The Use of a Structural N-gram Language Model in Generation-Heavy Hybrid Machine Translation

Nizar Habash

University of Maryland University Institute for Advanced Computer Studies
University of Maryland College Park
habash@umiacs.umd.edu

Abstract. This paper describes the use of a statistical structural N-gram model in the natural language generation component of a Spanish-English generation-heavy hybrid machine translation system. A structural N-gram model captures the relationship between words in a dependency representation without taking into account the overall structure at the phrase level. The model is used together with other components in the system for lexical and structural selection. An evaluation of the machine translation system shows that the use of structural N-grams decreases runtime by 60% with no loss in translation quality.

1 Introduction

Statistical N-gram models capturing patterns of local co-occurrence of contiguous words in sentences have been used in various hybrid implementations of Natural Language Generation (NLG) and Machine Translation (MT) systems [1–5]. Other types of language models that capture long-distance relationships, such as probabilistic context-free grammars (PCFG) or lexicalized syntax models, have been used in the parsing community with impressive improvements in parsing correctness [6–9]. In comparison, only one large-scale system built with NLG in mind uses a structural language model [4]. Additionally, the IBM Air Travel Reports system, which implements a dependency n-gram model, uses templates and focuses on travel reports only [10]. A recent study using the Charniak parser [11] as a lexicalized syntax model for generation purposes demonstrated the usefulness of these models in a variety of NLG tasks [12].

The focus of this paper is on the contributions of a specific type of a structural language model, Structural N-grams (SN-gram model)[1], for NLG in the MT context. Whereas syntax models address both parent-child relationships and sisterhood relationships, the SN-gram model characterizes the relationship between words in a dependency representation of a sentence without taking into account the overall structure at the phrase level. In other words, an independence in the behavior of the children relative to each other (their sisterhood relationships) is assumed in SN-grams.

Figure 1 exemplifies the differences between SN-grams (dashed arrows) and N-grams (solid arrows). In addition to capturing long-distance relationships between words (e.g., $have/has$ and $lining$), SN-grams are based on uninflected lexemes not on inflected surface forms. Therefore SN-grams can model more general relationships between lexical items. Moreover, SN-grams' effect is only seen on lexical selection whereas

[1] To distinguish between *Surface* N-gram models and *Structural* N-gram models, I will refer to them as N-gram and SN-gram models, respectively.

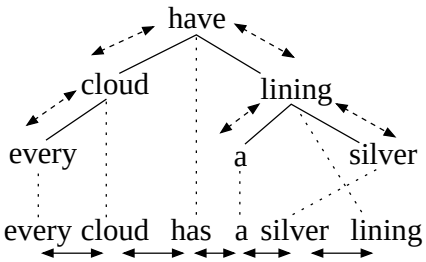

Fig. 1. SN-grams vs N-grams

the N-gram statistical ranking determines both lexical selection and linearization. Therefore, the two models are complimentary in many ways.

Two particular hybrid NLG systems are relevant to the work presented here: Nitrogen/Halogen and FERGUS. Nitrogen is a hybrid NLG system that uses N-grams models to rank through symbolically overgenerated lattices of possible output. A later version of Nitrogen, Halogen, improves on time-space efficiency by compressing the search space into *forests*, compact non-redundant syntactically-derived representations of lattices [13]. Although structural syntactic information is used in constructing forests, the only SLM used in Halogen is a surface N-gram model. FERGUS (Flexible Empiricist/Rationalist Generation Using Syntax) extends the use of N-gram models with a tree-based statistical model, SN-gram model and a lexicalized tree-based syntactic grammar [4]. The use of SN-grams for lexical selection was tested through an artificial expansion of words using WordNet supersynsets [14]. The experiment showed that lexical choice was improved using structural language models.

This paper describes the use of a statistical structural N-gram (SN-gram) model in EXERGE (Expansive Rich Generation for English), the natural language generation (NLG) component of the Spanish-English generation-heavy hybrid machine translation (GHMT) system Matador [15]. The next section is an overview of Matador and EXERGE. Section 3 describes the different uses of SN-grams in EXERGE. Finally, Section 4 presents an empirical evaluation of the contribution of SN-grams in EXERGE.

2 Overview of Matador and EXERGE

Matador is a Spanish-English MT system implemented in the Generation-heavy Hybrid MT (GHMT) approach [16, 15]. The focus of GHMT is addressing resource poverty in MT by exploiting symbolic and statistical target language resources in source-poor/target-rich language pairs. Expected source language resources include a syntactic parser and a word-based translation dictionary. No transfer rules, complex interlingual representations or parallel corpora are used. Rich target language symbolic resources such as word lexical semantics, categorial variations and subcategorization frames are used to overgenerate multiple structural variations from a target-glossed syntactic dependency representation of source language sentences. This symbolic overgeneration is constrained by multiple statistical target language models including N-grams and SN-grams. Some of the advantages of systems developed in this approach include: ease of retargetability to new source languages due to source-target asymmetry, performance stability across

different genres due to lack of need to train on parallel text, and improved grammaticality as compared to systems that do not use deep linguistic resources (for an evaluation comparing Matador to a statistical MT system, see [15]).

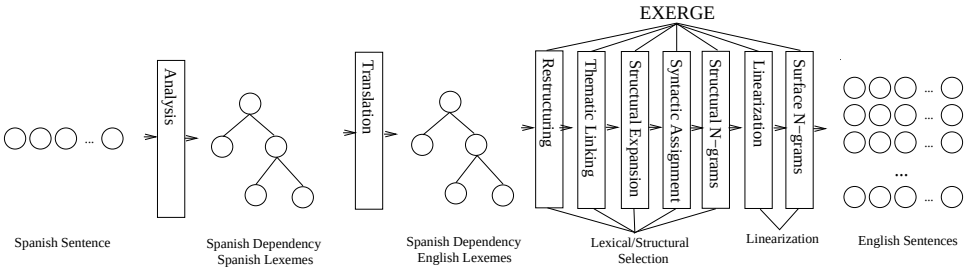

Fig. 2. Matador: Spanish-English Generation-Heavy Hybrid Machine Translation

Figure 2 describes the different components of Matador. There are three phases: Analysis, Translation and Generation. The last phase is marked as EXERGE — EXpansivE Rich Generation for English — a source-language-independent generation module for English. These three phases are very similar to other paradigms of MT: Analysis-Transfer-Generation or Analysis-Interlingua-Generation. However, these phases are not symmetric. The output of Analysis is a deep syntactic dependency that normalizes over syntactic phenomena such as passivization and morphological expressions of tense, number, etc. Translation converts the Spanish lexemes into ambiguous sets of English lexemes. The dependency structure of the Spanish is maintained. The last phase, Generation, is where most of the work is done to manipulate the input lexically and structurally and produce English sequences. The rest of this section discusses EXERGE's resources and major components.

EXERGE utilizes three symbolic and two statistical English resources. The first of the symbolic resources is the word-class lexicon, which defines verbs and prepositions in terms of their subcategorization frames and lexical conceptual primitives. A single verb or preposition can have multiple entries for each of its senses. For example, among other entries, run_1 as in ($John_{agent}$ $ran_{cause-go_{identificational}}$ $store_{theme}$) is distinguished from run_2 as in ($John_{theme}$ $ran_{go_{locational}}$). Second, the categorial-variation lexicon relates words to their categorial variants. For example, $hunger_V$, $hunger_N$ and $hungry_{AJ}$ are clustered together. So are $cross_V$ and $across_P$; and $stab_V$ and $stab_N$. The third symbolic resource is the syntactic-thematic linking map, which relates syntactic relations (such as subject and object) and prepositions to the thematic roles they can assign. For example, while a subject can take on just about any thematic role, an indirect object is typically a *goal*, *source* or *benefactor*. Prepositions can be more specific. For example, *toward* typically marks a *location* or a *goal*, but never a *source*.

EXERGE consists of seven steps (Figure 2). The first five are responsible for lexical and structural selection and the last two are responsible for linearization. Initially, the source language syntactic dependency, now with target lexemes, is normalized and restructured into a syntactico-thematic dependency format. The thematic roles are then determined in the thematic linking step. The syntax-thematic linking is achieved

through the use of thematic grids associated with English (verbal) head nodes together with the syntactic-thematic linking map. This step is a *loose* linking step that does not enforce the subcategorization-frame ordering or preposition specification. This looseness is important for linking from unknown non-English subcategorization frames.

Structural expansion explores conflated and inflated variations of the thematic dependency. Conflation is handled by examining all verb-argument pairs (V_{head},Arg) for *conflatability*. For example, in *John put salt on the butter*, *to put salt on* can be conflated as *to salt* but *to put on butter* cannot be conflated into *to butter*. The thematic relation between the argument and its head together with other lexical semantic features constrain this structural expansion. The fourth step maps the thematic dependency to a target syntactic dependency. Syntactic positions are assigned to thematic roles using the verb class subcategorization frames and argument category specifications. The fifth step prunes ambiguous nodes using a SN-gram model. The purpose of this step is to constrain the overgeneration of the previous steps in preparation for further expansion by the linearization step.

Next is the linearization step, where a rule-based grammar implemented using the linearization engine oxyGen [17] is used to create a word lattice that encodes the different possible realizations of the sentence. Finally, the word lattice is converted into a Halogen-compatible forest to be ranked with Halogen's statistical forest ranker [13].

In terms of input complexity and the balance of symbolic and statistical components, EXERGE is in between the hybrid NLG systems Nitrogen and FERGUS. FERGUS requires the shallowest input (closest to the target-language surface form) and employs the most in statistical and symbolic power. Nitrogen's input is the deepest (semantic representation) and its resources the simplest (an overgenerating grammar and n-gram model).

3 SN-grams in EXERGE

SN-grams are used in EXERGE for (1) lexical selection and (2) structural selection. First, SN-grams are used to prune the ambiguous nodes in the forest of syntactic dependencies produced after the structural expansion and syntactic assignment steps. This pruning is motivated by the need to control the size of the word lattices passed on to the n-gram language model, which tends to be the most expensive step in the whole system. For each tree, T in the forest, a bottom-up dynamic programming algorithm is used to calculate the maximum (joint) frequency of $(word, parent_{word})$ over all $words$ in the $nodes$ of T.[2] Once the scoring is completed, selecting the best unambiguous tree using the dynamic programming tables is straightforward.

As an example, Figure 3 displays the input to EXERGE resulting from the parsing and word-based translation for the Spanish sentence *Este último misil puede equiparse con ojivas nucleares que se están produciendo en Israel.* "This last missile can be equipped with nuclear warheads which are currently produced in Israel." The underlined lexical items in Figure 3 are what the SN-gram model selected for this example. These lexemes are later linearized into a lattice of possible sequences. The top ranked sequence based on a surface bigram model is *This last missile could be equipped with nuclear warheads that are being produced in Israel*.

[2] In a different version of the system, the conditional probability, $P(word|parent_{word})$, is used with no significant effect. This is consistent with findings in the parsing community [18].

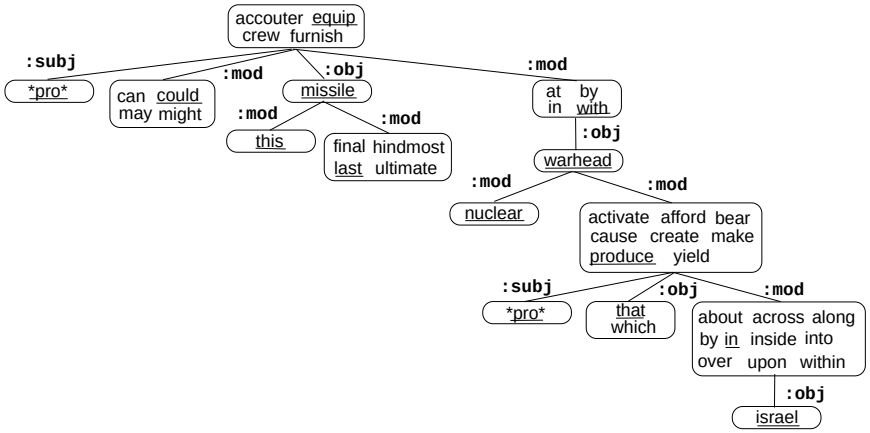

Fig. 3. SN-gram-based Lexical Selection

Secondly, since the symbolic resources used in the structural expansion phase focus on verbs only, a SN-gram-driven approach is used to expand the structure of *noun phrases*. This addresses cases where the direct modification relationship between two English nouns is expressed in Spanish using a preposition.[3] This process is done as follows. For every $parent$-$child$ pair of nominal nodes separated by a $preposition$, the pair is determined to prefer a direct modification relation over $preposition$ if the SN-gram frequency of $(child, parent)$ is higher than the frequency of $(preposition, parent)$ or the frequency of $(child, preposition)$. For example, the preposition in the Spanish *el mundo **en** desarrollo (the world **in** development/developing)* is replaced by a direct modification relationship since the SN-gram (*developing world*) is more common than (*world in*) and (*in development/developing*) in English.[4] Structural variations in noun-noun modification are common in translation between English and other languages (e.g., Japanese [20]).

The use of SN-grams in EXERGE for both lexical and structural choice in a large scale trans-lingual-setting is a major difference from FERGUS's use of SN-grams for lexical choice only in a monolingual setting. Nitrogen doesn't use SN-grams.

4 Evaluation

The contribution of SN-grams is evaluated by comparing translation quality and system efficiency of two versions of Matador, one implementing SN-grams and one without them. The evaluation metric used for translation quality is Bleu (BiLingual Evaluation Understudy) [21]. Bleu is a method of automatic translation evaluation that is quick, in-

[3] The technique presented here for structural selection using SN-grams can be used in reverse to allow translation of direct noun modification in the source language to prepositional modification in English.

[4] A relevant discussion of the translation of noun-noun compounds from Spanish to English is available in [19].

expensive and language independent.[5] The Bleu score is basically an N-gram precision variation calculated as the ratio of the number of N-gram sequences in the generated string that appear in the reference (gold standard) string to the total number of N-gram sequences in the generated string. Bleu is used with 1 to 4-grams and without case sensitivity.[6] System efficiency is measured in terms of CPU time (in seconds). [7]

The blind test set evaluated contained 2,000 Spanish sentences[8] from the UN Spanish-English corpus [24]. The gold standard translations used as references for the Bleu evaluation are the English side of the 2,000 sentences. There was one reference per sentence.

The SN-gram (structural bigram) model was created using 127,000 parsed sentences from the English UN corpus covering over 3 million words. The parsing was done using Connexor's English parser [25]. The resulting noisy treebank was traversed and parent-child instance (lexeme) pairs were counted to create the model. The language model totals 504,039 structural bigrams for 40,879 lexemes. A human checked Treebank was not used to collect SN-gram statistics because none that exist cover the domain of the test set.

The N-gram model was built using 500 thousand sentences from the UN corpus (50,000 from the UN Spanish-English corpus [24] and 450 thousand sentences from the English side of the Arabic-English UN corpus [26]). The Halogen ranking scheme used is bigrams with length normalization. One issue relevant to the N-gram model is the use of bigrams instead of trigrams, which are known to perform better. This decision is purely based on technical issues, namely that Halogen's runtime performance with trigrams is prohibitively long [12].

Table 1. Structural N-gram Evaluation

	Bleu Score	**Overall Runtime**	**Runtime (sec/sentence)**
with Structural N-grams	18.01 +/- 1.00	14,155 sec ≈ 3.9 hours	7.08
without Structural N-grams	17.94 +/- 1.04	34,908 sec ≈ 9.7 hours	17.45

The runtime and resulting Bleu scores are presented in Table 1. A breakdown of the time over the different Matador modules is presented in Figure 4. The Expansion module refers to the first five steps in EXERGE (see Figure 2). The use of SN-grams decreases runtime by 59.45% with no negative effect on text quality. Running the SN-gram pruning doubles the runtime of the expansion module. However the payoff is a 50% decrease in runtime of linearization and ranking, both of which are significantly costlier time-wise than expansion.

[5] Other metrics for evaluating natural language generation include tree-based metrics and combined (string-based and tree-based) metrics [22]. For excellent surveys of machine translation evaluation metrics and techniques, see [23].

[6] Throughout this paper, Bleu scores are presented multiplied by 100.

[7] The system ran on a SparcIII, with 750Mhz and 1GB of memory.

[8] Average sentence length in test set is 15.39 words/sentence.

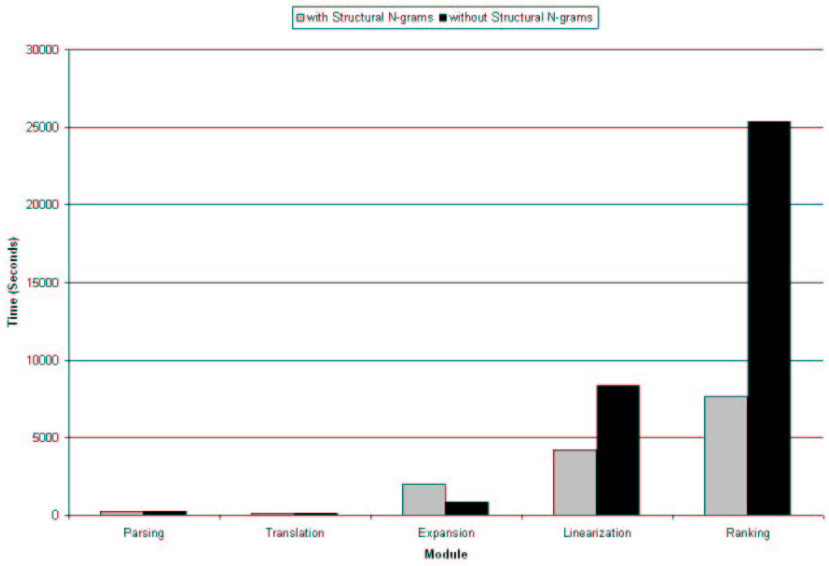

Fig. 4. Overall Runtime per Matador Module: with/without SN-grams

An investigation of the type of errors resulting from the use of SN-grams reveal the following two issues. First, the lack of syntactic knowledge such as part-of-speech information in the current implementation of SN-grams often leads to confusing relationships and erroneous selections, especially in a language like English where Verb/Noun homographs are common. [4] reported a slight increase in text accuracy when POS information was used as part of a structural language model. And secondly, the current implementation of SN-grams is only aware of bigram relations. This, together with the lack of part-of-speech information can lead to erroneous selections such as *they continued **to** their efforts to do X* instead of *they continued **in** their efforts to do X*. The facts that a noisy treebank was used to collect the statistics and that the coverage was limited to 3 million words are possible explanations for some of the errors resulting from using the SN-gram model.

5 Conclusions and Future Work

This paper described the use of a SN-gram model in EXERGE for lexical and structural selection. The use of SN-grams in a Spanish-English GHMT system decreases runtime by 59.45% with no loss in translation quality. The general lesson of this work is that the use of SN-grams as a pruning tool is desirable especially when there is a concern for efficiency. Future directions include (1) improving the quality of the SN-gram model by using more and better data from a clean dependency treebank; (2) including POS information in the SN-gram model; (3) integrating SN-grams models in the structural expansion step for verbs; (4) extending the use of SN-grams in the structural selection of noun phrases to capture more general phenomena than those addressed so far; and

finally, (5) extending the use of SN-grams to the thematic level of representation, where some syntactic variations are normalized, using a noisy thematic treebank.

Acknowledgments

This work has been supported, in part, by Army Research Lab Cooperative Agreement DAAD190320020, NSF CISE Research Infrastructure Award EIA0130422, and Office of Naval Research MURI Contract FCPO.810548265. I would like to thank Bonnie Dorr for her support and advice.

References

1. Knight, K., Hatzivassiloglou, V.: Two-Level, Many-Paths Generation. In: Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL-95), Cambridge, MA (1995) 252–260
2. Brown, R., Frederking, R.: Applying Statistical English Language Modeling to Symbolic Machine Translation. In: Proceedings of the Sixth International Conference on Theoretical and Methodological Issues in Machine Translation, Leuven, Belgium (1995) 221–239
3. Langkilde, I., Knight, K.: Generating Word Lattices from Abstract Meaning Representation. Technical report, Information Science Institute, University of Southern California (1998)
4. Bangalore, S., Rambow, O.: Corpus-Based Lexical Choice in Natural Language Generation. In: Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL 2000), Hongkong, China (2000)
5. Habash, N., Dorr, B., Traum, D.: Hybrid Natural Language Generation from Lexical Conceptual Structures. Machine Translation **17** (2003)
6. Collins, M.: Three Generative, Lexicalised Models for Statistical Parsing. In: Proceedings of the 35th Annual Meeting of the ACL (jointly with the 8th Conference of the EACL), Madrid, Spain (1997)
7. Charniak, E.: Statistical parsing with a context-free grammar and word statistics. In: Proceedings of the AAAI, Providence, RI, AAAI Press/MIT Press (1997) 598–603
8. Charniak, E.: Immediate-head parsing for language models. In: Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics. (2001)
9. Sima'an, K.: Tree-gram parsing: Lexical dependencies and structural relations. In: Proceedings of 38th Annual Meeting of the Association for Computational Linguistics (ACL'00), Hong Kong, China (2000)
10. Ratnaparkhi, A.: Trainable Methods for Surface Natural Language Generation. In: Proceedings of the 1st Annual North American Association of Computational Linguistics, NAACL-2000, Seattle, WA (2000) 194–201
11. Charniak, E.: A maximum-entropy-inspired parser. In: Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics NAACL-2000, Seattle, Washington (2000)
12. Daumé, H., Knight, K., Langkilde-Geary, I., Marku, D., Yamada, K.: The importance of lexicalized syntax models for natural language generation tasks. In: Proceedings of the International Natural Language Generation Conference (INLG-02), new York, New York (2002)
13. Langkilde, I.: Forest-based statistical sentence generation. In: Association for Computational Linguistics conference, North American chapter (NAACL'00). (2000)
14. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press (1998)
15. Habash, N.: Matador: A Large-Scale Spanish-English GHMT System. In: Proceedings of the Ninth Machine Translation Summit (MT SUMMIT IX), New Orleans, USA (2003)

16. Habash, N.: Generation-Heavy Machine Translation. In: Proceedings of the International Natural Language Generation Conference (INLG'02) Student Session, New York (2002)
17. Habash, N.: oxyGen: A Language Independent Linearization Engine. In: Fourth Conference of the Association for Machine Translation in the Americas, AMTA-2000, Cuernavaca, Mexico (2000)
18. Johnson, M.: Joint and Conditional Estimation of Tagging and Parsing Models. In: Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL-2001), Toulouse, France (2001)
19. Aymerich, J.: Generation of Noun-Noun Compounds in the Spanish-English Machine Translation System SPANAM. In: Proceedings of the Eighth Machine Translation Summit (MT SUMMIT VIII), Santiago de Compostela, Spain (2001)
20. Tanaka, T., Baldwin, T.: Translation Selection for Japanese-English Noun-Noun Compounds. In: Proceedings of the Ninth Machine Translation Summit (MT SUMMIT IX), New Orleans, USA (2003)
21. Papineni, K., Roukos, S., Ward, T., Zhu, W.: Bleu: a Method for Automatic Evaluation of Machine Translation. Technical Report RC22176(W0109-022), IBM Research Division, Yorktown Heights, NY (2001)
22. Bangalore, S., Rambow, O., Whittaker, S.: Evaluation Metrics for Generation. In: Proceedings of the 1st International Conference on Natural Language Generation (INLG 2000), Mitzpe Ramon, Israel (2000)
23. Hovy, E.: MT Evaluation Bibliography. In: The ISLE Classification of Machine Translation Evaluations International Standards for Language Engineering (ISLE), Information Sciences Institute, Los Angeles, CA (2000) http://www.isi.edu/natural-language/mteval/2e-MT-bibliography.htm.
24. Graff, D.: UN Parallel Text (Spanish-English), LDC Catalog No.: LDC94T4A (1994) Linguistic Data Consortium, University of Pennsylvania.
25. Tapanainen, P., Jarvinen, T.: A non-projective dependency parser. In: 5th Conference on Applied Natural Language Processing / Association for Computational Linguistics, Washington, D.C. (1997)
26. Jinxi, X.: UN Parallel Text (Arabic-English), LDC Catalog No.: LDC2002E15 (2002) Linguistic Data Consortium, University of Pennsylvania.