F. Mráz and M. Lopatková (Eds.): Theorietag 2012, Prague, 3. – 5.10.2012
Faculty of Mathematics and Physics, Charles University in Prague pp. 1 – 4

As Easy As Vanda, Two, Three: Components for Machine Translation Based on Formal Grammars

Matthias Büchse[(A)]

Chair of Foundations of Programming
Technische Universität Dresden
01062 Dresden, Germany
Matthias.Buechse@tu-dresden.de

Machine Translation is the task of enabling computers to translate text from one language into another. Statistical Machine Translation (SMT), in particular, applies methods from Statistics and Machine Learning to automatically select a translation function that performs well on existing translations, with the hope that it will also perform well on new sentences.

In recent years a lot of research has focused on using formal grammars and related formalisms for specifying translation functions. Among those are synchronous context-free grammars [1, 5], synchronous tree-substitution grammars [10], synchronous tree-adjoining grammars [27, 8], synchronous tree-sequence-substitution grammars [30], extended top-down tree-to-string transducers [16, 14, 12], and multi-bottom-up tree transducers [11, 22].

In principle, these formalisms are amenable to formal treatment, just like weighted string automata and weighted string transducers. The latter possess a rich theory with results about closure properties, characterizations, complexity and decidability. Building on that strong foundation, there is a versatile algorithmic toolbox, as witnessed by [24, 25, 2]. In conjunction, the theory and the toolbox allow for effective algebraic specification and subsequent implementation of tasks in areas such as speech recognition [26] and morphology [15].

However, in the SMT realm, this kind of comprehensive formal treatment has yet to happen. Most of the formalisms named above have been defined ad-hoc, so as to build a translation system which can be evaluated.[1] Core algorithms employed in those systems are often monolithic, and they are implemented in thousands of lines of code, as witnessed by open-source systems such as Moses [19], Joshua [21], or cdec [9]. The source code of most research systems, including Hiero [6], is not available.

Being aware of weighted string automata and weighted string transducers, the community does indeed express a desire for algebraic specification [18]. In fact, it has been shown that the string devices can be employed for central algorithms [17, 7]. Moreover, May and Knight made an effort to develop a toolkit, named Tiburon [23], for extended top-down tree transducers. However, it has not been employed for building research SMT systems.

In the long run, we intend to provide components for (partial) algebraic specification of research SMT systems with our system Vanda, which shall rest on three columns:

[(A)] financially supported by DFG VO 1011/6-1.

[1] This is an enormous feat by itself, one that requires immense engineering skill, and one we do not intend to devalue.

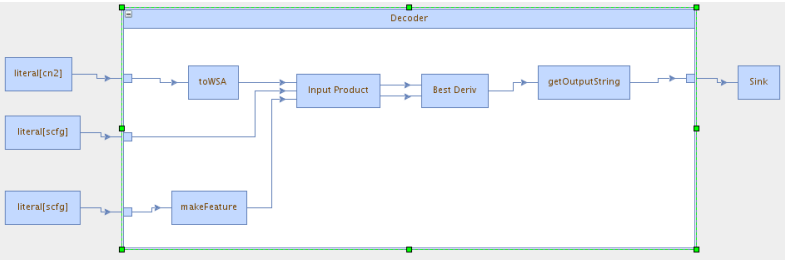

Figure 1: Representation of the decoding task as a workflow in Vanda Studio.

1. A theory and an algorithmic toolbox based on a versatile formal framework, namely interpreted regular tree grammars, or IRTGs [20]. This framework is based on the ideas of bimorphism semantics [3] and initial-algebra semantics [13], and it subsumes all of the formalisms mentioned initially.

2. Vanda Toolbox, a Haskell library that implements the algorithmic toolbox. Haskell is a clean and concise modern high-level language that is compiled into native code via the Glasgow Haskell Compiler. It features a powerful static type system, yet it can still be used for rapid prototyping because of automatic type inference at compile time.

3. Vanda Studio, a graphical (hyper)workflow management system that greatly facilitates conducting experiments by providing both a standardized, well-documented interface and the ability to specify alternatives within a single hyperworkflow. A prototype of Vanda Studio has been implemented [4].

As a proof of our concept, we[2] have implemented in Vanda Toolbox an IRTG-inspired representation of extended top-down tree-to-string transducers, along with suitable algorithms, e.g., for left/right product with regular weighted string/tree languages, binarization of rules, determining n best derivations, rule extraction, and inside-outside EM training. This implementation allows us to accomplish the following three tasks from the area of SMT:

Extraction Rule extraction from any given parallel corpus. To this end, we parse the target (English) side using the Berkeley parser [29], and we use GIZA++ [28] to obtain a word alignment for each sentence pair. We extract all rules that correspond to minimal fragments [12].

Training Estimating rule weights.

Decoding Translating any given sentence (so far without language model).

Each of the tasks can be carried out from within Vanda Studio; as an example, Figure 1 shows the workflow representation of the translation task. Each box in the outer (shaded) region corresponds to a part of a shell script to be run in a Unix environment, while the box labeled “Decoder” represents a Haskell program, and the inner boxes are Haskell functions.

[2]The author had help from his colleagues Toni Dietze, Johannes Osterholzer, and Linda Leuschner.

In my presentation, I will show how the three aforementioned tasks can be accomplished in a component-based manner using Vanda Toolbox and Vanda Studio, and I will report on the performance on medium-scale data.

References

[1] A. V. AHO, J. D. ULLMAN, Syntax directed translations and the pushdown assembler. *J. Comput. System Sci.* **3** (1969), 37–56.

[2] C. ALLAUZEN, M. RILEY, J. SCHALKWYK, W. SKUT, M. MOHRI, OpenFst: a general and efficient weighted finite-state transducer library. In: *Proceedings of the 12th international conference on Implementation and application of automata*. CIAA'07, Springer, 2007, 11–23.

[3] A. ARNOLD, M. DAUCHET, Bi-transduction de forêts. In: *Proc. 3rd Int. Coll. Automata, Languages and Programming*. Edinburgh University Press, 1976, 74–86.

[4] M. BÜCHSE, T. DIETZE, J. OSTERHOLZER, A. FISCHER, L. LEUSCHNER, Vanda: A Statistical Machine Translation Toolkit. In: M. DROSTE, H. VOGLER (eds.), *Proceedings of the Workshop Weighted Automata: Theory and Applications 2012*. 2012, 36–38.

[5] D. CHIANG, Hierarchical Phrase-Based Translation. *Comp. Ling.* **33** (2007) 2, 201–228.

[6] D. CHIANG, A. LOPEZ, N. MADNANI, C. MONZ, P. RESNIK, M. SUBOTIN, The Hiero machine translation system: extensions, evaluation, and analysis. In: *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*. ACL, Morristown, NJ, USA, 2005, 779–786.

[7] A. DE GISPERT, G. IGLESIAS, G. BLACKWOOD, E. R. BANGA, W. BYRNE, Hierarchical Phrase-Based Translation with Weighted Finite-State Transducers and Shallow-n Grammars. *Computational Linguistics* **36** (2010) 3, 505–533.

[8] S. DENEEFE, K. KNIGHT, Synchronous Tree-Adjoining Machine Translation. In: *EMNLP '09: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*. ACL, Morristown, NJ, USA, 2009, 727–736.

[9] C. DYER, A. LOPEZ, J. GANITKEVITCH, J. WEESE, F. TURE, P. BLUNSOM, H. SETIAWAN, V. EIDELMAN, P. RESNIK, cdec: A Decoder, Alignment, and Learning Framework for Finite-State and Context-Free Translation Models. In: *Proceedings of the ACL 2010 System Demonstrations*. ACL, Uppsala, Sweden, 2010, 7–12.

[10] J. EISNER, Learning non-isomorphic tree mappings for machine translation. In: *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 2*. ACL '03, ACL, Stroudsburg, PA, USA, 2003, 205–208.

[11] J. ENGELFRIET, E. LILIN, A. MALETTI, Extended Multi Bottom-up Tree Transducers. In: M. ITO, F. M. TOYAMA (eds.), *Proc. 12th Int. Conf. Developments in Language Theory*. LNCS 5257, Springer, 2008, 289–300.

[12] M. GALLEY, M. HOPKINS, K. KNIGHT, D. MARCU, What's in a translation rule? In: S. DUMAIS, D. MARCU, S. ROUKOS (eds.), *HLT-NAACL 2004: Main Proceedings*. ACL, Boston, Massachusetts, USA, 2004, 273–280.

[13] J. A. GOGUEN, J. W. THATCHER, E. G. WAGNER, J. B. WRIGHT, Initial algebra semantics and continuous algebras. *J. ACM* **24** (1977), 68–95.

[14] J. GRAEHL, K. KNIGHT, J. MAY, Training Tree Transducers. *Comp. Ling.* **34** (2008) 3, 391–427.

[15] T. HANNEFORTH, *fsm2* - A Scripting Language for Creating Weighted Finite-State Morphologies. In: C. MAHLOW, M. PIOTROWSKI (eds.), *SFCM*. Communications in Computer and Information Science 41, Springer, 2009, 48–63.

[16] L. HUANG, K. KNIGHT, A. JOSHI, A syntax-directed translator with extended domain of locality. In: *Proceedings of the Workshop on Computationally Hard Problems and Joint Inference in Speech and Language Processing*. CHSLP '06, ACL, Stroudsburg, PA, USA, 2006, 1–8.

[17] G. IGLESIAS, A. DE GISPERT, E. R. BANGA, W. BYRNE, Hierarchical Phrase-Based Translation with Weighted Finite State Transducers. In: *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. ACL, Boulder, Colorado, 2009, 433–441.

[18] K. KNIGHT, Capturing practical natural language transformations. *Machine Translation* **21** (2007) 2, 121–133.

[19] P. KOEHN, H. HOANG, A. BIRCH, C. CALLISON-BURCH, M. FEDERICO, N. BERTOLDI, B. COWAN, W. SHEN, C. MORAN, R. ZENS, C. DYER, O. BOJAR, A. CONSTANTIN, E. HERBST, Moses: open source toolkit for statistical machine translation. In: *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*. ACL '07, ACL, Stroudsburg, PA, USA, 2007, 177–180.

[20] A. KOLLER, M. KUHLMANN, A Generalized View on Parsing and Translation. In: *Proceedings of the 12th International Conference on Parsing Technologies*. ACL, Dublin, Ireland, 2011, 2–13.

[21] Z. LI, C. CALLISON-BURCH, C. DYER, J. GANITKEVITCH, S. KHUDANPUR, L. SCHWARTZ, W. N. G. THORNTON, J. WEESE, O. F. ZAIDAN, Joshua: an open source toolkit for parsing-based machine translation. In: *Proceedings of the Fourth Workshop on Statistical Machine Translation*. StatMT '09, ACL, Stroudsburg, PA, USA, 2009, 135–139.

[22] A. MALETTI, Why Synchronous Tree Substitution Grammars? In: *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. ACL, Los Angeles, California, 2010, 876–884.

[23] J. MAY, K. KNIGHT, Tiburon: a weighted tree automata toolkit. In: O. IBARRA, H. YEN (eds.), *CIAA 2006*. Lecture Notes in Comput. Sci. 4094, Springer, 2006, 102–113.

[24] M. MOHRI, Weighted automata algorithms. In: M. DROSTE, W. KUICH, H. VOGLER (eds.), *Handbook of Weighted Automata*. chapter 6, Springer, 2009, 213–254.

[25] M. MOHRI, F. C. N. PEREIRA, M. RILEY, The design principles of a weighted finite-state transducer library. *Theoret. Comp. Science* **231** (2000), 17–32.

[26] M. MOHRI, F. C. N. PEREIRA, M. RILEY, Weighted Finite-State Transducers in Speech Recognition. *Computer Speech and Language* **16** (2002) 1, 69–88.

[27] R. NESSON, S. M. SHIEBER, A. RUSH, Induction of Probabilistic Synchronous Tree-Insertion Grammars for Machine Translation. In: *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas (AMTA 2006)*. Boston, Massachusetts, 2006, 128–137.

[28] F. J. OCH, H. NEY, A systematic comparison of various statistical alignment models. *Computational Linguistics* **29** (2003) 1, 19–51.

[29] S. PETROV, L. BARRETT, R. THIBAUX, D. KLEIN, Learning Accurate, Compact, and Interpretable Tree Annotation. In: *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*. ACL, Sydney, Australia, 2006, 433–440.

[30] M. ZHANG, H. JIANG, A. AW, H. LI, C. L. TAN, S. LI, A Tree Sequence Alignment-based Tree-to-Tree Translation Model. In: *Proceedings of ACL-08: HLT*. ACL, Columbus, Ohio, 2008, 559–567.