Deep Neural Architecture for Recovering Dropped Pronouns in Korean

Sangkeun Jung and Changki Lee

Pronouns are frequently dropped in Korean sentences, especially in text messages in the mobile phone environment. Restoring dropped pronouns can be a beneficial preprocessing task for machine translation, information extraction, spoken dialog systems, and many other applications. In this work, we address the problem of dropped pronoun recovery by resolving two simultaneous subtasks: detecting zero-pronoun sentences and determining the type of dropped pronouns. The problems are statistically modeled by encoding the sentence and classifying types of dropped pronouns using a recurrent neural network (RNN) architecture. Various RNN-based encoding architectures were investigated, and the stacked RNN was shown to be the best model for Korean zero-pronoun recovery. The proposed method does not require any manual features to be implemented; nevertheless, it shows good performance.

Keywords: Deep learning, Dropped pronoun recovery, LSTM Encoding, Zero pronoun.

Manuscript received Apr. 6, 2017; revised Nov. 23, 2017; accepted Jan. 8, 2018.
Sangkeun Jung (hugmanskj@gmail.com), Seoul, Rep. of Korea.
Changki Lee (corresponding author, leeck@kangwon.ac.kr) is with the Department of Computer Science, Kangwon National University, Chuncheon, Rep. of Korea.

I. Introduction

Dropped arguments or zero pronouns are a frequent phenomenon in the Korean language [1]. Korean is a pro-drop language, meaning pronouns can be dropped from a sentence without causing the sentence to become ungrammatical or incomprehensible when the identity of the pronoun can be inferred from the context. The phenomenon occurs more frequently in the mobile texting and verbal communication environments, where the pronouns can be guessed in the discourse. Below are typical zero-pronoun examples, illustrated by Korean and translated English pairs. Dropped pronouns or arguments are enclosed in brackets, and the corresponding English words are underlined.

- [나는] 10 월 9일로 변경 하고 싶은데요.
  I want to change the date to October 9th.
- 저녁 먹을 때 [나를] 깨워 주세요.
  Please wake me up when dinner is ready.
- [제가] 지금 나가도 되나요?
  Can I leave now?

Recovering dropped pronouns could be beneficial for many natural language processing applications. In natural language understanding tasks, recovering the pronoun is essential to extract subject and object relations. Restoring dropped personal pronouns can be a useful preprocessing step for machine translation between pro-drop languages and non-pro-drop languages. Recovering pronouns can help build a robust alignment and language model for translation.

Traditional pronoun recovery can be categorized into two types: rule-based and statistics-based systems. Rule-based zero-pronoun recovery systems iteratively or sequentially apply rules inspired by linguistic knowledge to recover pronouns [2], [3]. Language dependency and huge rule-design costs are weaknesses of rule-based pronoun recovery systems. Statistics-based systems use probabilistic models for the recovery of pronouns [4]–[7]. Statistics-based systems are easier to build than rule-based systems owing to machine learning; however, to build a syntactically and semantically correct pronoun recovering system, a large amount of manual feature engineering is required.

Recent advances in artificial neural networks overcome these weaknesses. Deep neural networks are constructed with multiple levels of hidden layers, and each layer uses a non-linear activation function, which transforms each representation at one level into a representation at a higher and slightly more abstract level. Feature and solution representations are trained automatically by an iterative feed-forward and back-propagate process. In this paper, we introduce a novel deep neural architecture for recovering pronouns.

Usually, zero-pronoun recovery tasks consist of three problems: (1) detecting whether pronouns are dropped or not, (2) determining the types of dropped pronouns, and (3) locating the dropped positions. In this work, we address only the first and second problems. Two tasks are modeled and predicted simultaneously instead of using a cascading method. A long short-term memory (LSTM)-based recurrent neural network (RNN) is used for encoding the syntax and semantics of the sentence for both tasks.

The rest of the paper is structured as follows. Section II discusses related works. Section III describes the deep neural architecture for the zero-pronoun recovery problem. Variants of RNNs are introduced as effective encoders to recover dropped pronouns. We present the experimental results in Section IV. The conclusion is discussed in Section V.

II. Related Works

Pronouns are typically resolved in three steps: detecting zero pronouns, determining the type of dropped pronouns, and resolving the lexicon position of pronouns [2], [8]–[10]. In this work, we focused on detecting dropped pronouns and determining what type of pronouns they are, as done in [11] for Chinese text messages.

Dropped pronoun detection is related to empty category (EC) detection and resolution [12], [13]. Yang and Xue [11] addressed a simpler problem: detecting whether an empty category immediately precedes each word, but not determining the category. They used a maximum entropy classifier with lexical and parse-based features. Chung and Gildea [12] developed and compared a rule-based approach, a conditional random field (CRF) approach, and an approach based on training a parser with manually produced parses, including empty categories.

Dropped (zero) anaphora resolution involves identifying all dropped noun phrase slots in a document and, for each dropped phrase, identifying its antecedent or determining that one does not exist. Pronoun resolution is a special case, where the dropped noun phrase is restricted to being a pronoun [4]. Several studies have addressed dropped anaphora and dropped pronoun resolution in formally written Chinese text. Zhao and Ng [2] employed a rule-based procedure to detect dropped pronouns and to identify candidate antecedents. They used a decision tree classifier to assign dropped pronouns to antecedents. Yang and others [3] used a similar approach, except they used a more sophisticated rule-based approach (based on verbal logic valence theory) to identify the dropped pronoun.

Recently, more complex statistical methods have been adopted to resolve zero pronouns. Yang and others [5] trained a 17-class maximum entropy classifier to assign words to one of 16 types of dropped pronouns or "none." Giannella and others [4] used a CRF model to predict which words are at the start of an independent clause. Then, using the independent clause start information and lexical and syntactic information, they applied a CRF or a maximum-entropy classifier to predict whether a dropped personal pronoun immediately preceded each word.

Recently, deep neural network-based zero-pronoun recovery approaches have been introduced. Chen and Ng [14] exploit feed-forward networks to model zero-pronoun recovery. Wang and others [15] viewed the zero-pronoun recovery problem as a sequence labelling task, and developed a generation model based on recurrent neural networks and language models.

Compared to Chinese zero-pronoun works, relatively few works have been reported on the Korean zero-pronoun problem. In the work of [1], maximum entropy models are used to rank and resolve Korean anaphoric zero pronouns, while [16] used Weka as a machine learning tools to resolve the Korean zero-object problem.

In our view, ideal zero-pronoun recovery systems should have the following properties: (1) Minimal human design efforts are required. Syntactic and semantic features to detect dropped pronouns and determine the types of dropped pronouns should be learned automatically. (2) Detecting dropped pronouns and determining the type of zero pronouns should be resolved at the same time. A cascaded method could have error propagation between the two subtasks. Our proposed approach does not require any manual feature engineering. It can be trained in an end-to-end fashion with only tagged

https://doi.org/10.4218/etrij.2017-0085

data, and can detect dropped pronouns and types simultaneously.

III. Deep Neural Architecture for Recovering Zero Pronouns

1. Problem Formulation

This paper formulates the zero-pronoun problem as two subtasks: detecting zero pronouns and determining the type of dropped pronouns.

The zero-pronoun detection task is a simple binary classification problem. The system reads a sentence and then determines whether pronouns have been dropped (class: Dropped) or not (class: Not Dropped). Determining the type of dropped pronouns is a multi-classification task. The system should decide the corresponding correct types of dropped pronouns after reading a sentence. Seven types of dropped pronouns are defined for this task (Table 1).

Because the type determination task is only conducted when a zero pronoun is detected, we can simply combine the two tasks into a single task by extending the types of dropped pronouns from seven to eight (seven types for Dropped and one for Not Dropped).

Table 1. Types of dropped pronoun and corresponding examples. Dropped Korean pronouns or arguments are enclosed in brackets, and corresponding English words are underlined.

Category	Example
1st singular person	[내가] 카메라를 떨어뜨렸습니다.
	I dropped the camera.
1st plural person	[우리가] 학교 끝나고 쇼핑몰 입구에서 만나자.
	Let us meet at the entrance of the mall after school.
2nd singular person	[당신이] 이 도시 출신인가요?
	Are you from this city?
2nd plural person	[당신들이] 아직 인사들 안 나누시었죠?
	Did you not say hello yet?
3rd singular male person	[그가] 출장 중이에요.
	He is on a business trip.
3rd singular female person	[그녀가] 자연 분만이었나요, 제왕 절개이었나요?
	Was she a natural childbirth or a Caesarean section?
3rd plural person	[그들이] 늦어진 이유가 뭐래요?
	Why are they so late?
NotDropped	(class for sentence without drops)

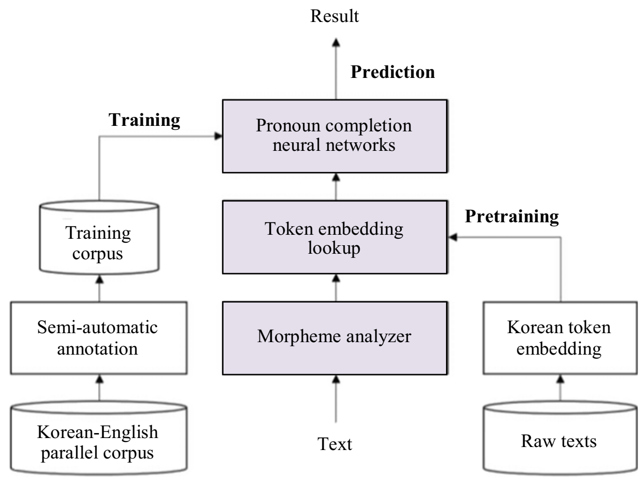

Fig. 1. Overall architecture of the proposed pronoun completion system.

A zero-pronoun task can be statistically formulated as follows:

$$P(z|\mathbf{X}),$$

where $z \in \{t_1, t_2, t_3, \ldots, t_8\}$ stands for the type of dropped pronoun and $\mathbf{X}$ is a sequence of tokens (words) ($X = [x_1, x_2, x_3, \ldots, x_n]$ and n = number of tokens).

The overall architecture of the proposed system is shown in Fig. 1.

2. Korean Token Embedding

In Korean, a word (space segmented) is composed of several morphemes, and there are many variants. Therefore, a problem of large vocabulary size arises if a basic word unit embedding technique is used. To avoid this problem, we used the morpheme-based modified *word2vec* [17] method.

News articles from 2012–2013 and Korean Wiki-Abstract texts (9 GB raw text, 2.9 billion morphemes) are used for training morpheme-embedding parameters after number (0–9 to 0) and alphabet (to lowercase) normalization.

Only the top 100,000 most frequent morpheme-tag units are selected for the vocabulary set, and all other units are handled as unknown symbols.

As an embedding learning technique, we employed the original word2vec's Continuous Bag-of-Words model (CBOW) and Skip-gram methods; however, they showed lower performance during initial experiments. To improve training speed, CBOW and skip-gram use the *sum* operator to merge all the results (multiplication of the input and the weight matrix) in the projection layer. As a result, all of the position information disappears. However, position information is very important in Korean. For example, the noun is always preceded by postpositional particles (가, 의, 을, ...), and the verb morpheme must be preceded by the ending of certain words (다, 네, 소서, ...).

To maintain and learn positional information, we modified the original CBOW method by changing the *sum* operator to a *concat* operator in the final projection layer, and we called this embedding technique *word2vec*(*N-gram*) (Fig. 2).

Figure 3 shows the embedding result (t-SNE) of word2vec(CBOW) and word2vec(N-gram). Korean morphemes are scattered without maintaining positional information in the word2vec(CBOW) results while word2vec(N-gram) preserves morpheme categorical information relatively well.

3. Recurrent Neural Network Based Encoding

An RNN is a neural network that consists of a hidden state h and an optional output y that operate on a variable length sequence $x = (x_1, x_2, \ldots, x_T)$, which is a Korean morpheme sequence. At each time t, the hidden state h_t of the RNN is updated by

$$h_t = f\left(\mathbf{U}(\mathbf{E}x_t) + \mathbf{V}h_{t-1} + \mathbf{b}_h\right), \qquad (1)$$

where $\mathbf{U}$ and $\mathbf{V}$ denote weight matrices (that is, $\mathbf{U}$ is the input-hidden weight matrix), $\mathbf{E}$ is a weight matrix for the word embedding, $\mathbf{b}_h$ denotes bias vectors, f is a nonlinear function (for example, sigmoid or tanh function). A graphical illustration of an RNN is shown in Fig. 4(a). Recently, [18] proposed a neural network architecture that learns to encode a variable-length sequence as a fixed-length vector representation. The encoder is an RNN that reads each symbol of an input sequence $\boldsymbol{X}$ sequentially. As it reads each symbol, the hidden state of the RNN changes according to (1). After reading the end of the sequence, the

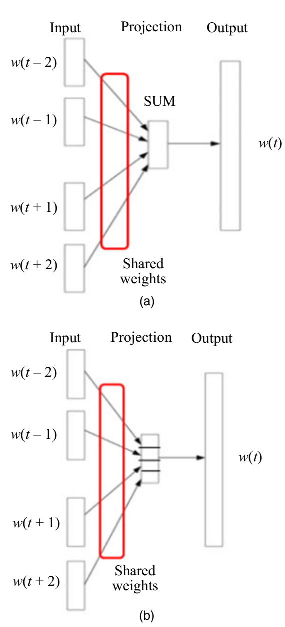

Fig. 2. (a) Original *word2vec*(*CBOW*) and (b) our *word2vec*(*N-gram*) methods. The *word2vec*(*N-gram*) method concatenates all the projected vectors instead of summation to keep the position information.

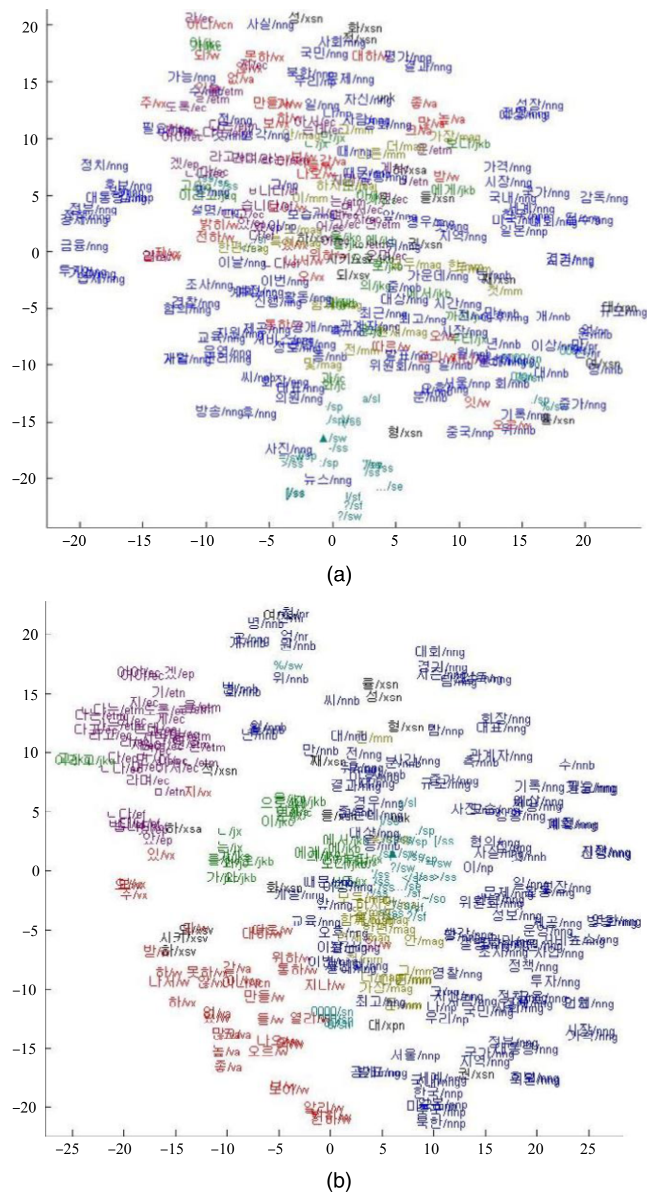

Fig. 3. (a) t-SNE results of original *word2vec*(*CBOW*) and (b) our *word2vec*(*N-gram*) methods. Nouns are shown in blue, verbs in red, post positional particles in green, and word endings in purple.

final hidden state of the RNN is a summary $\boldsymbol{S}$ of the whole input sequence [18]. In this work, zero-pronoun recovery is conducted based on this sentence summarization vector $\boldsymbol{S}$.

4. LSTM/GRU Encoding

It is well known that basic RNN (Fig. 4(a)) has problems handling "long-term dependencies." The problem was explored in depth in [19], [20]. LSTM networks – usually just called "LSTMs" – are introduced to tackle this problem in [19]. An LSTM is a special kind of RNN, capable of learning long-term dependencies. An LSTM block contains three gates that determine when the input is significant enough to remember (input gate), when it should continue to remember or forget the value (forget gate), and when it should output the value (output gate). An LSTM cell-based RNN is shown in Fig. 4(b).

Recently, [18] introduced the Gated Recurrent Unit (GRU), which is an architecture that is similar to LSTM, and [21] found GRUs to outperform LSTMs on a suite of tasks. In this work, LSTMs and GRUs are used as basic RNN blocks.

5. Stacked RNN Encoding

Stacked RNNs are built by stacking multiple LSTM/GRU layers (Fig. 5(a)). It has been argued that deep layers in RNNs allow the network to learn at different time scales across the input [22]. Deep LSTM/GRU RNNs offer another benefit over standard LSTM/GRU RNNs: they can make better use of parameters by

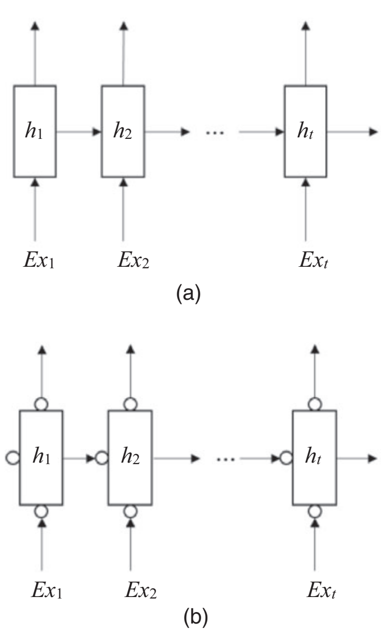

Fig. 4. (a) Basic RNN and (b) LSTM RNN. The three circles (top, left, and bottom) around hidden vector $\mathbf{h}_t$ in the LSTM block indicate the input, output, and forget gates, respectively.

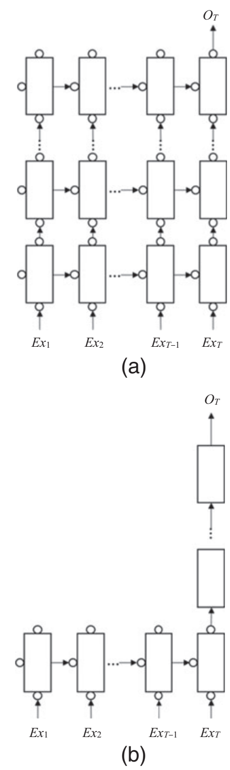

Fig. 5. (a) Stacked RNN and (b) RNN with deep output models.

Table 2. Example of dropped pronoun in parallel corpora and corresponding annotation.

Parallel sentence	
Korean	지금 운영체제에서는 실행이 불가능하니 [당신은] 웹페이지에서 추가로 다운로드 받으셔야 합니다. (Pronoun '당신은' is dropped.)
English	It is infeasible at the operating system now, **you** have to receive a download in the web page additionally. (Corresponding pronoun "**you**" exists.)
Annotation	지금 운영체제에서는 실행이 불가능하니 <za syn = "sub" per = "2" num = "1" position = "5" > 당신은 </za> 웹페이지에서 추가로 다운로드 받으셔야 합니다.
Korean	[내가] 10월 9일로 변경하고 싶은데 (Pronoun "내가" is dropped.)
English	**I**'d like to change the date to October 9th. (Corresponding pronoun "**I**" exists.)
Annotation	<za syn = "sub" per = "1" num = "1" position = "1" > 내가 </za> 10월 9일로 변경하고 싶은데요.

Table 3. Statistics of the built dropped pronoun corpus.

# of dropped subjects, objects, and indirect objects per sentence	
0	4,371
1	9,015
2	962
3	82
4	4
# of dropped subjects, objects, and indirect objects	
Subject	10,013
Object	236
Indirect subject	925
# of multiple dropped subjects in a sentence	
2	338
3	4
4	1

distributing them over the space through multiple layers [23].

6. RNN Encoding with Deep Output Network

We can put the other layers of a feedforward network (FFN) on top of the final output of the encoder (Fig. 5(b)). A number of theoretical results indicate that a deep, hierarchical model can be more efficient at representing some functions than a shallow one [20].

7. Dropped Pronoun Annotation

Unfortunately, there are few language resources for Korean zero-pronoun problems. In this work, we devised a semi-automatic dropped pronoun annotation processing method from parallel corpora. The semi-automatic

Table 4. Performance of various encoding methods. The word embedding dimension is 50. Dropout [P, H] stands for the dropout rate after embedding the projection (P) and hidden (H) layers, respectively. An asterisk (*) marks experiments that do not share the same data and are included for relative comparison. (The F1 difference between *2 Layer-Stacked GRU Enc*. at best parameter (800, 200) and *LSTM Enc*. at best parameter (1,600) is statistically significant according to a paired t–test at confidence level 0.95 (p–value = 0.035).

Model	1st dim	2nd dim	Dropout [P, H]	Precision	Recall	F1
LSTM Enc. (baseline)	400	N/A	[0.2, 0.5]	73.17	55.68	63.24
	800	N/A	[0.2, 0.5]	74.77	55.68	63.38
	1,600	N/A	[0.2, 0.5]	73.30	**64.97**	**68.88**
	3,200	N/A	[0.2, 0.5]	**75.36**	60.32	67.01
GRU Enc.	200	N/A	[0.2, 0.5]	70.51	61.02	65.42
	400	N/A	[0.2, 0.5]	74.85	58.00	65.36
	800	N/A	[0.2, 0.5]	**75.07**	**64.97**	**69.65**
	1,600	N/A	[0.2, 0.5]	74.24	62.18	67.68
GRU Enc. + deep output	400	400	[0.2, 0.5]	75.07	**64.97**	69.65
	800	100	[0.2, 0.5]	73.32	63.11	67.83
	800	200	[0.2, 0.5]	74.86	62.88	68.35
	800	**400**	[0.2, 0.5]	**76.02**	64.73	**69.92**
	800	800	[0.2, 0.5]	74.08	65.66	69.62
2 Layer-Stacked GRU Enc.	400	200	[0.2, 0.5]	75.60	65.43	70.15
	400	400	[0.2, 0.5]	71.99	**67.98**	69.93
	800	100	[0.2, 0.5]	**77.11**	65.66	**70.93**
	800	200	[0.2, 0.5]	73.98	67.29	70.47
	800	400	[0.2, 0.5]	73.20	65.89	69.35
Rule based (case frame) [6]	N/A	N/A	N/A	56.68*	56.91*	56.45*
Structural SVM [7]	N/A	N/A	N/A	67.24*	70.01*	68.58*

https://doi.org/10.4218/etrij.2017-0085

dropped pronoun annotation process consists of the following steps:

1. Extract candidate zero pronoun sentences from Korean–English parallel corpus automatically.
2. Human annotators tag zero pronoun information.
3. Cross-check the annotated sentences systemically.

Table 2 shows typical pronoun-dropped Korean sentence examples and the corresponding English sentences. In Korean, pronouns are dropped very often, while the corresponding English pronouns are not dropped. Those pair-sentences are collected as candidate sentences. In this study, 169,794 candidate sentences were collected from 330,974 parallel sentences. Human annotators examine the candidates and annotate the dropped pronoun information on sentences.

Using the above process, we built a total of 34,434 examples (14,434 from human-annotated data and 20,000 not-dropped sentences from machine-annotated data). In addition, 1,000 randomly sampled sentences from human-annotated sentences are used for testing, and the remaining sentences (33,434) are used for training. Statistics of the corpus are shown in Table 3.

IV. Experiments

To compare different encoding methods, a test was conducted. Precision, recall, and F1 measures are used. The experimental results are shown in Table 4, which shows that GRU RNN-based encoders are better than LSTM RNN encoders. The stacked GRU encoder shows the best performance.

Table 5. Good and bad test examples.

Good result examples	
1	[내가$_{\text{predicted}}$] 두 개를 더 샀습니다. I bought two more.
2	[내가$_{\text{predicted}}$] 오스트리아로 전화를 하고 싶어요. I want to call Austria.
3	[당신이$_{\text{predicted}}$] 송장 대금을 언제 지불했죠? When did you pay for the invoice?
Bad result examples	
1	[내가$_{\text{predicted}}$: 당신이$_{\text{reference}}$] 영수증을 끊어 줄 수 있나요? Can you give me a receipt?
2	[NotDropped$_{\text{predicted}}$: 우리가$_{\text{reference}}$] 그날은 별일이 없습니다. I have no other business that day.
3	[내가$_{\text{predicted}}$: NotDropped$_{\text{reference}}$] 베트남에서 열렸던 인천 아시아경기대회 홍보 기자회견에서도 JYJ는 몇 분 동안만 의무적인 모습으로 나타났다가 급히 떠나갔다. At the press conference of the Incheon Asian Games held in Vietnam, JYJ was at place just for a few minutes.

Table 5 shows good and bad pronoun recovery results. In many cases, the proposed model detects the dropped pronoun types correctly. However, for long sentences with a relatively large number of verbs, the model tends to detect the wrong pronoun types.

V. Conclusion

In this work, we proposed a novel deep neural architecture for Korean zero-pronoun resolution. We cast the zero-pronoun recovery problem into two subtasks, detecting zero pronouns and determining the type of dropped pronouns. These two subtasks were statistically resolved at the same time using an RNN-based encoding mechanism. To capture long-term dependency, LSTM- and GRU-based RNN encoders were used. RNN encoders with deep output models and stacked RNN were introduced to improve performance, and experiments showed that the stacked RNN was the best model. To overcome the lack of Korean zero-pronoun language resources, a semi-automatic corpus building process from Korean-English parallel corpora was introduced.

In this paper, we only addressed the problems of detecting zero pronouns and determining their types. To achieve full zero-pronoun recovery, dropped pronoun positions and lexicon should be recovered as well. In future work, we will investigate these problems. Pointer networks with an encoding-decoding mechanism might be promising candidates for solving them.

In this work, we focused on recovering pronouns in a sentence; however, some dropped pronouns cannot be recovered without context information across multiple sentences. Pronoun recovery using multiple sentences is planned for our future work.

Acknowledgements

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (No. NRF-2016R1C1B1014124).

References

[1] N.-R. Han, "*Korean Zero Pronouns: Analysis and Resolution*," Ph.D. Dissertation, Univ. Pennsylvania, Philladelphia, USA, 2006.

[2] S. Zhao and H.T. Ng, "Identification and Resolution of Chinese Zero Pronouns: A Machine Learning Approach," *Proc. Conf. EMNLP-CoNLL*, Prague, Czech Rep., June 28–30, 2007, pp. 541–550.

[3] W. Yang, R. Dai, and X. Cui, "Zero Pronoun Resolution in Chinese Using Machine Learning Plus Shallow Parsing," *Proc. Int. Conf. Inform. Autom.*, Changsha, China, June 20–23, 2008, pp. 905–910.

[4] C. Giannella, R. Winder, and S. Petersen, "*Dropped Pronoun Recovery in Chinese SMS*," Technical Paper, 2015.

[5] Y. Yang, Y. Liu, and N. Xu, "Recovering Dropped Pronouns from Chinese Text Messages," *Proc. Annu. Meet. Assoc. Comput. Linguistics Int. Conf. Natural Language Process.*, Beijing, China, July 26–31, 2015, pp. 309–313.

[6] S.J. Lee, C. Lee, and M. Jang, "Restoring an Elided Title for Encyclopedia QA System," *Proc. Korean Inform. Sci. Soc. Conf.*, vol. 32, no. 2, Nov. 2005, pp. 541–543.

[7] M.-K. Hwang, Y. Kim, D. Ra, S. Lim, and H. Kim, "Restoring Omitted Sentence Constituents in Encyclopedia Documents Using Structural SVM," *J. Intell. Inform. Syst.*, vol. 21, no. 2, 2015, pp. 131–150.

[8] S.P. Converse, "*Pronominal Anaphora Resolution in Chinese*," Ph.D. Dissertation, Univ. Pennsylvania, Philladelphia, USA, 2006.

[9] P. Jing and K. Araki, "Zero-Anaphora Resolution in Chinese Using Maximum Entropy," *IEICE Trans. Inform. Syst.*, vol. 90, no. 7, 2007, pp. 1092–1102.

[10] F. Kong and G. Zhou, "A Tree Kernel-Based Unified Framework for Chinese Zero Anaphora Resolution," *Proc. Conf. Empirical Methods Natural Language Process.*, Cambridge, MA, USA, Oct. 9–11, 2010, pp. 882–891.

[11] Y. Yang and N. Xue, "Chasing the Ghost: Recovering Empty Categories in the Chinese Treebank," *Proc. Int. Conf. Comput. Linguistics: Posters*, Beijing, China, Aug. 23–27, 2010, pp. 1382–1390.

[12] T. Chung and D. Gildea, "Effects of Empty Categories on Machine Translation," *Proc. Conf. Empirical Methods Natural Language Process.*, Cambridge, MA, USA, Oct. 9–11, 2010, pp. 636–645.

[13] S. Cai, D. Chiang, and Y. Goldberg, "Language-Independent Parsing with Empty Elements," *Proc. Annu. Meet. Assoc. Comput Linguistics: Human Language Technolo.*, Prtland, OR, USA, June 19–24, 2011, pp. 212–216.

[14] C. Chen and V. Ng, "Chinese Zero Pronoun Resolution with Deep Neural Networks," *Proc. Annu. Assoc. Comput. Linguistics*, Berlin, German, Aug. 7–12, 2016, pp. 778–788.

[15] L. Wang, X. Zhang, Z. Tu, H. Li, and Q. Liu, "Dropped Pronoun Generation for Dialogue Machine Translation," *IEEE Int. Conf. Acoust., Speech Signal Process.*, Shanghai, China, Mar. 20–25, 2016, pp. 6110–6114.

[16] A. Park, S. Lim, and M. Hong, "Zero Object Resolution in Korean," *Proc. Pacific Asia Conf. Language Inform. Comput.*, Shanghai, China, Oct. 30–Nov. 1, 2015, pp. 439–448.

[17] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *arXiv Prepr.*, arXiv1301.3781, 2013.

[18] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdaunau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation," *arXiv Prepr.*, arXiv1406.1078, 2014.

[19] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, 1997, pp. 1735–1780.

[20] Y. Bengio, ""Learning Deep Architectures for AI", Found. Trends®," *Mach. Learn.*, vol. 2, no. 1, Jan. 2009, pp. 1–127.

[21] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," *arXiv Prepr.*, arXiv1412.3555, 2014.

[22] M. Hermans and B. Schrauwen, "Training and Analyzing Deep Recurrent Neural Networks," *Proc. Int. Conf. Neural Inform. Process. Syst.*, Lake Tahoe, NV, USA, Dec. 5–10, 2013, pp. 190–198.

[23] H. Sak, A.W. Senior, and F. Beaufays, "Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling," *Annu. Conf. Int. Speech Commun. Assoc.*, Singapore, Sept. 14–18, 2014, pp. 338–342.

Sangkeun Jung received his BS, MS, and PhD degrees in computer engineering from POSTECH, Pohang, Rep. of Korea, from 2004 to 2010. From 2010 to 2012, he was a researcher with the Samsung Electronics, Suwon, Rep. of Korea. From 2012 to 2014, he was a researcher with the ETRI, Daejeon, Rep. of Korea. Since 2014, he has been a researcher at SK Telecom, Seoul, Rep. of Korea. His research interests include natural language processing, machine learning, and deep learning.

Changki Lee received his BS degree in computer science from the Korea Advanced Institute of Science and Technology, Daejeon, Rep. of Korea, in 1999. He received his MS and PhD degrees in computer engineering from POSTECH, Pohang, Rep. of Korea, in 2001 and 2004, respectively. From 2004 to 2012, he was a researcher with the ETRI, Daejeon, Rep. of Korea. Since 2012, he has been a professor of computer science at Kangwon National University, Chuncheon, Rep. of Korea. His research interests include natural language processing, machine learning, and deep learning.