Finding Common Ground: Towards a Surface Realisation Shared Task

Anja Belz
Natural Language Technology Group
Computing, Mathematical and Information Sciences
University of Brighton, Brighton BN2 4GJ, UK
a.s.belz@brighton.ac.uk

Mike White
Department of Linguistics
The Ohio State University
Columbus, OH, USA
mwhite@ling.osu.edu

Josef van Genabith and **Deirdre Hogan**
National Centre for Language Technology
School of Computing
Dublin City University
Dublin 9, Ireland
{dhogan,josef}@computing.dcu.ie

Amanda Stent
AT&T Labs Research, Inc.,
180 Park Avenue
Florham Park, NJ 07932, USA
stent@research.att.com

Abstract

In many areas of NLP reuse of utility tools such as parsers and POS taggers is now common, but this is still rare in NLG. The subfield of surface realisation has perhaps come closest, but at present we still lack a basis on which different surface realisers could be compared, chiefly because of the wide variety of different input representations used by different realisers. This paper outlines an idea for a shared task in surface realisation, where inputs are provided in a common-ground representation formalism which participants map to the types of input required by their system. These inputs are derived from existing annotated corpora developed for language analysis (parsing etc.). Outputs (realisations) are evaluated by automatic comparison against the human-authored text in the corpora as well as by human assessors.

1 Background

When reading a paper reporting a new NLP system, it is common these days to find that the authors have taken an NLP utility tool off the shelf and reused it. Researchers frequently reuse parsers, POS-taggers, named entity recognisers, coreference resolvers, and many other tools. Not only is there a real choice between a range of different systems performing the same task, there are also evaluation methodologies to help determine what the state of the art is.

Natural Language Generation (NLG) has not so far developed generic tools and methods for comparing them to the same extent as Natural Language Analysis (NLA) has. The subfield of NLG that has perhaps come closest to developing generic tools is surface realisation. Wide-coverage surface realisers such as PENMAN/NIGEL (Mann and Mathiesen, 1983), FUF/SURGE (Elhadad and Robin, 1996) and REALPRO (Lavoie and Rambow, 1997) were intended to be more or less off-the-shelf plug-and-play modules. But they tended to require a significant amount of work to adapt and integrate, and required highly specific inputs incorporating up to several hundred features that needed to be set.

With the advent of statistical techniques in NLG surface realisers appeared for which it was far simpler to supply inputs, as information not provided in the inputs could be added on the basis of likelihood. An early example, the Japan-Gloss system (Knight et al., 1995) replaced PENMAN's default settings with statistical decisions. The Halogen/Nitrogen developers (Langkilde and Knight, 1998a) allowed inputs to be arbitrarily underspecified, and any decision not made before the realiser was decided simply by highest likelihood according to a language model, automatically trainable from raw corpora.

The Halogen/Nitrogen work sparked an interest in statistical NLG which led to a range of surface realisation methods that used corpus frequencies in one way or another (Varges and Mellish, 2001; White, 2004; Velldal et al., 2004; Paiva and Evans, 2005). Some surface realisation work looked at directly applying statistical models during a linguistically informed generation process to prune

the search space (White, 2004; Carroll and Oepen, 2005).

While statistical techniques have led to realisers that are more (re)usable, we currently still have no way of determining what the state of the art is. A significant subset of statistical realisation work (Langkilde, 2002; Callaway, 2003; Nakanishi et al., 2005; Zhong and Stent, 2005; Cahill and van Genabith, 2006; White and Rajkumar, 2009) has recently produced results for regenerating the Penn Treebank. The basic approach in all this work is to remove information from the Penn Treebank parses (the word strings themselves as well as some of the parse information), and then convert and use these underspecified representations as inputs to the surface realiser whose task it is to reproduce the original treebank sentence. Results are typically evaluated using BLEU, and, roughly speaking, BLEU scores go down as more information is removed.

While publications of work along these lines do refer to each other and (tentatively) compare BLEU scores, the results are not in fact directly comparable, because of the differences in the input representations automatically derived from Penn Treebank annotations. In particular, the extent to which they are underspecified varies from one system to the next.

The idea we would like to put forward with this short paper is to develop a shared task in surface realisation based on common inputs and annotated corpora of paired inputs and outputs derived from various resources from NLA that build on the Penn Treebank. Inputs are provided in a common-ground representation formalism which participants map to the types of input required by their system. These inputs are automatically derived from the Penn Treebank and the various layers of annotation (syntactic, semantic, discourse) that have been developed for the documents in it. Outputs (realisations) are evaluated by automatic comparison against the human-authored text in the corpora as well as by by human assessors.

In the short term, such a shared task would make existing and new approaches directly comparable by evaluation on the benchmark data associated with the shared task. In the long term, the common-ground input representation may lead to a standardised level of representation that can act as a link between surface realisers and preceding modules, and can make it possible to use alternative surface realisers as drop-in replacements for each other.

2 Towards Common Inputs

One hugely challenging aspect in developing a Surface Realisation task is developing a common input representation that all, or at least a majority of, surface realisation researchers are happy to work with. While many different formalisms have been used for input representations to surface realisers, one cannot simply use e.g. van Genabith et al.'s automatically generated LFG f-structures, White et al's CCG logical forms, Nivre's dependencies, Miyao et al.'s HPSG predicate-argument structures or Copestake's MRSs etc., as each of them would introduce a bias in favour of one type of system.

One possible solution is to develop a meta-representation which contains, perhaps on multiple layers of representation, all the information needed to map to any of a given set of realiser input representations, a common-ground representation that acts as a kind of interlingua for translating between different input representations.

An important issue in deriving input representations from semantically, syntactically and discourse-annotated corpora is deciding what information *not* to include. A concern is that making such decisions by committee may be difficult. One way to make it easier might be to define several versions of the task, where each version uses inputs of different levels of specificity.

Basing a common input representation on what can feasibly be obtained from non-NLG resources would put everyone on reasonably common footing. If, moreover, the common input representations can be automatically derived from annotations in existing resources, then data can be produced in sufficient quantities to make it feasible for participants to automatically learn mappings from the system-neutral input to their own input.

The above could be achieved by doing something along the lines of the CoNLL'08 shared task on Joint Parsing of Syntactic and Semantic Dependencies, for which the organisers combined the Penn Treebank, Propbank, Nombank and the BBN Named Entity corpus into a dependency representation. Brief descriptions of these resources and more details on this idea are provided in Section 4 below.

3 Evaluation

As many NLG researchers have argued, there is usually not a single right answer in NLG, but various answers, some better than others, and NLG tasks should take this into account. If a surface realisation task is focused on single-best realizations, then it will not encourage research on producing all possible good realizations, or multiple acceptable realizations in a ranked list, etc. It may not be the best approach to encourage systems that try to make a single, safe choice; instead, perhaps one should encourage approaches that can tell when multiple choices would be ok, and if some would be better than others.

In the long term we need to develop task definitions, data resources and evaluation methodologies that properly take into account the one-to-many nature of NLG, but in the short term it may be more realistic to reuse existing non-NLG resources (which do not provide alternative realisations) and to adapt existing evaluation methodologies including intrinsic assessment of Fluency, Clarity and Appropriateness by trained evaluators, and automatic intrinsic methods such as BLEU and NIST. One simple way of adapting the latter, for example, could be to calculate scores for the n best realisations produced by a realiser and then to compute a weighted average where scores for realisations are weighted in inverse proportion to the ranks given to the realisations by the realiser.

4 Data

There is a wide variety of different annotated resources that could be of use in a shared task in surface realisation. Many of these include documents originally included in the Penn Treebank, and thus make it possible in principle to combine the various levels of annotation into a single common-ground representation. The following is a (non-exhaustive) list of such resources:

1. Penn Treebank-3 (Marcus et al., 1999): one million words of hand-parsed 1989 Wall Street Journal material annotated in Treebank II style. The Treebank bracketing style allows extraction of simple predicate/argument structure. In addition to Treebank-1 material, Treebank-3 contains documents from the Switchboard and Brown corpora.

2. Propbank (Palmer et al., 2005): This is a semantic annotation of the Wall Street Journal section of Penn Treebank-2. More specifically, each verb occurring in the Treebank has been treated as a semantic predicate and the surrounding text has been annotated for arguments and adjuncts of the predicate. The verbs have also been tagged with coarse grained senses and with inflectional information.

3. NomBank 1.0 (Meyers et al., 2004): NomBank is an annotation project at New York University that provides argument structure for common nouns in the Penn Treebank. NomBank marks the sets of arguments that occur with nouns in PropBank I, just as the latter records such information for verbs.

4. BBN Pronoun Coreference and Entity Type Corpus (Weischedel and Brunstein, 2005): supplements the Wall Street Journal corpus, adding annotation of pronoun coreference, and a variety of entity and numeric types.

5. FrameNet (Johnson et al., 2002): 150,000 sentences annotated for semantic roles and possible syntactic realisations. The annotated sentences come from a variety of sources, including some PropBank texts.

6. OntoNotes 2.0 (Weischedel et al., 2008): OntoNotes 1.0 contains 674k words of Chinese and 500k words of English newswire and broadcast news data. OntoNotes follows the Penn Treebank for syntax and PropBank for predicate-argument structure. Its semantic representation will include word sense disambiguation for nouns and verbs, with each word sense connected to an ontology, and coreference. The current goal is to annotate over a million words each of English and Chinese, and half a million words of Arabic over five years.

There are other resources which may be useful. Zettelmoyer and Collins (2009) have manually converted the original SQL meaning annotations of the ATIS corpus (et al., 1994)—some 4,637 sentences—into lambda-calculus expressions which were used for training and testing their semantic parser. This resource might make a good out-of-domain test set for generation systems trained on WSJ data.

FrameNet, used for semantic parsing, see for example Gildea and Jurafsky (2002), identifies a

sentence's frame elements and assigns semantic roles to the frame elements. FrameNet data (Baker and Sato, 2003) was used for training and test sets in one of the SensEval-3 shared tasks in 2004 (Automatic Labeling of Semantic Roles). There has been some work combining FrameNet with other lexical resources. For example, Shi and Mihalcea (2005) integrated FrameNet with VerbNet and WordNet for the purpose of enabling more robust semantic parsing.

The Semlink project (http://verbs.colorado.edu/semlink/) aims to integrate Propbank, FrameNet, WordNet and VerbNet.

Other relevant work includes Moldovan and Rus (Moldovan and Rus, 2001; Rus, 2002) who developed a technique for parsing into logical forms and used this to transform WordNet concept definitions into logical forms. The same method (with additional manual correction) was used to produce the test set for another SensEval-3 shared task (Identification of Logic Forms in English).

4.1 CoNLL 2008 Shared Task Data

Perhaps the most immediately promising resource is is the CoNLL shared task data from 2008 (Surdeanu et al., 2008) which has syntactic dependency annotations, named-entity boundaries and the semantic dependencies model roles of both verbal and nominal predicates. The data consist of excerpts from Penn Treebank-3, BBN Pronoun Coreference and Entity Type Corpus, PropBank I and NomBank 1.0. In CoNLL '08, the data was used to train and test systems for the task of producing a joint semantic and syntactic dependency analysis of English sentences (the 2009 CoNLL Shared Task extended this to multi-lingual data).

It seems feasible that we could reuse the CoNLL data for a prototype Surface Realisation task, adapting it and inversing the direction of the task, i.e. mapping from syntactic-semantic dependency representations to word strings.

5 Developing the Task

The first step in developing a Surface Realisation task could be to get together a working group of surface realisation researchers to develop a common-ground input representation automatically derivable from a set of existing resources. As part of this task a prototype corpus exemplifying inputs/outputs and annotations could be developed. At the end of this stage it would be useful to write a white paper and circulate it and the prototype corpus among the NLG (and wider NLP) community for feedback and input.

After a further stage of development, it may be feasible to run a prototype surface realisation task at Generation Challenges 2011, combined with a session for discussion and roadmapping. Depending on the outcome of all of this, a full-blown task might be feasible by 2012. Some of this work will need funding to be feasible, and the authors of this paper are in the process of applying for financial support for these plans.

6 Concluding Remarks

In this paper we have provided an overview of existing resources that could potentially be used for a surface realisation task, and have outlined ideas for how such a task might work. The core idea is to develop a common-ground input representation which participants map to the types of input required by their system. These inputs are derived from existing annotated corpora developed for language analysis. Outputs (realisations) are evaluated by automatic comparison against the human-authored text in the corpora as well as by by human assessors. Evaluation methods are adapted to take account of the one-to-many nature of the realisation mapping.

The ideas outlined in this paper began as a prolonged email exchange, interspersed with discussions at conferences, among the authors. This paper summarises our ideas as they have evolved so far, to enable feedback and input from other researchers interested in this type of task.

References

Colin F. Baker and Hiroaki Sato. 2003. The framenet data and software. In *Proceedings of ACL'03*.

A. Cahill and J. van Genabith. 2006. Robust PCFG-based generation using automatically acquired LFG approximations. In *Proc. ACL'06*, pages 1033–44.

Charles Callaway. 2003. Evaluating coverage for large symbolic NLG grammars. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI 2003)*, pages 811–817.

J. Carroll and S. Oepen. 2005. High efficiency realization for a wide-coverage unification grammar. In *Proceedings of the 2nd International Joint Conference on Natural Language Processing (IJCNLP'05)*, volume 3651, pages 165–176. Springer Lecture Notes in Artificial Intelligence.

M. Elhadad and J. Robin. 1996. An overview of SURGE: A reusable comprehensive syntactic realization component. Technical Report 96-03, Dept of Mathematics and Computer Science, Ben Gurion University, Beer Sheva, Israel.

Deborah Dahl et al. 1994. Expanding the scope of the ATIS task: the ATIS-3 corpus. In *Proceedings of the ARPA HLT Workshop*.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.

C. Johnson, C. Fillmore, M. Petruck, C. Baker, M. Ellsworth, J. Ruppenhoper, and E.Wood. 2002. Framenet theory and practice. Technical report.

K. Knight, I. Chander, M. Haines, V. Hatzivassiloglou, E. Hovy, M. Iida, S. Luk, R. Whitney, and K. Yamada. 1995. Filling knowledge gaps in a broad-coverage MT system. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI '95)*, pages 1390–1397.

I. Langkilde and K. Knight. 1998a. Generation that exploits corpus-based statistical knowledge. In *Proc. COLING-ACL*. http://www.isi.edu/licensed-sw/halogen/nitro98.ps.

I. Langkilde. 2002. An empirical verification of coverage and correctness for a general-purpose sentence generator. In *Proc. 2nd International Natural Language Generation Conference (INLG '02)*.

B. Lavoie and O. Rambow. 1997. A fast and portable realizer for text generation systems. In *Proceedings of the 5th Conference on Applied Natural Language Processing (ANLP'97)*, pages 265–268.

W. Mann and C. Mathiesen. 1983. NIGEL: A systemic grammar for text generation. Technical Report ISI/RR-85-105, Information Sciences Institute.

Mitchell P. Marcus, Beatrice Santorini, Mary Ann Marcinkiewicz, and Ann Taylor. 1999. Treebank-3. Technical report, Linguistic Data Consortium, Philadelphia.

Adam Meyers, Ruth Reeves, and Catherine Macleod. 2004. Np-external arguments a study of argument sharing in english. In *MWE '04: Proceedings of the Workshop on Multiword Expressions*, pages 96–103, Morristown, NJ, USA. Association for Computational Linguistics.

Dan I. Moldovan and Vasile Rus. 2001. Logic form transformation of wordnet and its applicability to question answering. In *Proceedings of ACL'01*.

Hiroko Nakanishi, Yusuke Miyao, and Jun'ichi Tsujii. 2005. Probabilistic models for disambiguation of an hpsg-based chart generator. In *Proceedings of the 9th International Workshop on Parsing Technology (Parsing'05)*, pages 93–102. Association for Computational Linguistics.

D. S. Paiva and R. Evans. 2005. Empirically-based control of natural language generation. In *Proceedings ACL'05*.

M. Palmer, P. Kingsbury, and D. Gildea. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Vasile Rus. 2002. *Logic Form For WordNet Glosses and Application to Question Answering*. Ph.D. thesis.

Lei Shi and Rada Mihalcea. 2005. Putting pieces together: Combining framenet, verbnet and wordnet for robust semantic parsing. In *Proceedings of CICLing'05*.

Mihai Surdeanu, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. 2008. The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. In *CoNLL '08: Proceedings of the Twelfth Conference on Computational Natural Language Learning*, pages 159–177.

S. Varges and C. Mellish. 2001. Instance-based natural language generation. In *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL '01)*, pages 1–8.

E. Velldal, S. Oepen, and D. Flickinger. 2004. Paraphrasing treebanks for stochastic realization ranking. In *Proceedings of the 3rd Workshop on Treebanks and Linguistic Theories (TLT '04)*, Tuebingen, Germany.

Ralph Weischedel and Ada Brunstein. 2005. Bbn pronoun coreference and entity type corpus. Technical report, Linguistic Data Consortium.

Ralph Weischedel et al. 2008. Ontonotes release 2.0. Technical report, Linguistic Data Consortium.

Michael White and Rajakrishnan Rajkumar. 2009. Perceptron reranking for ccg realisation. In *Proceedings of the 2009 Conference on Empririal Methods in Natural Language Processing (EMNLP'09)*, pages 410–419.

M. White. 2004. Reining in CCG chart realization. In A. Belz, R. Evans, and P. Piwek, editors, *Proceedings INLG'04*, volume 3123 of *LNAI*, pages 182–191. Springer.

Luke Zettlemoyer and Michael Collins. 2009. Learning context-dependent mappings from sentences to logical forms. In *Proceedings of ACL-IJCNLP'09*.

H. Zhong and A. Stent. 2005. Building surface realizers automatically from corpora. In A. Belz and S. Varges, editors, *Proceedings of UCNLG'05*, pages 49–54.