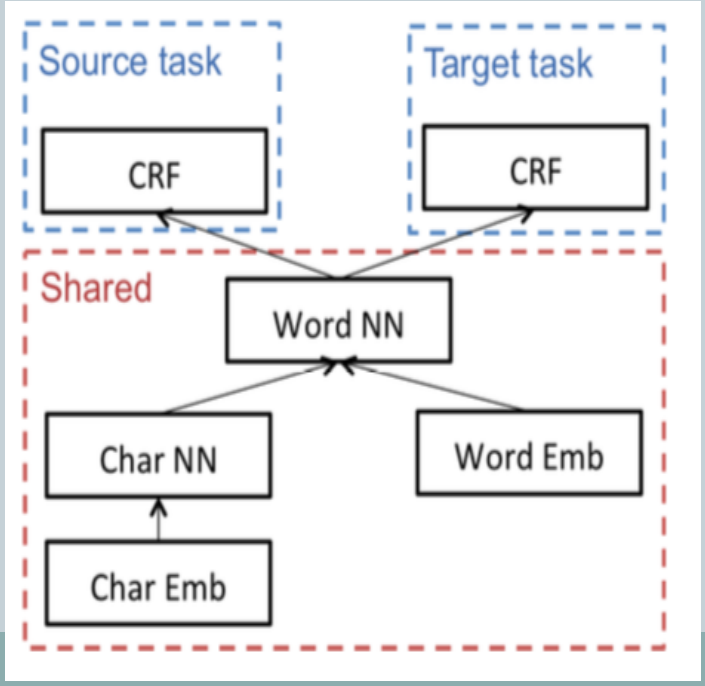
T-B: used for cross-domain transfer with disparate label sets, and cross-application transfer.

e.g. Cross-domain transfer with disparate label sets:

POS tags in the Genia biomedical corpus & POS tags in Twitter (e.g., “URL”)

Cross- application transfer:

POS tagging, chunking and named entity recognition

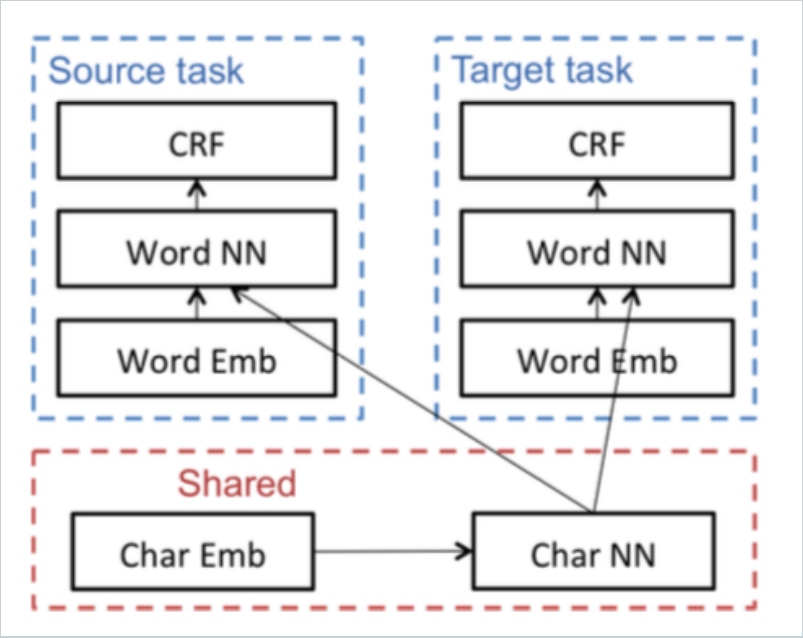


In this situation,we untie the parameter sharing in the CRF layer—i.e., each task learns a separate CRF layer.

Transfer models

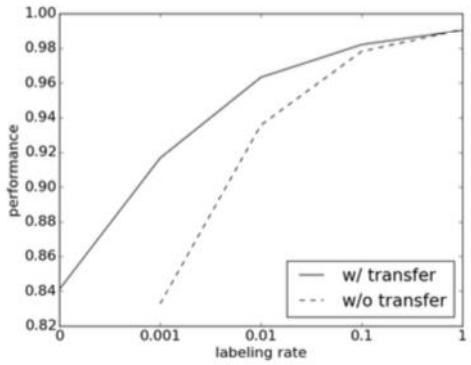
T-C：used for cross-lingual transfer.

It is very difficult for transfer learning between languages with disparate alphabets (e.g., English and Chinese)

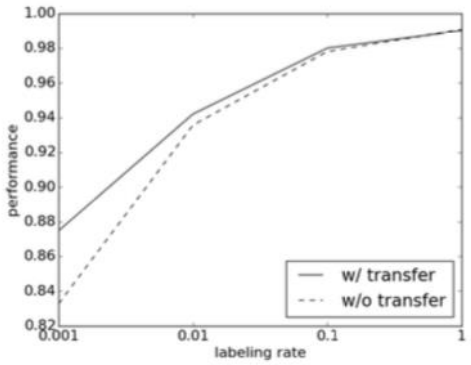


In this situation, we share the character embeddings and the character-level layer between different languages for transfer learning,.

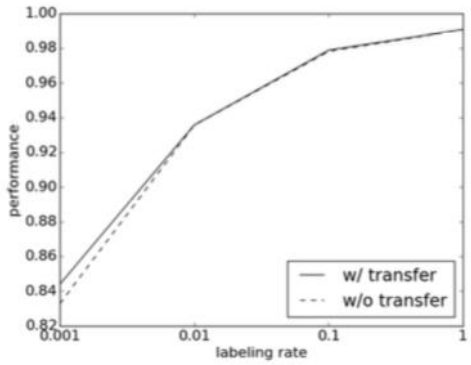
Experiment



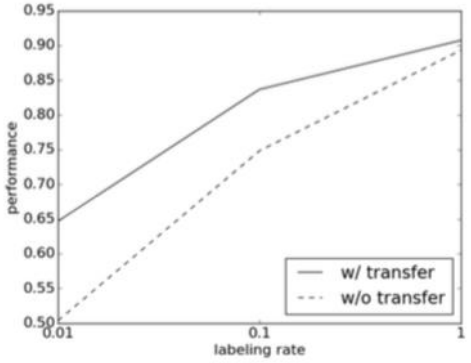
(a) Transfer from PTB to Genia.



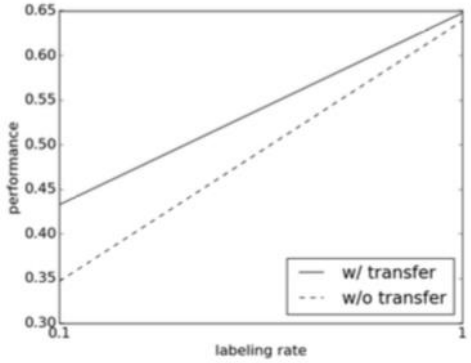
(b) Transfer from CoNLL 2003 NER to Genia.



(c) Transfer from Spanish NER to Genia.

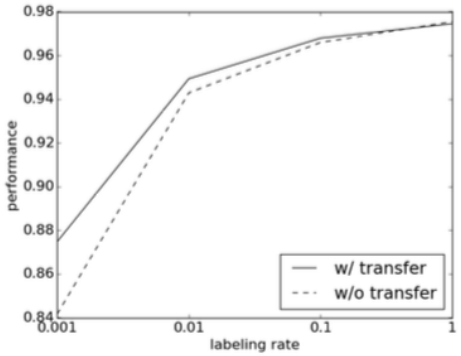


(d) Transfer from PTB to Twitter POS tagging.

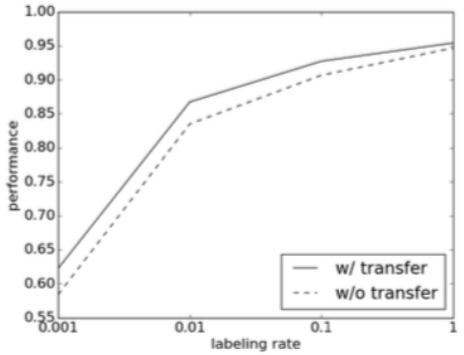


(e) Transfer from CoNLL 2003 to Twitter NER.

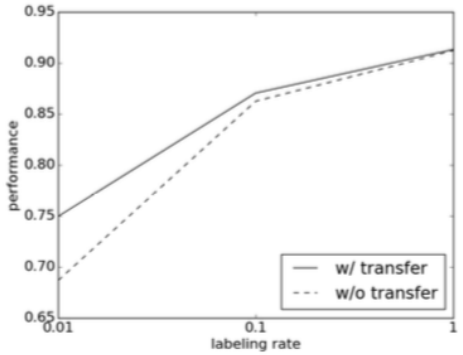
Experiment



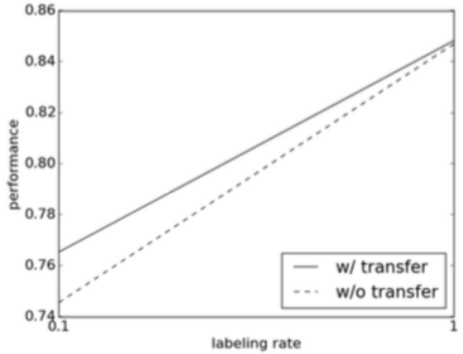
(f) Transfer from CoNLL 2003 NER to PTB POS tagging.



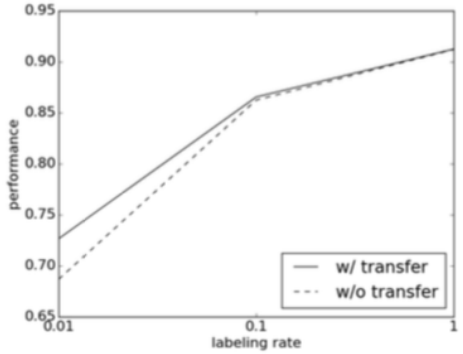
(g) Transfer from PTB POS tagging to CoNLL 2000 chunking.



(h) Transfer from PTB POS tagging to CoNLL 2003 NER.



(i) Transfer from CoNLL 2003 English NER to Spanish NER.



(j) Transfer from Spanish NER to CoNLL 2003 English NER.

Experiment

Table 1: Dataset statistics.

Benchmark	Task	Language	# Training Tokens	# Dev Tokens	# Test Tokens
PTB 2003	POS Tagging	English	912,344	131,768	129,654
CoNLL 2000	Chunking	English	211,727	-	47,377
CoNLL 2003	NER	English	204,567	51,578	46,666
CoNLL 2002	NER	Dutch	202,931	37,761	68,994
CoNLL 2002	NER	Spanish	207,484	51,645	52,098
Genia	POS Tagging	English	400,658	50,525	49,761
Twitter	POS Tagging	English	12,196	1,362	1,627
Twitter	NER	English	36,936	4,612	4,921

Experiment

Table 2: Improvements with transfer learning under multiple low-resource settings (%). "Dom", "app", and "ling" denote cross-domain, cross-application, and cross-lingual transfer settings respectively. The numbers following the slashes are labeling rates (chosen such that the number of labeled examples are of the same scale).

Source	Target	Model	Setting	Transfer	No Transfer	Delta
PTB	Twitter/0.1	T-A	dom	83.65	74.80	8.85
CoNLL03	Twitter/0.1	T-A	dom	43.24	34.65	8.59
PTB	CoNLL03/0.01	T-B	app	74.92	68.64	6.28
PTB	CoNLL00/0.01	T-B	app	86.73	83.49	3.24
CoNLL03	PTB/0.001	T-B	app	87.47	84.16	3.31
Spanish	CoNLL03/0.01	T-C	ling	72.61	68.64	3.97
CoNLL03	Spanish/0.01	T-C	ling	60.43	59.84	0.59
PTB	Genia/0.001	T-A	dom	92.62	83.26	9.36
CoNLL03	Genia/0.001	T-B	dom&app	87.47	83.26	4.21
Spanish	Genia/0.001	T-C	dom&app&ling	84.39	83.26	1.13
PTB	Genia/0.001	T-B	dom	89.77	83.26	6.51
PTB	Genia/0.001	T-C	dom	84.65	83.26	1.39

Experiment

Table 3: Comparison with state-of-the-art results (%).

Model	CoNLL 2000	CoNLL 2003	Spanish	Dutch	PTB 2003
Collobert et al. (2011)	94.32	89.59	–	–	97.29
Passos et al. (2014)	–	90.90	–	–	–
Luo et al. (2015)	–	91.2	–	–	–
Huang et al. (2015)	94.46	90.10	–	–	97.55
Gillick et al. (2015)	–	86.50	82.95	82.84	–
Ling et al. (2015)	–	–	–	–	**97.78**
Lample et al. (2016)	–	90.94	85.75	81.74	–
Ma & Hovy (2016)	–	91.21	–	–	97.55
Ours w/o transfer	94.66	91.20	84.69	85.00	97.55
Ours w/ transfer	**95.41**	**91.26**	**85.77**	**85.19**	97.55

Experiment Conclusion

- Factors are crucial for the performance transfer learning approach:

  a) label abundance for the target task,

  b) relatedness between the source and target tasks

  c) the number of parameters that can be shared.

Conclusions

Contributions:

1) Achieving significant improvement on various datasets under low-resource conditions, as well as new state-of-the-art results on some of the benchmarks

2) Proposing three neural network architectures for the settings of cross-domain, cross-application, and cross-lingual transfer and drawing some valuable experiment conclusions.

Shortages:

1) It is not clear why choosing GRU for embedding, instead of LSTM.

2) Only discussing model-based transfer

Improvements:

1)A mixed structure of LSTM and GRU

2)Combining model-based transfer with resource-based transfer for cross-lingual transfer learning.

Thanks!