NLPCC-ICCPOL 2016

大连理工大学 DALIAN UNIVERSITY OF TECHNOLOGY

Combing Word Embedding & Semantic Lexicon for Chinese Word Similarity Computation

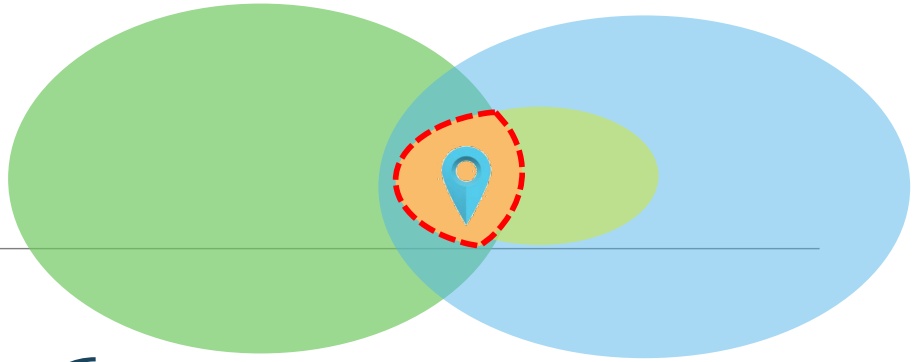
Jiahuan Pei, Cong Zhang,
Degen Huang(✉), Jianjun Ma

Dalian University of Technology

Outline

- **Introduction**
  - Chinese Word Similarity Computation
- **Methodology**
  - Cilin-based Word Similarity Computation
  - Embedding-based Word Similarity Computation
  - Combination Strategies
  - Posterior Improvements
- **Experiments**
- **Results & Analysis**
- **Conclusion**

Introduction

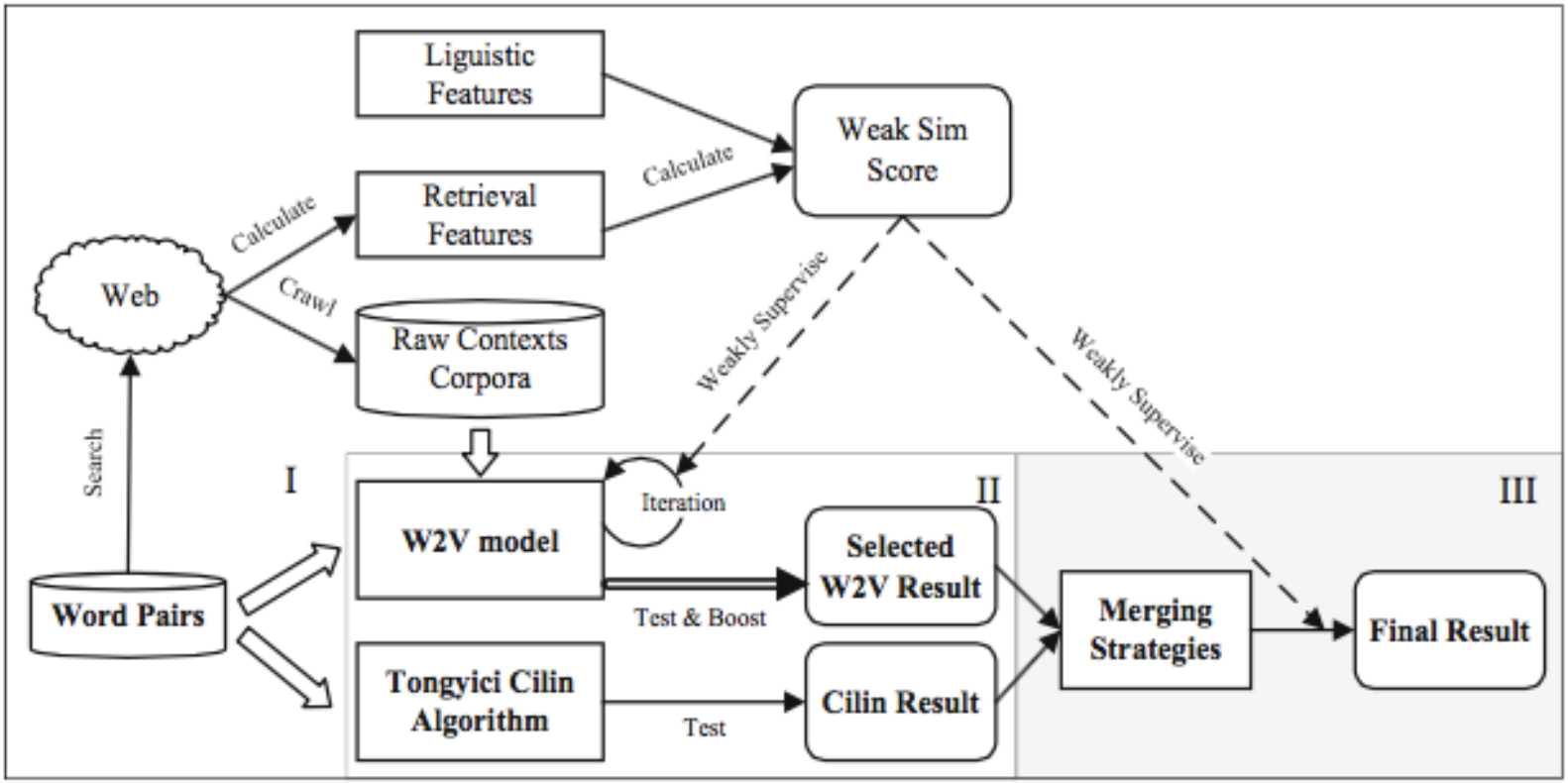


- **Motivation**
  - Chinese Word Similarity Computation
- **Methods Comparisons**
  - Lexicon-based Methods
  - Corpus-based Methods
    - Embedding-based Methods
  - Our focus
- **Drawbacks of basic Lexicon-based methods**
  - Limited to manual semantic resources
  - Both members must be presented in the lexicons
- **Drawbacks of basic embedding-based methods**
  - Semantic similarity OR Conceptual association ?
  - Synonyms OR Antonyms ?
  - Indistinguishable polysemy phenomenon
  - Lack of contexts for the single-character word (only for Chinese)

  Caused by *distributional hypothesis*
- **Ensemble methods !**

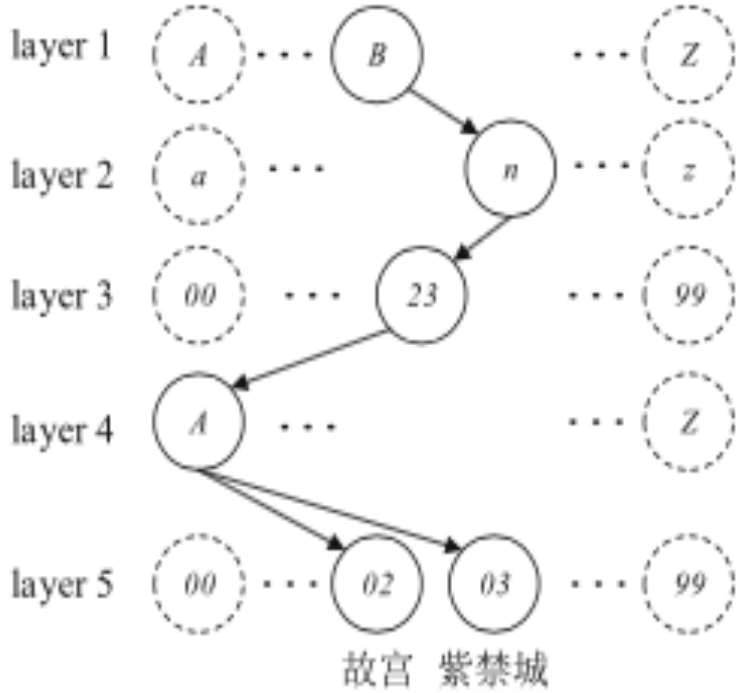
Introduction

- **The general architecture**



Methodology-I

- **Cilin-based Word Similarity Computation**



$$SIM_{cilin}(w_a, w_b) = \max_{1 \leqslant i \leqslant A; 1 \leqslant j \leqslant B} \{sim(c_a^i, c_b^j)\}$$

$$sim(c_a^i, c_b^j) = \begin{cases} 1.0, & if\ code(c_a^i) = code(c_b^j)\ \&\ end\ with\ \text{“=”} \\ \lambda_5, & elif\ code(c_a^i) = code(c_b^j)\ \&\ end\ with\ \text{“\#”} \\ \lambda_6, & elif\ c_a^i, c_b^j\ not\ in\ the\ same\ tree \\ \lambda_{l-1} \times \cos(n_{l-1} \times \frac{\pi}{180}) \times (\frac{n_{l-1}-k+1}{n_{l-1}}), & else \end{cases}$$

linear transformation: $f(x) = 9x + 1$

Methodology-II

- **Embedding-based Word Similarity Computation**
  - **Basic skip-gram model**

$$Q = \frac{1}{N}\sum_{n=1}^{N}\sum_{-c\leq j\leq c, j\neq 0} \log p(w_{n+j}|w_n)$$

$$p(w_{n+j}|w_n) = \frac{\exp(\mathbf{w}_{n+j}^{(2)} \cdot \mathbf{w}_t^{(1)})}{\sum_{k=1}^{V}\exp(\mathbf{w}_k^{(2)} \cdot \mathbf{w}_t^{(1)})}$$

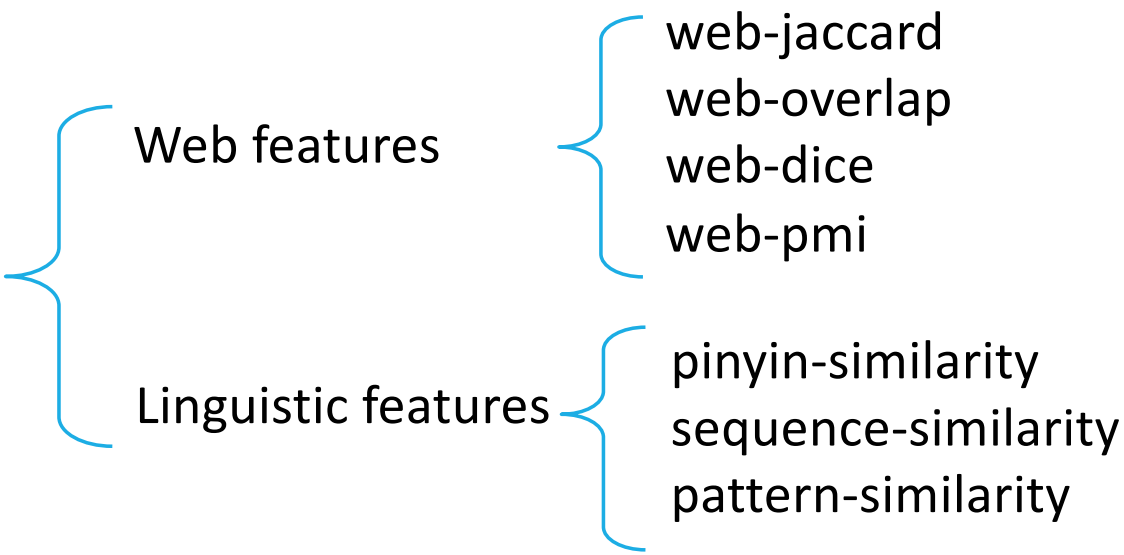
  - **Our Improved Model**

$$J = \max_{1\leqslant iter\leqslant I}\{\Phi_{iter}(\overrightarrow{S_{pred}}, \overrightarrow{S_{wss}})\}$$

Hypothesis – Not only the contexts of the words, but also similarity- related retrieval statistics and linguistic features can reflect the degree of the word similarity.

Methodology-II

- **Embedding-based Word Similarity Computation**
  - **Computation of WSS**



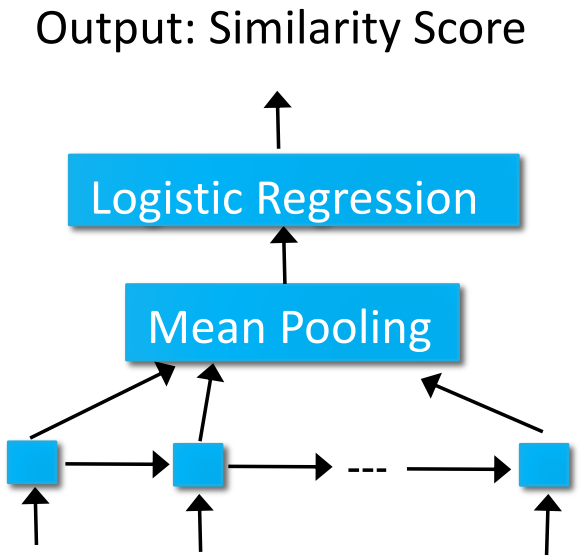
Methodology-III

- **Combination Strategies**
  - *Sc* : result of cilin-based method
  - *Sv* : result of embedding-based method
  - *Sm*: merged score

Max	$S_m = \min\{S_c, S_v\}$
Min	$S_m = \min\{S_c, S_v\}$
Replace 1	$S_m = \begin{cases} S_c, S_c \neq 1 \\ S_v, S_c = 1 \end{cases}$
Replace 1 and 10	$S_m = \begin{cases} S_c, S_c \neq 1 \wedge S_c \neq 10 \\ S_v, S_c = 1 \vee S_c = 10 \end{cases}$
Arithmetic Mean	$S_m = (S_c + S_v)/2$
Geometric Mean	$S_m = \sqrt{S_c * S_v}$

Methodology-IV

- **Posteriori Improvements**
  - Improving Embedding Model by Machine Translation
    1. Translation
    2. Spelling checking and length filtering
    3. Translated embedding replacement
  - Refitting by Sequence Learning via LSTM



Experiments

- **Data Set**
  - Benchmark dataset from NLPCC-ICCPOL 2016 CLSC shared task
- **Evaluation**
  - Spearman correlation coefficient

$$\rho = 1 - \frac{6\sum_{i=1}^{n}(R_{Xi} - R_{Yi})^2}{n(n^2-1)}$$

  - Pearson correlation coefficient

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

Results and Analysis

Table 1. Results of W2V models based on different corpora

No.	Corpora	ρ	r	ρ'	r'
1	Xieso (62M)	0.205	0.203	0.249	0.230
2	Datatang (199M)	0.267	0.272	0.337	0.343
3	News (381M)	0.311	0.305	0.317	0.277
4	News + Xieso	**0.311**	**0.311**	**0.359**	**0.310**
5	Wiki (1.1G)	0.211	0.213	0.324	0.343
6	News + Xieso + Wiki	0.178	0.197	0.221	0.220
7	News + Xieso + Datatang	0.174	0.190	0.211	0.207
8	News + Xieso + DataTang + Wiki	0.214	0.239	0.314	0.308

Results and Analysis

Table 2. Comparison between the original and weakly supervised W2V models

No.	Strategy	ρ	r	ρ'	r'
1	Original W2V	0.296	0.241	0.330	0.262
2	Weakly supervised W2V	**0.311**	**0.311**	**0.359**	**0.310**

Results and Analysis

Table 3. Merging results based on 6 strategies

No.	Strategy	ρ	r	ρ'	r'
1	Replace 1 and 10	0.104	0.090	0.096	0.074
2	Replace 1	0.457	0.446	0.258	0.223
3	Min	0.301	0.314	0.288	0.296
4	Max	0.469	0.464	0.290	0.254
5	Arithmetic mean	0.457	0.455	**0.335**	**0.306**
6	Geometric mean	**0.478**	**0.468**	0.326	0.285

Results and Analysis

Table 4. Comparison between single and merging models

No.	Strategy	ρ	r
1	Cilin	0.405	0.393
2	W2V	0.311	0.311
3	Cilin + W2V	**0.457**	**0.455**

Table 5. Result of improvement by translation and LSTM networks

No.	Strategy	ρ	r
1	Baseline	0.457	0.455
2	Baseline + Translation	0.531	0.476
3	Baseline + Translation + LSTM	**0.541**	**0.514**

Conclusion

- **Our work**
  - A novel framework for the CLSC task
- **Tricks**
  - Dictionary-based similarity scores and corpora extension
  - Translation to improve the Chinese embedding with an English one
  - WSS generated from retrieval and manual features for weak supervision
  - Learning the words co-occurrence sentences via LSTM
- **Final performance**

Thanks for attention!