Response Generation in Dialogue using a Tailored PCFG Parser

Caixia Yuan Xiaojie Wang Qianhui He
School of Computer Science
Beijing University of Posts and Telecommunications
{yuancx, xjwang}@bupt.edu.cn
alisonchinabupt@gmail.com

Abstract

This paper presents a parsing paradigm for natural language generation task, which learns a tailored probabilistic context-free grammar for encoding meaning representation (MR) and its corresponding natural language (NL) expression, then decodes and yields natural language sentences at the leaves of the optimal parsing tree for a target meaning representation. The major advantage of our method is that it does not require any prior knowledge of the MR syntax for training. We deployed our method in response generation for a Chinese spoken dialogue system, obtaining results comparable to a strong baseline both in terms of BLEU scores and human evaluation.

1 Introduction

Grammar based natural language generation (NLG) have received considerable attention over the past decade. Prior work has mainly focused on hand-crafted generation grammar (Reiter et al., 2005; Belz, 2008), which is extensive, but also expensive. Recent work automatically learns a probabilistic regular grammar describing Markov dependency among fields and word strings (Konstas and Lapata, 2012a, Konstas and Lapata, 2013), or extracts a tree adjoining grammar provided an alignment lexicon is available which projects the input semantic variables up the syntactic tree of their natural language expression (Gyawali and Gardent, 2014). Although it is a consensus that at a rather abstract level natural language generation can benefit a lot from its counterpart natural language understanding (NLU), the problem of leveraging NLU resources for NLG still leaves much room for investigation.

In this paper, we propose a purely data-driven natural language generation model which exploits a probabilistic context-free grammar (PCFG) parser to assist natural language generation. The basic idea underlying our method is that the generated sentence is licensed by a context-free-grammar, and thus can be deduced from a parsing tree which encodes hidden structural associations between meaning representation and its sentence expression. A tailored PCFG, i.e., a PCFG easily tailored to application-specific concepts, is learned from pairs of structured meaning representation and its natural language sentence and then used to guide generation processes for other previously unseen meaning representations. Table 1 exemplifies a record from the application under consideration.

Our model is closest to (Konstas and Lapata, 2012a) and (Konstas and Lapata, 2013) who reformulate the Markov structure between a meaning representation and a string of text depicted in (Liang, et al., 2009) into a set of CFG rewrite rules, and then deduce the best derivation tree for a database record. Although this Markov structure can capture a few elements of rudimentary syntax, it is essentially not linguistic grammars. Thus the sentences produced by this model are usually ungrammatically informed (for instance, its 1-BEST model produces grammatically illegal sentences like "Milwaukee Phoenix on Saturday on Saturday on Saturday on Saturday"). (Konstas and Lapata, 2013) claims that long range dependency is an efficient complementary to CFG grammar, and incorporates syntactic dependency between words into the reranking procedure to enhance the performance. Although conceptually similar, our model directly learns more grammatical rewrite rules from hybrid syntactic trees whose nonterminal nodes are comprised of phrasal nodes inheriting from a common syntactic parser, and conceptual nodes designed for encoding target meaning representation. Therefore, the learning aspect of two models is fundamentally different. We have a single CFG grammar that applies throughout, where-

Proceedings of the 15th European Workshop on Natural Language Generation (ENLG), pages 81–85,
Brighton, September 2015.

Table 1: Examples of meaning representation input as a structured database and its corresponding natural language expression. Each meaning representation has several fields, each field has a value.

Meaning representation	action1 confirm	object1 person	value11 100	value12 120	action2 request	object2 date	value21 null	value22 null
Text	与会人数在100人到200人之间，请问您在哪天开会？(The number of participants is between 100 and 200. When is the meeting scheduled?)							

as they train different CFG grammar and dependency grammar respectively.

The major advantage of our approach is that it learns a tailored PCFGs directly from MR and NL pairs, without the need to manually define CFG derivations, which is one of the most important prerequisites in (Belz and Kow, 2009) and (Konstas and Lapata, 2013), and thus porting our method to another applications is relatively easy. We demonstrate the versatility and effectiveness of our method on response generation for a Chinese spoken dialogue system (SDS)[1].

2 Problem Formulation

2.1 The grammar

Following most previous works in this area (Liang, et al., 2009; Konstas and Lapata, 2013), we use the term record r to refer to a (m, w) pair. Each meaning representation m is described as several fields f, each field has a value $f.v$. As exemplified in Table 1, each m in the referred SDS system has eight fields (e.g., action, object1, value11), each field has a specific value. The value can be a string (e.g., confirm, person), or a numeric quantity (e.g., 100, 120), or null. The text is simply a sequence of words $w = (w_1, ..., w_{|w|})$.

Our goal is to learn a PCFG for interpreting a MR using NL expression. In order to generate more coherent sentence, the established grammar should capture recursive structure of phrases. Meanwhile, in order to generate sentence expressing target meanings, the grammar should also capture concept embeddings corresponding to desired meaning fields. Under this framework, a tailored PCFG grammar we used for generation can be described as a 6-tuple:

$$G = \langle N_p, N_c, T, S, L, \lambda \rangle \quad (1)$$

where N_p is a finite set of non-terminal symbols produced by a common parser, N_c is a finite set of concept symbols related to specific record fields, T is a finite set of NL terminal symbols (words), $S \in N_p$ is a distinguished start symbol, L is a lexicon which consists of a finite set of production rules, and λ is a set of parameters that defines a probability distribution over derivations under G.

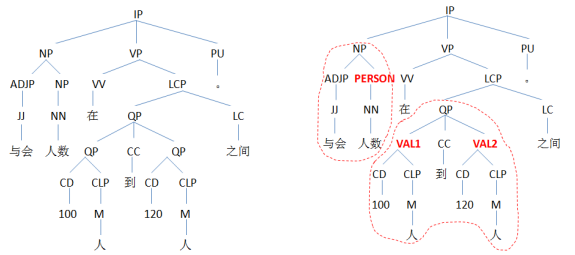
2.2 Grammar Induction

In this section, we present a learning procedure for the grammar described above. The input to the learning algorithm is a set of training sentences paired with their correct meaning representations (as illustrated in Table 1). The output from the learning algorithm is a PCFG describing both phrase and concept embeddings. The learning algorithm assumes that a common phrase structure parser is available, but it does not require any prior knowledge of the MR syntax.

As a concrete example, consider the record in Table 1. We first analyze its sentence expression using the Stanford parser (Chen and Manning, 2014) whose nonterminals are syntactic categories (e.g., NP, VP, JJ, NN). Figure 1(a) outlines the partial parser tree of sentence in Table 1. The meaning of the sentence is then integrated by adding conceptual symbols of its subparts into the parser tree. Figure 1(b) shows a hybrid parse tree of Figure 1(a). Here the nonterminal symbols in bold, PERSON, VAL1 and VAL2, represent domain-specific concepts corresponding to fields person, value1 and value2.

To get the hybrid parse tree, we first align phrases in the NL with the actual MR fields mentioned using the model of (Liang, et al., 2009) which is learned in an unsupervised manner using EM to produce which words in the text were spanned by the fields. The aligned pairs are recorded in a temporary table. Then for each phrase in the table, we find the minimal subtree spanning it, and modify its ancestor node attached directly below the subtree's root node to the conceptual symbol of its aligned field. All ancestor nodes keep unchanged for phrases not in the alignment table. The cen-

[1] A demo can be found at http://www.aidc.org.cn:8008/WebContent/



(a) (b)

Figure 1: Example of (a) a syntactic tree and (b) its corresponding hybrid tree from which the tailored PCFG defined in Formula (1) is constructed. The subtree circled by dotted line contains conceptual node and its terminal derivations.

tral characteristic of a tree structured representation is that component concept appears as a node in a tree, with its word realizations as terminal nodes derived by it. For example, the concept PERSON has a terminal node "人数", and VALUE1 "100人", these could then form part of the representation for the sentence "与会人数在100 人到200 人之间。(The number of participants is between 100 and 200.)" The use of a recursive hybrid syntactic and conceptual structure is one characteristic that distinguishes the proposed grammar from earlier work in which meaning is represented by logical forms or regular grammars (Lu and Ng, 2011; and Konstas and Lapata, 2013).

Given hybrid trees, N_p, N_c, T, S and the set of derivations that are possible are fixed, we only need to learn a probabilistic model parameterized by λ. Since the "correct" correspondence between NL words and MR fields is fully accessible, i.e., there is a single deterministic derivation associated with each training instance, model parameter λ can be directly estimated from the training corpus by counting. Because the derived trees output by parser can be noisy, we need to process them to obtain cleaner PCFG rules. We compare the 3-best trees produced by the Stanford Parser, and prune off the inconsistent components voted by majorities when extracting and counting rules.

2.3 Decoding

Our goal in decoding is to find the most probable sentence $\hat{s}$ for a given meaning expression m:

$$\hat{s} = g(\underset{D \ s.t. \ m(D)=m}{\operatorname{argmax}} P(D|G) \cdot \ln(|D|+1)) \quad (2)$$

where g is a function that takes as input a derivation tree D and returns $\hat{s}$, $m(D)$ refers to the meaning representation of a derivation D, and $P(D|G)$ is product of weights of the PCFG rules used in a derivation D, the factor $\ln(|D|+1)$, offers a way to compensate the output sentence length $|D|$. We use a decoding paradigm introduced in (Konstas and Lapata, 2013) which is essentially a bottom-up chart-parsing algorithm without forcing the input to exhibit linear structure. It first fills the diagonal cell of the chart with terminal words with the top scoring words emitted by the unary rules of the type $A \rightarrow \alpha$, where A is a non-terminal symbol, and α is a terminal word.

In order to search among exponentially many possible tree structures for a given MR, a k-best decoder is achieved by adding to the chart a list of the top k words and production rules, then an external language model is used to rerank the derived partial trees in a timely manner with cube pruning (Huang and Chiang, 2005).

3 Empirical Evaluation

We conducted experiments on a Chinese spoken dialogue system (SDS) for booking meeting room. Our NLG module receives structured input from dialogue management (DM) module and generates natural language response to user. The structured input includes dialogue actions (e.g., greet, request, confirm), objects (e.g., date, budget, location) and object values which can be a null. The SDS corpus consists of 1,406 formal meaning representations, along with their Chinese NL expressions written by 3 Chinese native speakers. The average sentence length for the example data is 15.7 Chinese words. We randomly select 1,000 record pairs as training data, and the remaining 406 is used as testing data.

To evaluate the quality of the generated sentences, the BLEU score (Papineni et al., 2002) is computed by comparing system-generated sentences with human-written sentences. In addition, we evaluated the generated text via a human judgment as designed in (Angeli et al., 2010). The subjects were presented with a MR and were asked to rate its corresponding NL expression along two dimensions: grammatical fluency and semantic correctness. A five point rating scale is designed where a higher number indicates better performance. The averaged score of three human evaluators was computed.

In order to compare our work with previous related work, Table 2 summarizes results achieved

Table 2: BLEU scores, and human ratings for syntactic fluency (SF) and semantic correctness (SC) of different systems.

system	BLEU	SF	SC
1-BEST-Konstas	9.32	2.29	1.94
k-BEST-Konstas	21.85	3.91	3.12
1-BEST-Our	30.88	4.36	3.95
k-BEST-Our	31.96	4.34	4.33
HUMAN	–	4.76	4.89

	action1	object1	vlue11	other fields
	confirm	place	北京	null
1-BEST-Konstas	会议在北京在北京(The meeing is in Beijing in Beijing)			
k-BEST-Konstas	地点在北京，在北京召开(The place is in beijing, take place in Beijing)			
1-BEST-Our	初步定在北京，好的(Scheduled in Beijing, alright)			
k-BEST-Our	会议将在北京召开，对吗(The meeting will be held in Beijing, right?)			

Figure 2: An example of generations produced by each of the four models.

using the proposed tailored PCFGs with that using the grammar described in (Konstas and Lapata, 2013). 1-BEST signifies results obtained from the basic decoder described in Section 2.3, and k-BEST is results of the k-best decoder reranked with a bigram language model. Here we set $k = 20$ without more fine-tuning work.

To make intensive comparisons, the length of the generated sentence is not restricted as a fixed number, while varying from 1 to a length of the longest sentence in the training data. The sentences with different length are overall sorted to obtain the 1-BEST and the k-BEST.

From Table 2, we find that differences in BLEU scores between 1-BEST-Konstas and 1-BEST-Our are statistically significant (9.32 vs. 30.88). Since the only difference between these two results is the grammar used, we have reason to justify that the tailored grammar learnt from the hybrid phrase-concept trees is superior for modeling NL and MR correspondence to that used in (Konstas and Lapata, 2013). It is interesting to notice that k-BEST-Konstas observes substantial increase in performance compared to 1-BEST-Konstas, while k-BEST-Our only achieves a slight increase compared to 1-BEST-Our. Statistical language model offers potentially significant advantages for the sequential Markov grammar as reported in (Konstas and Lapata, 2013), but it contributes little to the tailored PCFGs. This also verifies the robustness of the proposed method.

Table 2 also summarizes the human ratings for each system and the gold-standard human-authored sentences. From Table 2 we can observe that our method consistently produce good Chinese sentences in terms of both grammatical coherence and semantic soundness, which is consistent with the results of automatic evaluation. Another major advantage of our method over method of (Konstas and Lapata, 2013) is that it does not require any prior knowledge of the MR syntax for training. Therefore, transplanting our method to other NLG application is relatively easy.

Figure 2 shows the generations of the four models on an example. 1-BEST-Konstas is only able to form Markov but not grammatical associations. k-BEST-Konstas improves it by accounting for more possible associations, but errors are still made due to the lack of syntactic structure. 1-BEST-Our and k-BEST-Our remedies this. However, unexpected sentences are still produced in the cases of long rang correlation. For example, k-BEST-Our produced a sentence " 会议日期什么时候举行呢？(When is the meeting date held?)" which is a grammatically well-formed sentence but has poor fluency and meaning. As perceived in the work of syntactic parsing, PCFG is very difficult to capture long range dependency of word strings.

4 Conclusions

We have presented a PCFG-based natural language generation method. In particular, the method learns tailored PCFG rules from hybrid phrase-concept trees automatically augmented from the output of a common syntactic parser. A compelling advantage of the proposed method is that it does not rely on prior knowledge of the MR syntax for training. We have shown the competitive results in a Chinese spoken dialogue system. Future extensions include deploying more efficient decoding algorithms, and richer structural features to rerank the derivations.

Acknowledgments

This work was partially supported by Natural Science Foundation of China (No. 61202248, No. 61273365), Discipline Building Planing 111 Base Fund (No. B08004).

References

Gabor Angeli, Percy Liang, and Dan Klein. 2010. A Simple Domain-Independent Probabilistic Approach to Generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 502-512, Cambridge, MA.

Ioannis Konstas and Mirella Lapata. 2013. A Global Model for Concept-to-Text Generation. *Journal of Artificial Intelligence Research*, 48(2013): 305-346.

Ioannis Konstas and Mirella Lapata. 2012a. Concept-to-Text Generation via Discriminative Reranking. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 369-378, Jeju, South Korea.

Ioannis Konstas and Mirella Lapata. 2012b. Unsupervised Concept-to-text Generation with Hypergraphs. In *Proceedings of 2012 Conference of the North American Chapter of the of the ACL: Human Language Technologies*, pp.752-761, Montreal, Canada.

Liang Huang and David Chiang. 2005. Better K-best Parsing. In *Proceedings of the 9th Interna-tional Workshop on Parsing Technology*, pp. 53-64, Vancouver, British Columbia.

David Chiang. 2007. Hierarchical Phrase-Based Translation. *Computational Linguistics*, 33(2): 201-228.

Percy Liang, Michael I. Jordan, and Dan Klein. 2009. Learning Semantic Correspondences with Less Supervision. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pp. 91-99, Suntec, Singapore.

Wei Lu and Hwee Tou Ng. 2011. A Probabilistic Forest-to-String Model for Language Generation from Typed Lambda Calculus Expressions. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pp. 1611-1622, Edinburgh, Scotland, UK.

Anja Belz. 2008. Automatic Generation of Weather Forecast Texts Using Comprehensive Probabilistic Generation-space Models. *Natural Language Engineering*, 14(4):431-455.

Anja Belz and Eric Kow. 2009. System Building Cost vs. Output Quality in Data-to-text Generation. In *Proceedings of the 12th European Workshop on Natural Language Generation*, pp. 16-24, Athens, Greece.

Danqi Chen and Christopher D. Manning. 2014. A Fast and Accurate Dependency Parser using Neural Networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 740-750, Doha, Qatar.

Nathan McKinley and Soumya Ray. 2014. A Decision-Theoretic Approach to Natural Language Generation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pp. 552-561, Baltimore, Maryland, USA.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu 2002. BLEU: a Method for Automatic Evaluation of Machine Translation In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311-318. Philadelphia, PA.

Bikash Gyawali and Claire Gardent 2014. Surface Realisation from Knowledge-base. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pp. 424-434. Baltimore, Maryland.

Miguel Ballesteros, Bernd Bohnet, Simon Mille, and Leo Wanner. 2015. Data-driven Sentence Generation with Non-isomorphic Trees. In *Proceedings of the 2015 Annual Conference of the North American Chapter of the ACL*, pp. 387-397, Denver, Colorado.

Verena Rieser and Oliver Lemon. 2009. Natural Language Generation as Planning Under Uncertainty for Spoken Dialogue Systems. In *Proceedings of the 12th Conference of the European Chapter of the ACL*, pp. 683-691, Athens, Greece

Yuk Wah Wong and Raymond J. Mooney. 2007. Generation by Inverting a Semantic Parser That Uses Statistical Machine Translation. In *Proceedings of the Human Language Technology and the Conference of the North American Chapter of the Association for Computational Linguis-tics*, pp. 172-179, Rochester, NY.

Ondrej Dusek and Filip Jurcicek. 2015. Training a Natural Language Generator from Unaligned Data. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp.410-419, Beijing, China.

Michael White and Rajakrishnan Rajkumar. 2009. Perceptron Reranking for CCG Realization. In *Proceedings of the 2009 Conference on EmpiricalMethods in Natural Language Processing*, pp.451-461, Suntec, Singapore.