A Scalable Architecture for Bilingual Lexicography

MS-CIS-97-01

I. Dan Melamed

University of Pennsylvania
School of Engineering and Applied Science
Computer and Information Science Department

Philadelphia, PA 19104-6389

1997

A Scalable Architecture for Bilingual Lexicography

I. Dan Melamed
Dept. of Computer and Information Science
University of Pennsylvania
Philadelphia, PA 19104 USA
melamed@unagi.cis.upenn.edu

1 Introduction

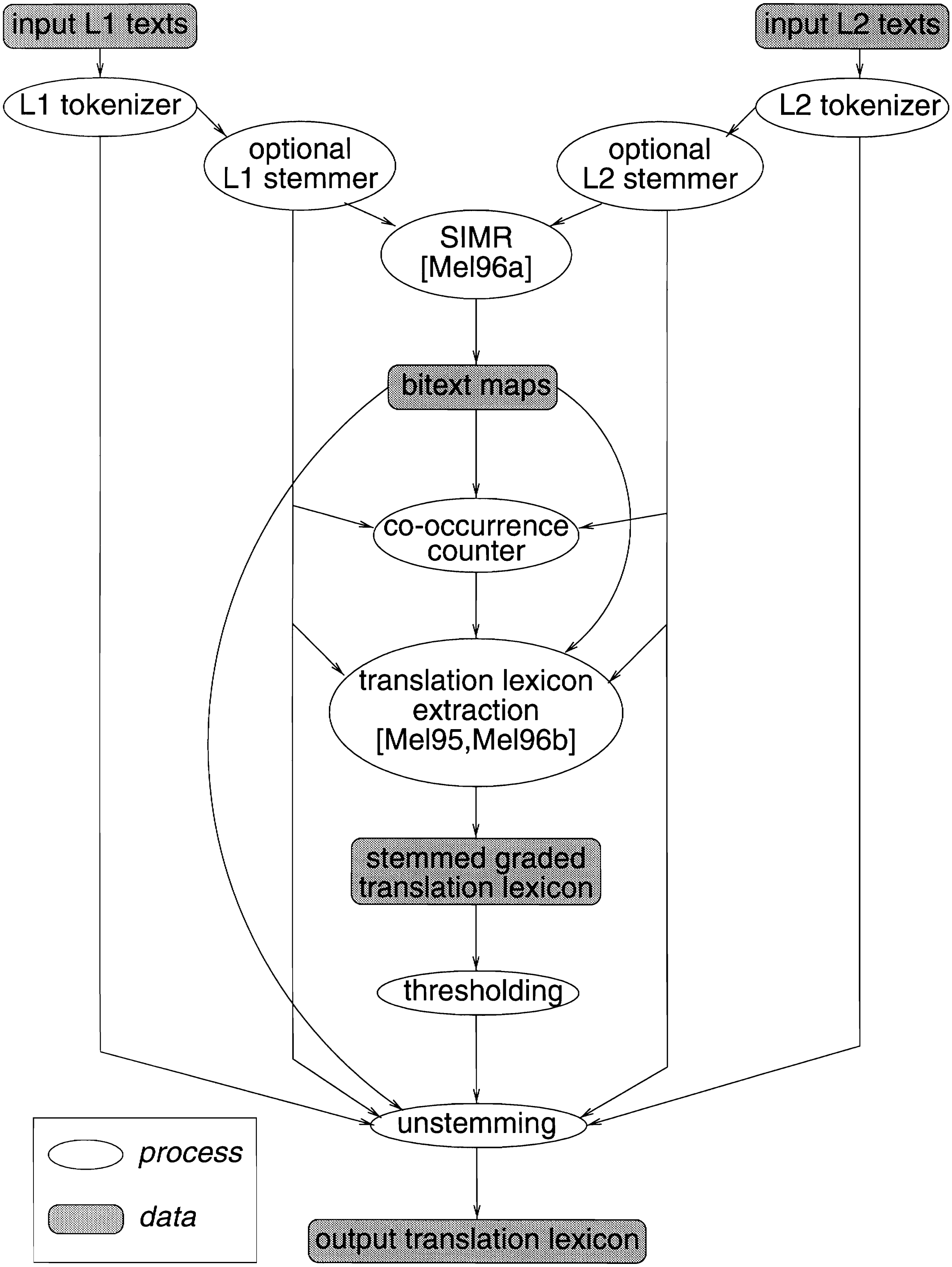
SABLE (Scalable Architecture for Bilingual LExicography) is a turn-key system for producing clean broad-coverage translation lexicons from raw, unaligned parallel texts (bitexts). SABLE is designed to work for any text genre, in any pair of languages. As long as the input texts are mutual translations, the relative word order of the input languages makes no difference. No SABLE component makes any assumptions about the kinds of text units in the input: no component makes any use of sentence boundaries. SABLE was designed with the following features in mind:

- *Black box functionality*: Automatic construction of translation lexicons requires only that the user provide the input bitexts and identify the two languages involved.
- *Robustness*: SABLE copes well with omissions and inversions in translations.
- *Scalability*: SABLE has been used successfully on bitexts larger than 130MB.
- *Portability*: SABLE was initially implemented for French/English, then ported to Spanish/English and to Korean/English [Mel97b]. The porting process has been streamlined and documented [Mel96c].
- *Independence from linguistic resources*: SABLE does not rely on any language-specific resources other than tokenizers and a heuristic for identifying word pairs that are mutual translations, though users can easily reconfigure the system to take advantage of such resources as language-specific stemmers, part-of-speech taggers, and stop lists when they are available.

A data flow diagram for SABLE is on the next page. The following is a brief description of SABLE's main components. See [Mel95, Mel96a, Mel97b, Mel97c] for more details.

2 Tokenization and Stemming

A **tokenizer**'s job is to identify the smallest content-bearing units in text. A **stemmer**'s job is to replace all morphological variants of one lemma with a unique symbol, without assigning that symbol to other lemmas. Not all stemmers are lemmatizers, because a stem may or may not correspond to a word's lemma or root form. Lemmatizers and other stemmers help to alleviate the sparse data problem during lexicon induction, but SABLE can work without them. The current implementation includes a good tokenizer and lemmatizer for English, and fair tokenizers and stemmers for French, Spanish and Korean.



SABLE data flow diagram for languages L1 and L2.

3 Mapping Bitext Correspondence

After both halves of the input bitext(s) have been tokenized, SABLE invokes the *Smooth Injective Map Recognizer (SIMR)* algorithm [Mel96a] and related components to produce a bitext map. A bitext map is an injective partial function between the character positions in the two halves of the bitext. Each point of correspondence (x, y) in the bitext map indicates that the word centered around character position x in the first half of the bitext is a translation of the word centered around character position y in the second half. Since bitext maps can represent crossing correspondences, they are a richer representation of bitext correspondence than "alignments" [Mel96a].

SIMR produces bitext maps a few points of correspondence at a time, by interleaving a point generation phase and a point selection phase. SIMR is equipped with several "plug-in" matching heuristics which are based on cognates [SFI92, Mel95, Mel96a] and/or "seed" translation lexicons [Mel97b]. Correspondence points are generated using a subset of these matching heuristics; the particular subset depends on the language pair and the linguistic resources available for that language pair. SIMR filters candidate points of correspondence using a geometric pattern recognition algorithm. The recognized patterns may contain non-monotonic sequences of points of correspondence, to account for word order differences between languages. The filtering phase can be efficiently interleaved with the point generation phase so that SIMR's expected running time and space are linear in the size of the input bitext.

SIMR's matching heuristics all work at the word level, which is a happy medium between larger text units like sentences and smaller text units like character n-grams. Algorithms that map bitext correspondence at the phrase or sentences level are limited in their applicability to bitexts that have easily recognizable phrase or sentence boundaries, and Church [Chu93] reports that such bitexts are far more rare than one might expect. Moreover, even when these larger text units can be found, their size imposes an upper bound on the resolution of the bitext map. On the other end of the spectrum, character-based bitext mapping algorithms [Chu93] are limited to language pairs where cognates are common; in addition, they may easily be misled by superficial differences in formatting and page layout and must sacrifice precision to be computationally tractable. Word-level matching predicates are also more versatile because words bear semantic content, unlike character n-grams. Therefore, word-level matching predicates can be augmented with semantic filters based on translation lexicons [Mel95], part-of-speech information [Mel95], lists of *faux-amis* [Mac95] and/or semantic entropy thresholds [Mel97a].

4 Translation Lexicon Extraction

Early efforts at extracting translation lexicons from bitexts deemed two tokens to co-occur if they occurred in aligned sentence pairs [G&C91]. Bitext maps admit a more general definition of token co-occurrence. SABLE counts two tokens as co-occurring if their point of correspondence lies within a short distance δ of the interpolated bitext map in the bitext space, as illustrated in Figure 1. To ensure that interpolation is well-defined, minimal sets of non-monotonic points of correspondence are replaced by the lower left and upper right corners of their minimum enclosing rectangles (MERs).

SABLE uses token occurrence and co-occurrence counts to induce an initial translation lexicon, using the method described in [Mel95]. The *iterative filtering* module then alternates between estimating the most likely translations among word tokens in the bitext and estimating the most likely translations between word types. This re-estimation paradigm was pioneered by Brown et al. [BD+93a]. However, their models were not designed to produce deterministic translation lexicons. Though some have tried, it is not clear how to extract translation lexicons from Brown et al.'s

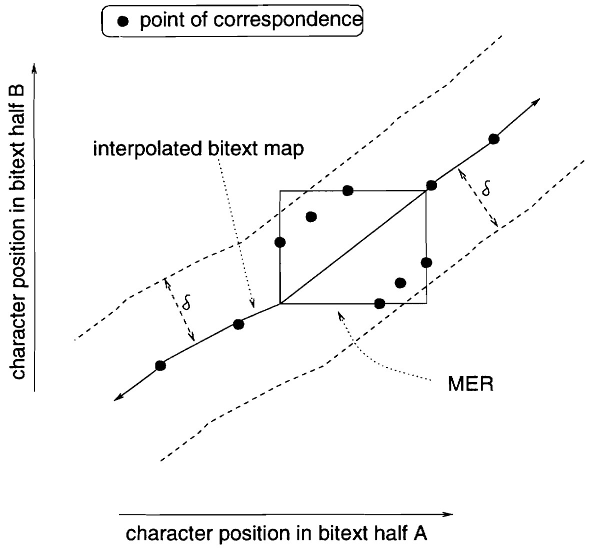

● point of correspondence
character position in bitext half B
interpolated bitext map
δ
δ
MER
character position in bitext half A

Figure 1: *Word token pairs whose points of correspondence lie between the dashed boundaries count as co-occurrences.*

models [W&X95]. In contrast, SABLE automatically constructs an explicit translation lexicon, consisting of word type pairs that are not filtered out during the iterative filtering cycle [Mel96b]. Neither of the translation lexicon extraction modules pay any attention to word order, so they work just as well for language pairs with different word order.

5 Thresholding

As with the output of most lexicon construction methods, SABLE output exhibits a tradeoff between recall and precision. Translation lexicon recall can be automatically computed with respect to the input bitext [Mel96b]. SABLE users have the option of specifying the recall they desire in the output. By default, SABLE will choose a likelihood threshold that is known to produce reasonably high precision.

6 Unstemming

When SABLE is dealing with a language for which a stemmer is available, all processing is done on word stems rather than on the inflected surface forms. Thus, the output of the Thresholding module is a stemmed translation lexicon. Some applications prefer lexicon entries to pair up inflected surface forms. However, the surface form cannot be recovered deterministically just by looking at the lexicon entry, because, in general, all morphological variants of the same word may have the same stem, but only some of the variants may co-occur with a given translation.

To determine which surface forms actually co-occur with which translations, SABLE does an extra sweep through the input bitexts, linking tokens using the competitive linking algorithm [Mel96b, Mel97c]. SABLE records the pair of surface forms that corresponds to each linked pair of word stems. At the end, every stemmed lexicon entry is replaced by every pair of surface forms corresponding to that entry. The result is an unstemmed translation lexicon.

7 Conclusion

SABLE is a practical tool for bitext analysis. It can be used off the shelf for a variety of multilingual applications in computational linguistics and bilingual lexicography.

Acknowledgements

Development of SABLE was supported by Sun Microsystems Laboratories Inc., and by ARPA Contract #N66001-94C-6043. SABLE's English stemmer is based on the morphological analyzer described in [KY+92]. SIMR was initially developed using data provided by the Center for Information Technology Innovation (CITI), Laval, Canada. The linguistic resources for porting SABLE to Korean/English were provided by Young-Suk Lee of MIT's Lincoln Laboratories. Philip Resnik of the University of Maryland wrote the Spanish tokenizer.

References

[BD+93a] P. F. Brown, V. J. Della Pietra, S. A. Della Pietra & R. L. Mercer, "The Mathematics of Statistical Machine Translation: Parameter Estimation," *Computational Linguistics 19*(2), 1993.

[G&C91] W. Gale & K. W. Church, "Identifying Word Correspondences in Parallel Texts," *DARPA SNL Workshop*, 1991.

[Chu93] K. W. Church, "Char_align: A Program for Aligning Parallel Texts at the Character Level," *31st Annual Meeting of the Association for Computational Linguistics*, Columbus, OH, 1993.

[KY+92] D. Karp, Y. Schabes, M. Zaidel, and D. Egedi, "A Freely Available Wide Coverage Morphological Analyzer for English," *Proceedings of the International Conference on Computational Linguistics (COLING '92)*, Nantes, France, August 1992.

[Mac95] E. Macklovitch, "Peut-on verifier automatiquement la coherence terminologique?" *Proceedings of the IV^{es} Journées scientifiques, Lexicommatique et Dictionnairiques,* organized by AUPELF-UREF, Lyon, France, 1995.

[Mel95] I. D. Melamed "Automatic Evaluation and Uniform Filter Cascades for Inducing N-best Translation Lexicons," *Proceedings of the Third Workshop on Very Large Corpora*, Boston, MA, 1995.

[Mel96a] I. D. Melamed, "A Geometric Approach to Mapping Bitext Correspondence," *Conference on Empirical Methods in Natural Language Processing*, Philadelphia, U.S.A, 1996.

[Mel96b] I. D. Melamed, "Automatic Construction of Clean Broad-Coverage Translation Lexicons," to appear in *Conference of the Association for Machine Translation in the Americas*, Montreal, Canada, 1996.

[Mel96c] I. D. Melamed, "Porting SIMR to New Language Pairs," IRCS Technical Report #96-26, 1996.

[Mel97a] I. D. Melamed, "Measuring Semantic Entropy," *Proceedings of the SIGLEX Workshop on Tagging Text with Lexical Semantics*, Washington, DC, 1997.

[Mel97b] I. D. Melamed, "A Portable Algorithm for Mapping Bitext Correspondence," *Proceedings of the 35th Conference of the Association for Computational Linguistics (ACL'97)*, Madrid, Spain, 1997.

[Mel97c] I. D. Melamed, "A Word-to-Word Model of Translational Equivalence," *Proceedings of the 35th Conference of the Association for Computational Linguistics (ACL'97)*, Madrid, Spain, 1997.

[P&M97] P. Resnik & I. D. Melamed, "Semi-Automatic Acquisition of Domain-Specific Translation Lexicons," *Proceedings of the Conference on Applied Natural Language Processing*, Washington, D.C., 1997.

[SFI92] M. Simard, G. F. Foster & P. Isabelle, "Using Cognates to Align Sentences in Bilingual Corpora," in *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation*, Montreal, Canada, 1992.

[W&X95] D. Wu & X. Xia, "Learning an English-Chinese Lexicon from a Parallel Corpus," *First Conference of the Association for Machine Translation in the Americas*, Columbia, MD, 1994.