Structure-based Generation System for E2E NLG Challenge

Dang Tuan Nguyen and Trung Tran
University of Information Technology, VNU-HCM
Ho Chi Minh City, Vietnam
dangnt@uit.edu.vn, ttrung@nlke-group.net

Abstract

This paper describes the structure-based generation system (SBG System) for End-to-End Natural Language Generation (E2E NLG) Challenge. The input of SBG System is each meaning representation (MR) in E2E data, which is a new dataset for training end-to-end, data-driven natural language generation systems in the restaurant domain. The output of SBG System is the corresponding Natural Language (NL) Reference for each MR. We follow the traditional approach when building SBG System including two main sub-tasks. The first sub-task is sentence planning, in which we create the overall sentence structures and determine the appropriate structure for each input MR. The second sub-task is a surface realization, in which we find the exact word forms and linearize the structure into a string.

The generated NL references from development and experiment sets by SBG System are compared to a baseline as well as other high-score systems with both automatic and human evaluation. The evaluation results show that our method generates high-quality NL references and has a meaningful contribution to the NLG state-of-the-art.

1 Introduction

Natural language generation (NLG) plays a critical role in recent interaction systems. So far, end-to-end (E2E) NLG methods (Mairesse et al. 2010; Wen et al. 2015; Chen and Mooney 2008) were limited to small, de-lexicalised data sets. For new application domain, the NLG systems should be re-developed so that they can replicate the rich dialogue and discourse phenomena.

In E2E NLG Challenge[1], the Committee focuses on recent E2E, data-driven NLG methods (Wen et al. 2015; Mei et al. 2016; Dusek and Jurcicek 2016; Lampouras and Vlachos 2016). From the original architecture, these methods should have two sub-tasks: sentence planning and surface realisation from non-aligned data.

In this challenge[1], (Novikova et al. 2016, 2017) provide a new crowd-sourced data set of 50k instances in the restaurant domain. Each example consists of a dialogue act-based meaning representation (MR) and 8.1 references in natural language. The primary task which the submitters should follow is to generate an utterance from a given MR, which is a) similar to human-generated reference texts, and b) highly rated by humans.

The primary purpose of this article is to present our system called structure-based generation system (SBG System) for Challenge[1]. The input of SBG System is each MR. The output of SBG System is the corresponding Natural Language (NL) Reference for each MR. Based on the traditional approach (Reiter and Dale 1997), our system performs two main sub-tasks: (i) sentence planning in which we create the overall sentence structures and determine the appropriate structure for each input MR; (ii) surface realization in which we identify the exact word forms and linearize the structure into a string.

The rest of article is separated as follows. We introduce the generation setting in Section 2 and describe our generator architecture in Section 3. Section 4 details the experiments and analyzes the results. We offer conclusions in Section 5.

2 Generator Setting

The input to our generator is predicates of an MR entry from crowd-sourced dataset[1]. Following the traditional architecture, our generator operates in two levels, producing structure-type of the output sentences and the natural language strings (see Table. 1). The first level corresponds to the sentence planning NLG stage. At this stage, our generator decides the structure-type of the output sentences. At The second level, our generator corre-

[1] http://www.macs.hw.ac.uk/InteractionLab/E2E/

sponds to the surface realization NLG stage, producing the final natural language reference.

Flat MR	name[The Cricketers], eatType[restaurant], food[chinese], priceRange[less than £20], customer rating[low], area[city centre], familyFriendly[yes], near[All Bar One]
Structure	{name[] is a} {eatType[]} {providing food[]} {in the priceRange[]} {.} {It is located in the area[]} {.} {It is near near[]} {.} {Its customer rating is customer rating[]} {.}
NL Reference	The Cricketers is a restaurant providing Chinese food in the less than £20. It is located in the city centre. It is near All Bar One. Its customer rating is low.

Table 1: An example of a 3-tube <Flat MR – Structure – NL Reference>.

The structure-type of an NL reference is generated from the general graph of predicate relationships (see Fig. 1).

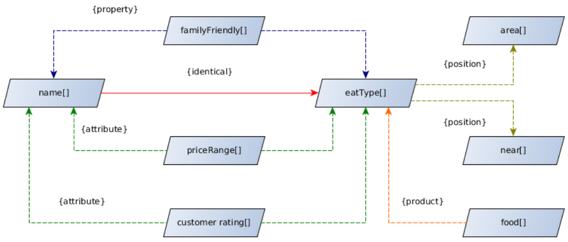


Figure 1: General graph of predicate relationships.

3 The Structure-based Generation System

Based on the traditional approach when building a NLG system, SBG System consists of three main components operating main processes. The first component corresponds to construct the general structures from the graph of predicate relationships. At the second component, the primary process is to collect the appropriate English words and phrases for each value of each predicate. The third component corresponds to two primary operations: (i) generate the suitable structure from input MR; (ii) complete the final reference.

The general architecture of SBG System is shown in Fig. 2.

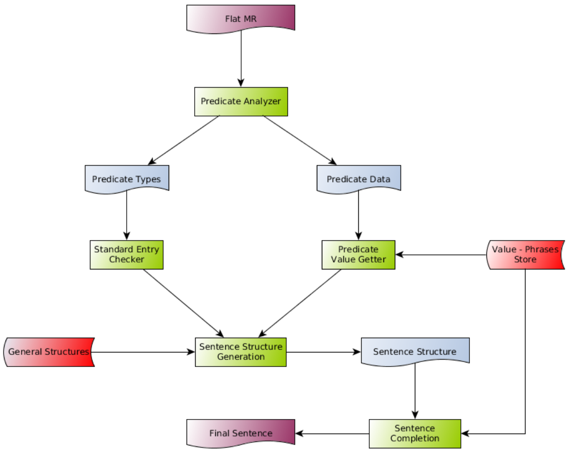


Figure 2: The general design of SBG System.

3.1 Structure Builder

According to crowd-sourced dataset[1], there are eight types of the predicate, as an example in Table 1. We classify into five groups based on their pragmatic meaning and define the corresponding relationships between them (see Fig. 1 and Table 2).

Group	Predicate, Meaning and Relationships
1	• Predicate: name[], eatType[] • Meaning: the main object • Relationship: <identical> – name[] {is} eatType[]
2	• Predicate: area[], near[] • Meaning: place • Relationship: main object ← {position}
3	• Predicate: food[] • Meaning: production • Relationship: main object ← {product}
4	• Predicate: familyFriendly[] • Meaning: experience • Relationship: main object ← {property}
5	• Predicate: priceRange[], customer rating[] • Meaning: attribute • Relationship: main object ← {attribute}

Table 2: Groups of predicates.

After analyzing the relationships between groups in Fig.1 and Table 2, we also apply knowledge about English clause structures in linguistic theory Functional Grammar (FG - Halliday and Matthiessen 2004) to form the basic structure for all NL references (see Fig. 2). Note that due to there is only one object, therefore we use the pronoun "it" to refer to this object in the structure.

{name[] is} {eatType[]} {.} {It provides} {food[]} {.} {It has} {priceRange[]} {.} {It has} {customer rating[]} {.} {It is located in} {area[]} {.} {It is near} {near[]} {.}

Figure 3: Basic structure for all NL references.

At the next step, we modify the basic structure in Fig. 3 and create new general structures with following actions. The first action is to change the positions of predicate elements. The idea of this action is to consider the possible grammatical role of each predicate, according to its pragmatic meaning and relationship. The second action is to use sophisticated phrase structures in FG. The third action is to apply the transformation rules for sentences having the same meaning in Transformational-Generative Grammar (TGG - Chomsky 2002) as well as an idea in (Tran 2011).

3.2 Data Source Collector

One of the most challenging tasks in a NLG system is to select the appropriate words and phrases for the surface realization stage. We deal with this task by operating two steps.

At the first step, we analyze each predicate to determine elements: type; value; phrase. As an example, consider the Flat MR in Table 1, we have pairs of types (outside the brackets "[]") and corresponding values (inside the brackets "[]"): "name[]" – "The Cricketers"; "eatType[]" – "restaurant"; "food[]" – "chinese"; "priceRange[]" – "less than £20"; "customer rating[]" – "low"; "area[]" – "city centre"; "familyFriendly[]" – "yes"; "near[]" – "All Bar One".

As in Fig. 4, each predicate has one type (e.g. food[]) and several values (e.g. Italian, Chinese), in which each value has several corresponding real phrases.

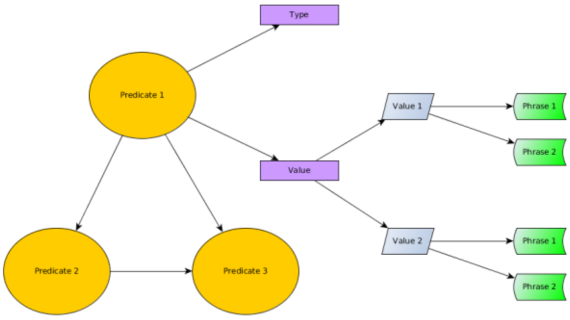
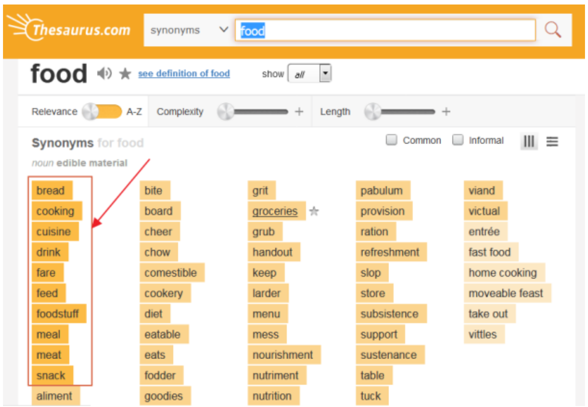


Figure 4: Predicate description.

At the second step, we use words in value and type elements as the keywords and collect the synonyms which have the most similar meaning in thesaurus website[2,3]. Another way to collect the similar phrases is that we apply different phrase structures in TGG as well as collect from crowd-sourced dataset[1]. As an example, consider predicate "food[]" having values "Italian", we collect the synonyms as in Fig. 5. We then combine with values of this predicate to create a list of phrases: {Italian food; Italian cuisine; Italian meals,…}.



Synonyms for food

noun edible material

bread
cooking
cuisine
drink
fare
feed
foodstuff
meal
meat
snack
aliment
bite
board
cheer
chow
comestible
cookery
diet
eatable
eats
fodder
goodies
grit
groceries
grub
handout
keep
larder
menu
mess
nourishment
nutriment
nutrition
pabulum
provision
ration
refreshment
slop
store
subsistence
support
sustenance
table
tuck
viand
victual
entrée
fast food
home cooking
moveable feast
take out
vittles

Figure 5: Synonyms of "food".

In Table 3, we present the example values and corresponding phrases for each predicate.

Predicate Type	Example Values	Example Phrases
name[]	Alimentum	Alimentum
	Aromi	Aromi
eatType[]	pub	pub
	restaurant	restaurant
food[]	Chinese	Chinese food; Chinese cuisine;
	Italian	Italian food; Italian cuisine;
priceRange[]	high	high price range; price range of high;
	less than £20	price range of less than £20; lessd than £20 price range;
customer rating[]	high	high customer ratings; customer ratings are high;
	1 out of 5	customer rating of 1 out

[2] http://www.thesaurus.com

[3] https://www.phrases.org.uk/phrase-thesaurus/index.html

		of 5; 1 out of 5 customer rating;
area[]	riverside	riverside
	city centre	city centre
familyFriendly[]	yes	family friendly yes; kid-friendly;
	no	non-family-friendly; not kid-friendly;
near[]	The Bakers	The Bakers
	The Rice Boat	The Rice Boat

Table 3: Example values and phrases of each predicate.

3.3 Reference Generator

As illustrating in Fig. 2, we perform the reference generator component with following steps:

- **Step 1**. We analyze the input MR entry to determine: entry in reduced-type (contains predicates without corresponding value); the corresponding value of each predicate.
- **Step 2**. We propose rules for generating the appropriate structure of the output NL reference. With each rule, we use existing predicates and corresponding values as two constraint factors to find the proper general structure and modify this to generate the final structure.
- **Step 3**. We replace the elements in the structure by appropriate phrases from data source collector component.

The final result after operating the above three steps is the NL reference of the SBG system.

4 Experiment and Evaluation

According to Challenge[1], to measure the scores, we used four metrics[4]: BLEU (Papineni et al. 2002), NIST (Doddington 2002), METEOR (Lavie and Agarwal 2007), ROUGE-L (Lin 2004), CIDEr (Vedantam et al. 2015). For the comparison, to establish a baseline on the task data, we also use Tgen[5] (Dusek and Jurcicek 2016a), one of the famous E2E data-driven systems. TGen is based on sequence-to-sequence modelling with attention (seq2seq) (Bahdanau et al. 2015).

We test and evaluate our system on two sections: the development and real e2e experiment.

4.1 Testing and Evaluating in Development Section

The development section is built for preliminary testing. This section includes 547 entries in original type (contains predicates with corresponding values) or 25 entries in reduced type (contains predicates without corresponding value). Each entry consists of a different number of predicates (from 3 to 8 predicates).

With the development section, we only apply automatic evaluation. The results are shown in Table 4. There we can see that, with the development section, our system surpasses the baseline at ROUGE-L and CIDEr scores. However, the BLEU and NIST scores of our system are lower than the baseline's.

Metric	**SBG System Value**	**Baseline Value**
BLEU	0.6828	0.6904
NIST	8.3052	8.4529
ROUGE-L	0.730	0.726
CIDEr	2.465	2.403

Table 4: Automatic evaluation results when testing SBG System on the development section.

4.2 Testing and Evaluating with E2E Experiment Section

The real e2e experiment section includes 630 entries in original type (contains predicates with corresponding values). Each entry consists of a different number of predicates (from 3 to 8 predicates). With this section, the Organising Committee[1] test and evaluate in two steps: automatic evaluation and human evaluation (the full results can be found in Challenge[1]).

At the automatic evaluation step, the score results when comparing our SBG system with the baseline are shown in Table 5.

Metric	**SBG System Value**	**Baseline Value**
BLEU	0.599	0.6593
NIST	7.9277	8.6094
METEOR	0.4346	0.4483
ROUGE-L	0.6634	0.685
CIDEr	2.0783	2.2338

Table 5: SBG System results in the experiment section.

[4] According to (Novikova et al., 2017), we used MT-Eval script (BLEU, NIST) and the COCO Caption (Chen et al., 2015) metrics (METEOR, ROUGE- L, CIDEr). https://github.com/tuetschek/e2e-metrics

[5] TGen is freely available at https://github.com/UFAL-DSG/tgen

At the human evaluation, the Organising Committee[1] using TrueSkill algorithm (Sakaguchi et al. 2014) to calculate the scores. The full results can be found at (Ondrej Dusek et al. 2018) and Challenge[1]. They compare 20 primary systems and the baseline using the CrowdFlower platform. There is two rank types: (i) Quality is defined as the overall quality of the utterance, would be considered the primary measure; (ii) Naturalness has defined the extent to which a native speaker could have produced the utterance.

With the corresponding TrueSkill final scores, 20 primary systems are ordered by ranges and grouped into 5 clusters from best to worst. The systems in the same cluster are considered to show the similar performance and share the same position. Due to there are two rank types are *Quality* and *Naturalness*, there are also two ways of clustering, which means one system can be in one cluster according to Quality scores and in another cluster according to Naturalness scores.

According to the final results, our SBG system is in cluster 2 in both ways, which mean our system is the second best (same as other systems in cluster 2) according to both Quality and Naturalness scores. Table 6 and 7, in turn, present the Quality and Naturalness scores of the highest system in each cluster, baseline and our SBG system.

Cluster / Position	**True-Skill**	**Range**	**System**
1	0.300	(1-1)	<anonymous 2> – <anonymous 2>
2	0.228	(2-4)	UKP-TUDA – ukp-tuda
	0.184	**(3-5)**	**SBG System – test_e2e_result_2 final_TSV**
	0.184	*(3-6)*	*BASELINE – baseline*
3	-0.078	(15-16)	Thomson Reuters NLG – Primary_2_test_train_dev
4	-0.152	(17-19)	<anonymous 5> – primary_submission-temperature_1.1
5	-0.426	(20-21)	Chen Shuang – Primary_NonAbstract-beam1

Table 6: The Quality Scores of Highest Systems in Each Cluster, Baseline and Our SBG System.

Cluster / Position	**True-Skill**	**Range**	**System**
1	0.211	(1-1)	Sheffield NLP – sheffield_primarySystem2_var1
2	0.171	(2-3)	<anonymous 2> – <anonymous 2>
	0.101	*(4-8)*	*BASELINE – baseline*
	0.091	**(5-8)**	**SBG System – test_e2e_result_2 final_TSV**
3	-0.053	(13-16)	Thomson Reuters NLG – Primary_1_submission_6_beam
4	-0.144	(18-19)	FORGe – E2E_UPF_1
5	-0.243	(20-21)	Thomson Reuters NLG – Primary_2_test_train_dev

Table 7: The Naturalness Scores of Highest Systems in Each Cluster, Baseline and Our SBG System.

The testing results show that our SBG system generates good quality references from meaning representations in both development and real e2e experiment sections. Based on cursory checks, our system was able to create long, grammatical, meaningful, multi-sentence output, as illustrated by the following example: "*The Cricketers is a restaurant providing Chinese food in the less than £20. It is located in the city centre. It is near All Bar One. Its customer rating is low.*".

5 Conclusion

We have presented a structure-based method for generating natural language references from restaurant-domain meaning representations dataset[1]. Our generation system followed traditional approach with two main sub-tasks: (i) create the overall sentence structures which are called sentence planning; (ii) determine the exact word forms and linearize the structure into a string which is called surface realization. The experiment results with both automatic and human evaluation show that our method overcomes the challenges from E2E dataset[1]: (i) references have lexical richness, syntactic variation and discourse phenomena; (ii) generating systems should have a content selection.

In future works, we intend to apply more knowledge in linguistic theories, e.g. TGG and FG, to improve the quality and naturalness of generated sentences. Besides, we expand our method and test with other datasets for a broader comparison. Also, we hope to apply the idea in SBG method for other NLP field, e.g. summarization (Tran and Nguyen 2015, 2016).

References

Dzmitry Bahdanau, Kyunghyun Cho and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations*. San Diego, CA, USA.

David L. Chen and Raymond J. Mooney. 2008. Learning to sportscast: A test of grounded language acquisition. In *Proceedings of the 25th international conference on Machine learning (ICML)*. Helsinki, Finland, pages 128–135.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollar, and C. Lawrence Zitnick. 2015. *Microsoft COCO Captions: Data Collection and Evaluation Server*.

Noam Chomsky. 2002. *Syntactic Structures*, Second Edition. Mouton de Gruyter.

Vera Demberg and Johanna D Moore. 2006. Information presentation in spoken dialogue systems. In *Proceedings of the 11th Conference of the European Chapter of the ACL (EACL)*. pages 65–72.

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram cooccurrence statistics. In *Proceedings of the Second International Conference on Human Language Technology Research*. San Diego, CA, USA, pages 138–145.

Ondrej Dusek, Jekaterina Novikova and Verena Rieser. 2018. *Findings of the E2E NLG challenge*.

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*. Beijing, China and Association for Computational Linguistics, pages 451–461.

Ondrej Dusek and Filip Jurcicek. 2016a. Sequence-to-Sequence Generation for Spoken Dialogue via Deep Syntax Trees and Strings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. Berlin, Germany and Association for Computational Linguistics, pages 45–51.

Ondrej Dusek and Filip Jurcicek. 2016b. A context-aware natural language generator for dialogue systems. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Los Angeles, CA, USA and Association for Computational Linguistics, pages 185–190.

Michael Halliday and Christian Matthiessen. 2004. *An Introduction to Functional Grammar*, Third Edition, Hodder Arnold.

Gerasimos Lampouras and Andreas Vlachos. 2016. Imitation learning for language generation from unaligned data. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. Osaka, Japan, pages 1101–1112.

Alon Lavie and Abhaya Agarwal. 2007. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Prague, Czech Republic, pages 228–231.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL- 04 workshop*. Barcelona, Spain, pages 74–81.

Francois Mairesse, Milica Gasic, Filip Jurcicek, Simon Keizer, Blaise Thomson, Kai Yu, and Steve Young. 2010. Phrase-based statistical language generation using graphical models and active learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Uppsala, Sweden, pages 1552–1561.

Hongyuan Mei, Mohit Bansal and Matthew R. Walter. 2016. What to talk about and how? Selective Generation using LSTMs with Coarse-to-Fine Alignment. In *Proceedings of NAACL-HLT 2016*. San Diego, California and Association for Computational Linguistics, pages 720–730.

Jekaterina Novikova, Ondrej Dusek and Verena Rieser. 2017. The E2E Dataset: New Challenges For End-to-End Generation. In *Proceedings of the 18th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Saarbrucken, Germany and Association for Computational Linguistics, pages 201–206.

Jekaterina Novikova, Oliver Lemon and Verena Rieser. 2016. Crowd-sourcing NLG Data: Pictures Elicit Better Data. In *Proceedings of The 9th International Natural Language Generation conference*. Edinburgh, UK and Association for Computational Linguistics, pages 265–273.

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*. Association for Computational Linguistics, Philadelphia, PA, USA, pages 311–318.

Ehud Reiter and Robert Dale. 1997. *Building Natural Language Generation System*. Cambridge University Press.

Verena Rieser, Oliver Lemon, and Simon Keizer. 2014. Natural language generation as incremental planning under uncertainty: Adaptive information presentation for statistical dialogue systems. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 22(5):979–993.

Keisuke Sakaguchi, Matt Post and Benjamin Van Durme. 2014. Efficient Elicitation of Annotations for Human Evaluation of Machine Translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Baltimore, Maryland USA and Association for Computational Linguistics, pages 1–11.

Amanda Stent, Rashmi Prasad, and Marilyn Walker. 2004. Trainable sentence planning for complex information presentations in spoken dialog systems. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics*. Barcelona, Spain, pages 79–86.

Trung Tran. 2011. *Phương pháp xác định những câu hỏi tương đương nghĩa cho hệ thống tìm kiếm thư viện bằng truy vấn tiếng Việt* [The method of identifying questions having the equivalent meaning for the library finding system by Vietnamese queries]. Master Thesis. University of Information Technology, VNU-HCM, Vietnam.

Trung Tran and Dang Tuan Nguyen. 2015. Modelling Consequence Relationships between Two Action, State or Process Vietnamese Sentences for Improving the Quality of New Meaning-Summarizing Sentence. *International Journal of Pervasive Computing and Communications* 11(2):169–190.

Trung Tran and Dang Tuan Nguyen. 2016. Algorithm of Computing Verbal Relationships for Generating Vietnamese Paragraph of Summarization from The Logical Expression of Discourse Representation Structure. *Vietnam Journal of Computer Science* 3(1):35–46.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. CIDEr: Consensus-based image description evaluation. In *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Boston, MA, USA, pages 4566–4575.

Marilyn A Walker, Stephen J Whittaker, Amanda Stent, Preetam Maloor, Johanna Moore, Michael Johnston, and Gunaranjan Vasireddy. 2004. Generation and evaluation of user tailored responses in multi-modal dialogue. *Cognitive Science* 28(5):811–840.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke and Steve Young. 2015. Semantically Conditioned LSTM-based Natural Language Generation for Spoken Dialogue Systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal and Association for Computational Linguistics, pages 1711–1721.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal, pages 1711–1721.