ASNLU at the NTCIR-14 finnum task: Incorporating Knowledge into DNN for Financial Numeral Classification

Chao-Chun Liang and Keh-Yih Su
Institute of Information Science, Academia Sinica, Taiwan

Abstract

This paper describes our work for solving the financial numeral classification problem in the NTCIR-14 FinNum task, and discusses experimental results. After implementing the three proposed vanilla neural network models (CNN, RNN, and RNN with CNN filters), we further incorporate POS and NE linguistic features. Inspired by human observation, we also propose a pre-processing procedure, which splits numerals in the Twitter string in advance, to reduce the OOV rate in the test set. Experimental results show both approaches improve the performance significantly.

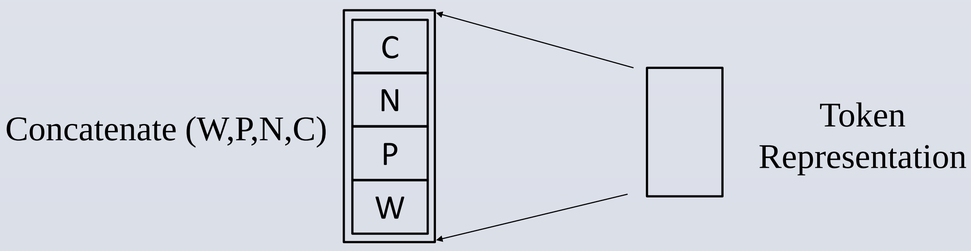
Financial Numeral Classification

- Model the Financial Numeral Classification as a sequence labeling task.
- Each word is tagged by a label, which is a member of the union of 'O' and the pre-specified FinNum-Category set.

Tweet	**8**	breakouts:	$CHMT	(stop:	**$17.99**	).
Main Category	**Quantity**	O	O	O	**Monetary**	O
Sub Category	**Quantity**	O	O	O	**stop loss**	O

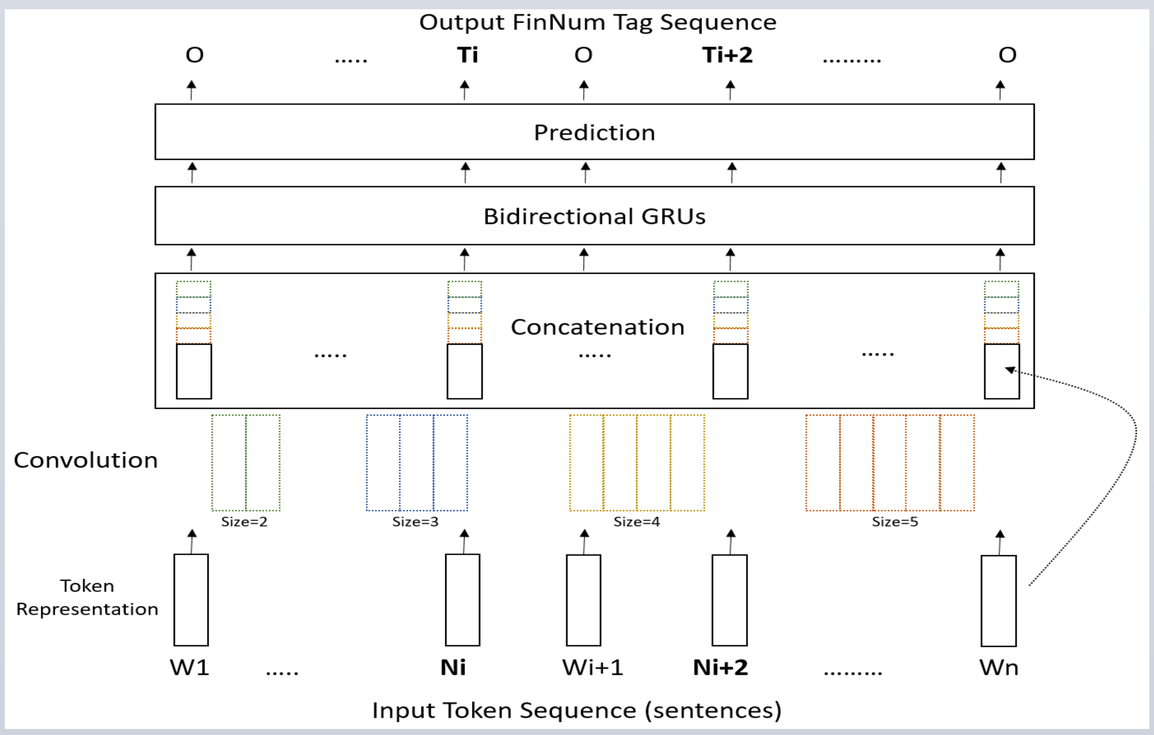
External Knowledge Token Representation

- Pre-trained word embedding (**W**).
- Part-of-Speech (**P**) and named-entity (**N**).
- Category Pattern (**C**). Collect six common handcrafted patterns that includes "Company" (e.g. '$NTNX'), "Money" (e.g. "$ 20'), "Product number" (e.g., "CYC065'), "Date" (e.g., "11/09/17"), "Time" (e.g., "3.25 pm") and "Number" (e.g., "67").



CNN, RNN, RNN+CNN based Approaches

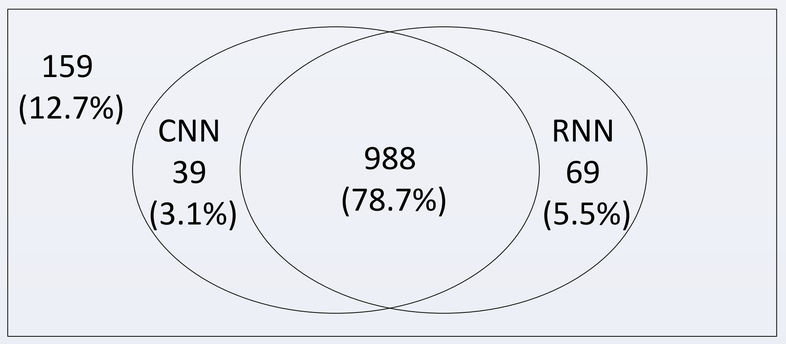
- Detect local pattern features by CNN based approach.
- Capture context information by RNN based approach.
- Capture local patterns in RNN by RNN+CNN based approach.



Proposed RNN architecture with CNN filters

Evaluation Results

- The CNN-based model and RNN-based model capture different features and complement each other.



- Linguistic information improves performance significant while OOVs provide no information useful for category identification.
- Handcrafted category patterns do not have enough coverage and gain limited improvement.

7-category classification						
	CNN		RNN		RNN with CNN	
	Micro	Macro	Micro	Macro	Micro	Macro
None	81.83	69.54	84.22	73.36	82.71	69.63
+POS&NE	88.21	79.14	88.45	78.63	**89.72**	80.93
+POS&NE +Patterns	87.73	78.47	88.76	83.55	89.24	**81.50**
17-(sub)category classification						
None	69.88	58.66	75.22	71.72	73.94	65.54
+POS&NE	75.14	65.77	78.49	72.37	78.17	70.16
+POS&NE +Patterns	76.41	68.5	79.36	70.5	**79.12**	**72.51**

Results after Number-Splitting Preprocess

- Split each token with numbers into individual sub-tokens. For example, "$80" is split to "$" and "80" two sub-tokens.
- Reduced OOV ratios to 25%, 22%, and 23% (from 40+%, 30+%, 30+%) on the training, development, and test sets, respectively.
- Micro F-measures of CNN, RNN, and RNN+CNN models gained 7.7%, 8.1% and 9.4% respectively (without POS&NE).
- Experimental results show that the category patterns automatically learned by CNN outperform the handcrafted patterns.

7-category classification						
	CNN		RNN		RNN with CNN	
	Micro	Macro	Micro	Macro	Micro	Macro
None	89.56	83.17	92.27	86.60	92.11	88.18
+POS&NE	90.68	83.60	91.95	**88.36**	**92.99**	88.25
17-(sub)category classification						
None	76.57	66.84	**85.34**	**80.93**	82.23	76.92
+POS&NE	77.37	68.31	85.18	79.36	84.46	76.34

Conclusion

- Incorporating linguistic knowledge into DNN models improves performance of financial numeral classification task.
- A suitable pre-processing (i.e., splitting numerals in the Twitter string in advance) for reducing OOV rate significantly improves performance.

NTCIR-14-2019