A Hybrid Text Normalization System using Multi-head Self-attention for Mandarin

Junhui Zhang, Junjie Pan, Xiang Yin, Chen Li, Shichao Liu, Yang Zhang, Yuxuan Wang, Zejun Ma

How can this paper be helpful?

Can it only be applied to Mandarin?

- Nope.

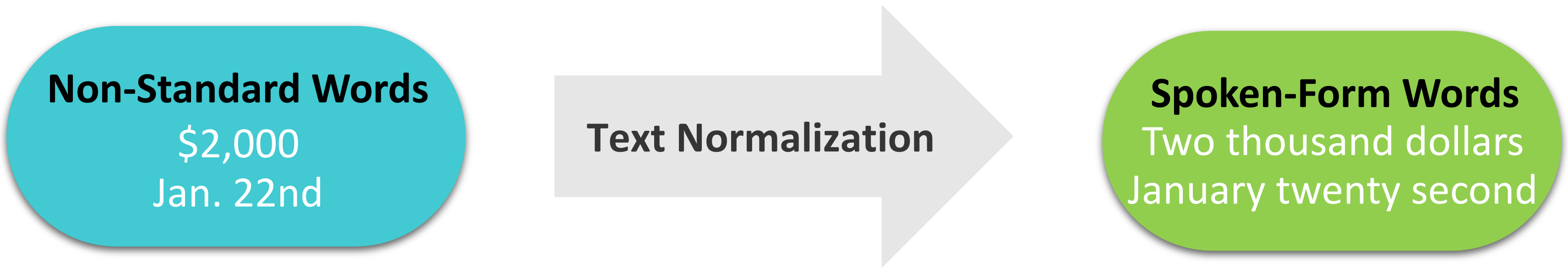
To what languages could it be helpful?

- Any language with a rule-based Text Normalization system.

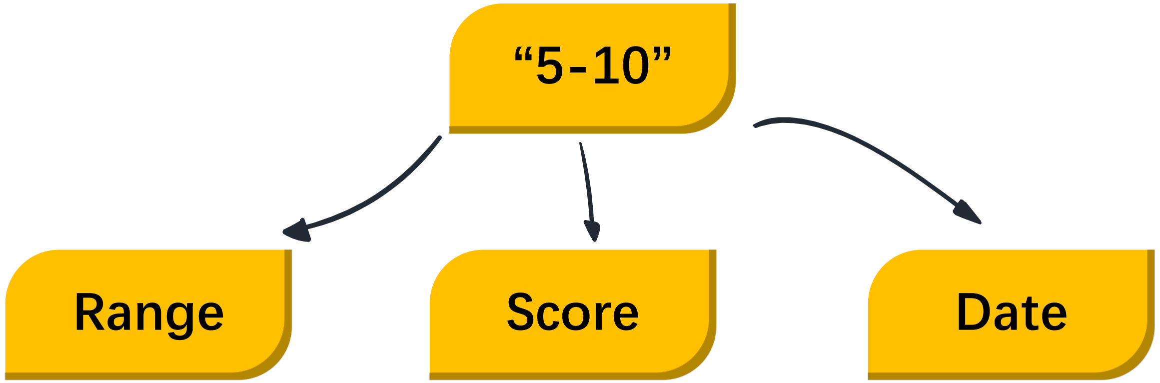
Goal for this paper?

- Improve the performance of a rule-based model.
- Combine system flexibility and model generalizability.

Text Normalization (TN)



Challenge: Ambiguous Cases



Rule-based TN System

Match non-standard words with rules

- Regular Expressions
- Keywords
- Priority

Pros:

- Flexible (add new rules easily)
- Highly developed (handle various cases)

Cons:

- Hard to improve on general cases

Neural TN Model

Classification Neural model

- Carefully designed pattern groups
- Multi-head self-attention

Table 1. Examples of some dataset pattern rules.

Pattern Name	Pattern Example
A_Read_No_Zero	200 people
A_Spell_Keep_Zero	The 2020 Conference
B_Percent	Only 10% of students voted
B_Range	about 10-15 degree
B_Score_Ratio	Team A is 30-10 leading
B_Slash_Per	There are five people/group
B_Time	It starts at 10:30
B_Date_YMD	Today is 2019-10-01
A_Two_Liang	2个人 (2 people)
A_One_Yao_Spell	打911 (Call 911)

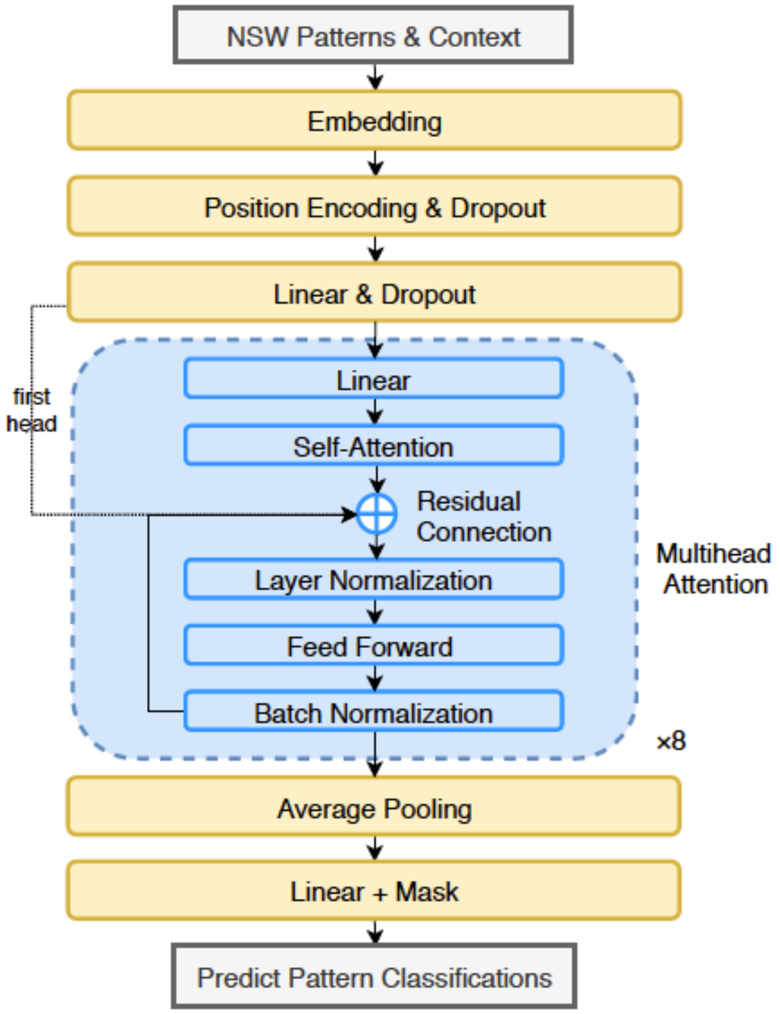


Fig. 2. Multi-head self-attention model structure.

ByteDance字节跳动

Neural TN Model

Word Embedding

- A pretrained Word2Vec model on Wikipedia text.
- Finetune on a pre-trained BERT model.

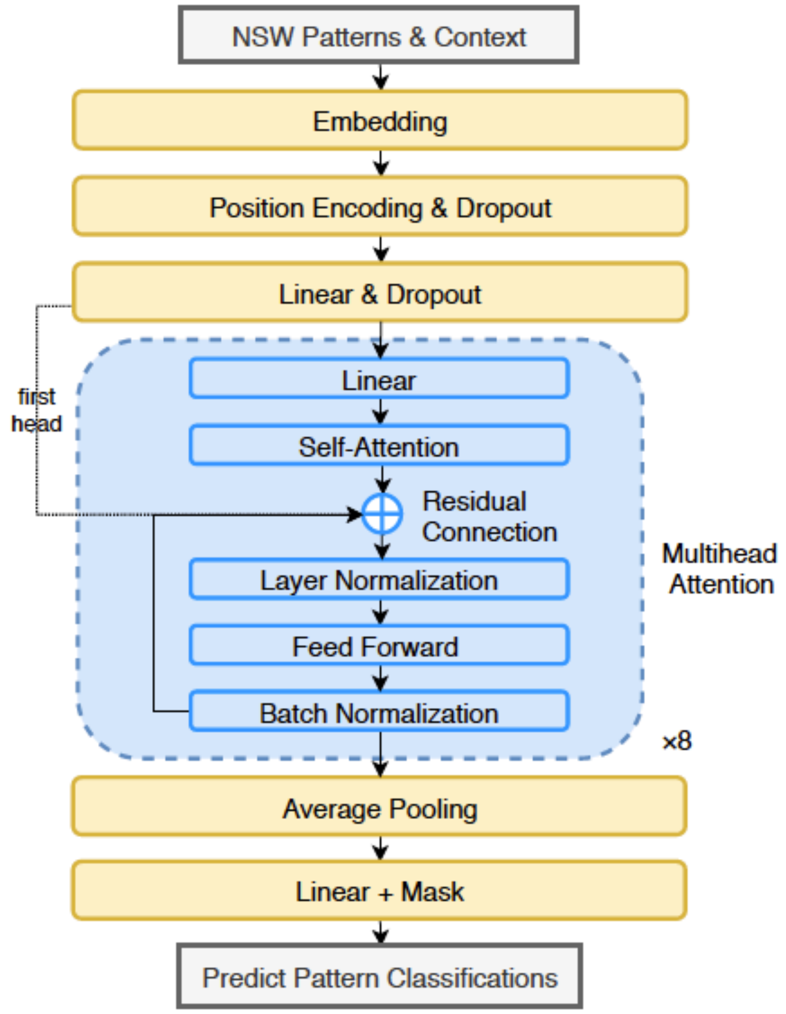


Fig. 2. Multi-head self-attention model structure.

Proposed Hybrid System

- Priority Check (rule-based TN model)
  - Easy to add user-defined strings.
  - Easy to add special cases (e.g. 911).
  - Handles 22.8% non-standard words.
- Main Module (neural TN model)
  - Handles 77.2% non-standard words.
- Pattern Check (rule-based TN model)
  - 2.2% failed patterns from the main module.
  - Normalization of all remaining patterns.

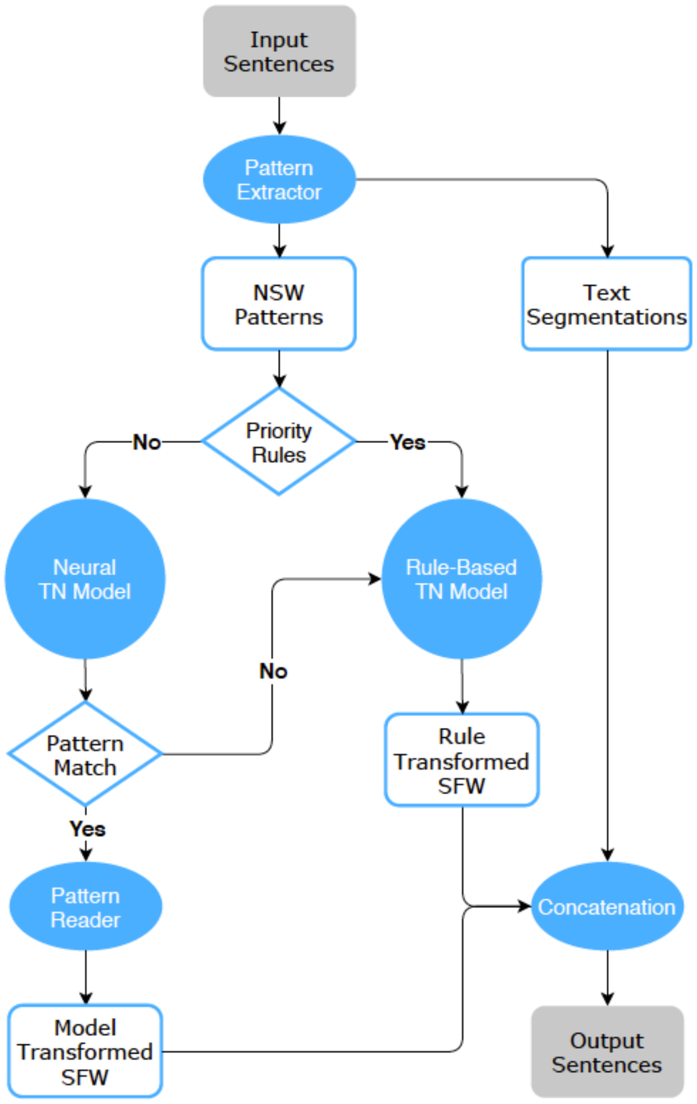


Fig. 1. Flowchart of the proposed hybrid TN system.

ByteDance字节跳动

Another Challenge – Imbalanced Dataset

- The dataset is imbalanced
  - Top 5 patterns take up > 90%.
  - Leading to a less robust neural model.
- Solutions
  - Introduce focal loss.

$$L = \begin{cases} -\alpha_t (1-p)^{\gamma} \log(p), \text{if} & y = 1 \\ -\alpha_t p^{\gamma} \log(1-p), \text{if} & y = 0 \end{cases} \quad (1)$$

  - Data expansion.
    - Data duplication, context replacing, random digits change…

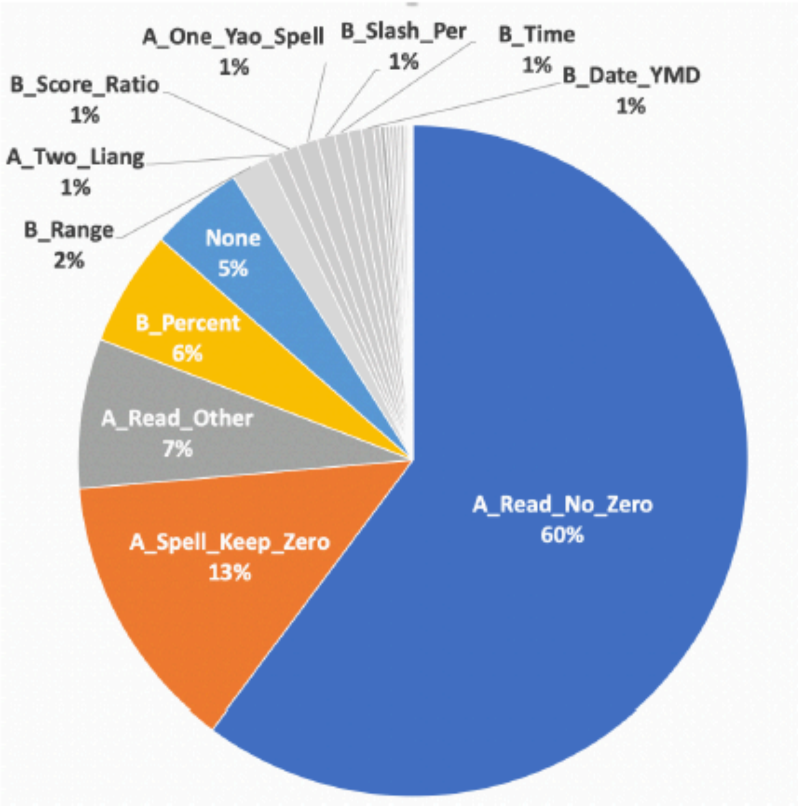


Fig. 3. Label distribution for dataset.

Experimental Result – Neural Model

- Proposed system has the following configurations:
  - Word2Vec Word Embedding.
  - Focal Loss without data expansion.
  - Bi-classification mask (whether a symbol exists).

Table 2. Comparison of different experimental setups.

Experimental setup	Accuracy
Model 1 (proposed)	0.916
Model 2 (+ BERT)	0.904
Model 3 (+ pad 0's)	0.914
Model 4 (+ max window)	0.907
Model 5 (+ CE loss)	0.913
Model 6 (- mask)	0.910
Model 7 (+ data expansion)	0.908

Experimental Result – Neural Model

- Neural model performance on different pattern groups.

Table 3. Model performance on the test dataset.

Pattern Name	Precision	Recall	F_1
A_Read_No_Zero	0.974	0.979	0.977
A_Spell_Keep_Zero	0.932	0.916	0.924
B_Percent	0.998	0.990	0.994
B_Range	0.932	0.932	0.932
B_Time	0.969	0.912	0.939
B_Score_Ratio	0.962	0.962	0.962
B_Slash_Per	0.994	0.966	0.980
B_Date_YMD	1.000	0.923	0.960
A_Two_Liang	0.613	0.797	0.693
A_One_Yao_Spell	0.637	0.631	0.634
Overall Accuracy		0.916	

Experimental Result – Proposed System

- Performance comparison on golden test set (~70,000 sentences)
  - Increased accuracy by 1.9% on sentence level.
  - On average, 95.5% pattern accuracy is achieved.
  - Our service shows the system is more robust on different types of news.

Table 4. Model performance on the news golden set.

	Sentence Accuracy	Pattern Accuracy
Rule-based TN model	0.867	0.946
Proposed TN system	0.886	0.955

Thanks for watching!