10 Open Questions in Computational Morphonology

Grzegorz Kondrak
Department of Computing Science
University of Alberta
gkondrak@ualberta.ca

Abstract

The objective of this paper is to initiate discussion within the SIGMORPHON community around several issues that involve computational morphology, phonology, phonetics, orthography, syllabification, transliteration, machine translation, inflection generation, and native language identification.

1 Morphology in Machine Translation

In contrast with English, which is a morphologically simple language, many languages have dozens of different wordforms for any given lemma, some of which are unattested even in large monolingual corpora. In Statistical Machine Translation (SMT), lexical sparsity in such languages is often addressed by performing morphological segmentation, which simplifies the correspondence between the tokens in the source and target language. When translating into English from a morphologically complex language, the segmentation is a form of preprocessing performed before the the translation process. Since the English words are not segmented, the output of the decoder can be directly compared to the reference translation. However, when translating in the opposite direction, the segmentation must be reversed to make the generated text readable. *Desegmentation* is typically performed as a post-processing step that is independent from the decoding process. Unfortunately, the pipeline approach may prevent the desegmenter from recovering from errors made by the decoder, including output morpheme sequences that cannot be combined into valid words.

Salameh et al. (2014) propose to replace the pipeline approach with a solution inspired by finite-state methods. They perform desegmentation directly on the search graph of a phrase-based decoder, which is represented as a *lattice* encoding a large set of possible decoder outputs. The lattice, which can be interpreted as a finite–state acceptor over target strings, is composed with a *desegmenting transducer* which consumes morphemes and outputs desegmented words. The desegmenting transducer, in turn, is constructed from a table that maps morpheme sequences to words. The lattice desegmentation algorithm effectively combines both segmented and desegmented views of the target language, and allows for inclusion of features related to the desegmentation process, as well as an unsegmented language model. The results on English-to-Arabic indicate significant improvements in translation quality. However, the morphology of Arabic is largely concatenative, with relatively simple morpheme-boundary adaptations. In contrast, many European languages are classified as *inflecting*, with affixes that represent several rather than a single morpheme. The question remains whether a morphologically-aware approach can be developed to improve translation into inflecting languages as well.

2 Inflection Generation

An alternative to the morphological segmentation approach is to reduce the diverse forms in the training bitext to lemmas, and, at test time, reconstruct the wordforms in the target language directly from lemmas annotated with morphological features. Note that the wordforms that have not been seen in training pose a problem for language models, and are typically shunned by the current SMT systems.

Although complex morphology leads to a high type-to-token ratio, words tend fo fall into certain inflectional paradigms. Individual inflections are obtained by combining a specific affix with a stem. These combinations are rarely concatenative, often affecting characters at the end or even in the middle of a stem.

Proceedings of the 2014 Joint Meeting of SIGMORPHON and SIGFSM, pages 64–68,
Baltimore, Maryland USA, June 27 2014.

For languages without hand-built morphological analyzers and generators, automated learning of morphological paradigms is the only option. Dreyer and Eisner (2011) propose a Dirichlet process mixture model and loopy belief propagation to learn complete paradigms starting from an initial small set of seed paradigms. An unannotated corpus is utilized to guide the predictions of the model by reducing the likelihood of generating unseen wordforms. Durrett and DeNero (2013) align the lemmas with inflected forms to identify spans that change for the inflections, and learn explicit rules for applying those changes in contexts in which they appear. Their joint model is aware of complete paradigms, and is able to correct errors made on individual inflections.

Nicolai et al. (2014) train a discriminative string transducer on lemma-inflection pairs, and apply a separate re-ranking step to take advantage of the paradigmatic constraints. In spite of its relative simplicity, their string transduction approach outperforms the previous approaches to learning morphological paradigms on several European languages. The question remains whether the string transduction approach is also superior to more complex methods on languages with different morphological systems.

3 From Syntax to Morphology

In some languages, syntactic function of phrases is mainly marked by word position and prepositions, while other languages rely on morphology to a greater degree. Similarly, verbal attributes such as tense, person, and gender, can be either encoded morphologically or lexically. Chahuneau et al. (2013) propose a discriminative model for translating into morphologically rich languages that predicts inflections of target words from source-side annotations that include POS tags, dependency parses, and semantic clusters. In other words, they exploit the syntax of the source language to select the most likely wordforms in the target language,

The open question in this case is whether instead of learning a prediction model separately for each language pair, the morphological features could be mapped directly on the source words. For example, in the phrase *she would have asked*, the actual morphological marking is minimal, but the context disambiguates the person, number, gender, and aspect of the verb. Explicit morphological annotation could not only help machine translation, but also provide a rich source of information in the monolingual context, which would go well beyond POS tagging.

4 Transliteration and Morphology

Transliteration is sometimes defined as "phonetic translation" (Knight and Graehl, 1997). In fact, it is straightforward to train a transliteration model using SMT toolkits by treating individual characters as words, and words as sentences. However, unless substantial modifications are made, the accuracy of such a system will be mediocre. Transliteration needs a dedicated approach in order to fully exploit the source-side context and other constraints.

The way we define tasks in NLP is important, because the definitions (and shared tasks) tend to guide research in a particular direction. New papers are expected to show improvement over previously published results, preferably on already established benchmarks. Redefining a task carries the risk of being interpreted as an attempt to avoid a fair experimental comparison, or as a misdirected effort to investigate irrelevant problems.

The NEWS Shared Task on Machine Transliteration was held four times between 2009 and 2012 (Zhang et al., 2012). With the exception of the 2010 edition that included a transliteration mining task, the shared task was invariably defined in terms of learning transliteration models from the training sets of word pairs. This framework seems to ignore the fact that many of the transliteration target words can be found in monolingual corpora, in a marked contrast with the prevalent SMT practice of avoiding unseen words. Cherry and Suzuki (2009) show that the inclusion of a target lexicon dramatically improves transliteration accuracy. Unfortunately, the paper has largely been ignored by the transliteration community (perhaps because it strays from the standard task formulation), as well as the SMT community (perhaps because it shows only modest gains in terms of BLEU score).

Another drawback of limiting the training data to a list of name pairs is the lack of the context that is required to account for morphological alterations. For example, the title of the Russian Wikipedia page that corresponds to *Presidency of Barack Obama* back-transliterates as *Presidentstvo Baraka Obamy*, where the personal

name appears in the genetive case. Simply including morphological variants in the training data without their context is likely to confuse a transliteration model. How to best combine transliteration with morphology remains an open question.

5 Transliteration and Orthography

Transliteration is more than just phonetic translation. In the idealized model of Knight and Graehl (1997) a human transliterator pronounces a name in the source language, modifies the pronunciation to fit the target language phonology, and writes it down using the orthographic rules of the target script. In reality, however, the source orthography strongly influences the form of the transliteration. For example, the Russian transliteration of the name *Dickens* on Wikipedia back-transliterates as *Dikkens*, although *Dykynz* would be much closer to the original pronunciation. For less well-known names that first appear in English-language news, human transliterators are often in the dark because the correct pronunciation may be difficult to guess from the spelling.

Al-Onaizan and Knight (2002) report that a spelling-based model outperforms a phonetic-based model even when pronunciations are extracted from a pronunciation dictionary. Bhargava and Kondrak (2012) present a re-ranking approach that is able to improve spelling-based models by consulting the supplied pronunciations. It remains an open question how to design a superior joint model that would generate transliterations directly from both spelling and pronunciation.

6 Transliteration and Decipherment

Although transliteration is typically defined as conversion between writing scripts, the proper form strongly depends on the particular target language with its phonological and orthographic constraints. For example, the name of the city that hosted the recent Winter Olympics is represented in various European languages as *Sochi*, *Sotchi*, *Sotschi*, *Sotsji*, *Sotji*, *Sotši*, *Soči*, *Soczi*, *Szocsi*, etc. In order to derive language-specific transliteration models, we would need to collect training data for thousands of possible language pairs.

Ravi and Knight (2009) introduce the task of unsupervised transliteration without parallel resources. They formulate the problem as decipherment, and reconstruct cross-lingual phoneme mapping tables from Japanese words of English origin, achieving approximately 50% character accuracy on U.S. names written in the Katakana script.

Hauer et al. (2014) frame transliteration as a substitution cipher, and apply a mixture of character- and word-level language models to the decipherment of a known language written in an unknown script. The authors treat a short text in Serbian as enciphered Croatian, and attempt to recover the "key", which is the mapping between the characters in the two writing scripts. In reality, Croatian and Serbian are distinct but closely related languages, that are written in different scripts and exhibit differences in both lexicon and grammar. In particular, 30 Serbian Cyrillic characters correspond to 27 letters in Croatian Latin, with three of the characters represented in the other script as digraphs (e.g., *nj*). The decipherment error rate plateaus at about 3% at the ciphertext length of 50 words. In contrast, a pure frequency-based approach fails on this task with a mapping error rate close to 90%. The question remains whether a more flexible approach could be applied successfully to unsupervised transliteration of languages that are less closely related.

7 Phonetic Similarity of Translations

Words that are phonetically similar across different languages tend to be transliterations, or at least share the same origin. For this reason, words on two sides of a bitext are more likely to correspond to each other if they exhibit phonetic similarity (Kondrak, 2005). This is true even for completely unrelated languages because of the prevalence of loanwords, proper names, and technical terms. Orthographic similarity, which reflects phonetic similarity, has been exploited in the past to improve word and sentence alignment in SMT, and other NLP tasks.

Surprisingly, the correlation with phonetic similarity appears to hold for any translations, defined as words that express the same meaning in some context. Kondrak (2013) observes that even after all cognates and loanwords are removed from consideration, the similarity between the words from different languages for the same concept is significantly higher on average than the similarity between the words for different concepts (as measured by the Longest Common Subsequence Ratio). This seems to contradict the Saussurean principle of the arbitrariness of the linguistic sign.

Kondrak (2013) proposes to explain this phe-

nomenon by positing a chain of correlations between the following word characteristics: translatability, frequency, length, and similarity. The key observation is that translations are on average closer in terms of their length than random words. First, pairs of cross-lingual translations exhibit a correlation with respect to the logarithm of their frequencies. Intuitively, translations refer to the same semantic concepts, which tend to be expressed with similar frequency across languages. Second, the connection between word frequency and length is well established (Zipf, 1936). Finally, pairs of words that differ in length are less likely to be considered similar, which is reflected by word similarity measures. In summary, the reason for the greater phonetic similarity of translations lies in the similarity of their frequencies, which is reflected by the similarity of their lengths. This hypothesis remains to be verified on other languages and data sets.

8 L1 Phonology in L2

The task of Native Language Identification (NLI) is to determine the first language (L1) of the writer of a text in another language (L2) (Tetreault et al., 2013). Koppel et al. (2005) report 80% accuracy in classifying a set of English texts into five L1 languages using a multi-class linear SVM with features including function words, POS bigrams, and character n-grams. Tsur and Rappoport (2007) observe that limiting the set of features to the relative frequency of the 200 most frequent character bigrams yields a respectable accuracy of about 65%. They interpret this as evidence that the choice of words in L2 is strongly influenced by the phonology of L1. As the orthography of alphabetic languages is representative of their phonology, character bigrams appear to capture these phonetic preferences.

In order to test the above hypothesis, Nicolai and Kondrak (2014) design an algorithm to identify the most discriminative words and the corresponding character bigrams. They find that the removal of such words results in a substantial drop in the accuracy of the classifier that is based exclusively on character bigrams, and that the majority of the most indicative character bigrams are common among different language sets. They conclude that the effectiveness of a bigram-based classifier in identifying the native language of a writer is primarily driven by the relative frequency of words rather than by the influence of the phonology of L1. Although this provides evidence against the hypothesis of Tsur and Rappoport (2007), the question to what degree the L1 phonology affects L2 writing remains open.

9 English Orthography

The English spelling system is notorious for its irregularity. Kominek and Black (2006) estimate that it is about 3 times more complex than German, and 40 times more complex than Spanish. This is confirmed by lower accuracy of letter-to-phoneme systems on English (Bisani and Ney, 2008). A survey of English spelling (Carney, 1994) devotes 120 pages to describe phoneme-to-letter correspondences, and lists 226 letter-to-phoneme rules, almost all of which admit exceptions.

In view of this, the claim of Chomsky and Halle (1968) that English orthography is "close to optimal" could be interpreted as facetious. The question is how we could validate the accuracy of this statement from the computational perspective. It would seem to require answering at least the following three questions: (a) what is the optimal orthography for English, (b) how to measure the distance between alternative orthographies, and (c) what distance should be considered "close".

10 Syllabification and Morphology

Orthographic syllabification of words is sometimes referred to as hyphenation. Bartlett et al. (2008) propose a sequence prediction approach to syllabify out-of-dictionary words based on letter n-gram features. Despite its high accuracy, their system suffers from the lack of awareness of compound nouns and other morphological phenomena. For example, *hold-o-ver* is incorrectly syllabified as *hol-dov-er*.

Yao and Kondrak (2014) demonstrate that the accuracy of orthographic syllabification can be improved by using morphological information. In particular, incorporating oracle morphological segmentation substantially reduces the syllabification error rate on English and German. If unsupervised segmentation is used instead, the error reduction is smaller but still significant. However, they are unable to achieve any error reduction using a *supervised* segmentation approach, even though it is much more accurate than the unsupervised approach. The confirmation and explanation of this surprising result remains an open question.

References

Yaser Al-Onaizan and Kevin Knight. 2002. Machine transliteration of names in Arabic texts. In *Workshop on Computational Approaches to Semitic Languages*.

Susan Bartlett, Grzegorz Kondrak, and Colin Cherry. 2008. Automatic syllabification with structured SVMs for letter-to-phoneme conversion. In *ACL*, pages 568–576.

Aditya Bhargava and Grzegorz Kondrak. 2012. Leveraging supplemental representations for sequential transduction. In *NAACL-HLT*, pages 396–406.

Maximilian Bisani and Hermann Ney. 2008. Joint-sequence models for grapheme-to-phoneme conversion. *Speech Communication*, 50(5):434–451.

Edward Carney. 1994. *A Survey of English Spelling*. Routledge.

Victor Chahuneau, Eva Schlinger, Noah A. Smith, and Chris Dyer. 2013. Translating into morphologically rich languages with synthetic phrases. In *EMNLP*, pages 1677–1687.

Colin Cherry and Hisami Suzuki. 2009. Discriminative substring decoding for transliteration. In *EMNLP*, pages 1066–1075.

Noam Chomsky and Morris Halle. 1968. *The Sound Pattern of English*. Harper & Row.

Markus Dreyer and Jason Eisner. 2011. Discovering morphological paradigms from plain text using a Dirichlet process mixture model. In *EMNLP*, pages 616–627.

Greg Durrett and John DeNero. 2013. Supervised learning of complete morphological paradigms. In *NAACL-HLT*, pages 1185–1195.

Bradley Hauer, Ryan Hayward, and Grzegorz Kondrak. 2014. Solving substitution ciphers with combined language models. Submitted for publication.

Kevin Knight and Jonathan Graehl. 1997. Machine transliteration. In *ACL*, pages 128–135.

John Kominek and Alan W. Black. 2006. Learning pronunciation dictionaries: Language complexity and word selection strategies. In *HLT-NAACL*, pages 232–239.

Grzegorz Kondrak. 2005. Cognates and word alignment in bitexts. In *MT Summit*, pages 305–312.

Grzegorz Kondrak. 2013. Word similarity, cognation, and translational equivalence. In Lars Borin and Anju Saxena, editors, *Approaches to Measuring Linguistic Differences*, pages 375–386. De Gruyter Mouton.

Moshe Koppel, Jonathan Schler, and Kfir Zigdon. 2005. Determining an author's native language by mining a text for errors. In *SIGKDD*, pages 624–628.

Garrett Nicolai and Grzegorz Kondrak. 2014. Does the phonology of L1 show up in L2 texts? In *ACL*.

Garret Nicolai et al. 2014. In preparation.

Sujith Ravi and Kevin Knight. 2009. Learning phoneme mappings for transliteration without parallel data. In *NAACL*, pages 37–45.

Mohammad Salameh, Colin Cherry, and Grzegorz Kondrak. 2014. Lattice desegmentation for statistical machine translation. In *ACL*.

Joel Tetreault, Daniel Blanchard, and Aoife Cahill. 2013. A Report on the First Native Language Identification Shared Task. In *Workshop on Innovative Use of NLP for Building Educational Applications (BEA8)*.

Oren Tsur and Ari Rappoport. 2007. Using classifier features for studying the effect of native language on the choice of written second language words. In *Workshop on Cognitive Aspects of Computational Language Acquisition*, pages 9–16.

Lei Yao and Grzegorz Kondrak. 2014. In preparation.

Min Zhang, Haizhou Li, A Kumaran, and Ming Liu. 2012. Report of NEWS 2012 machine transliteration shared task. In *4th Named Entity Workshop*, pages 10–20.

George Zipf. 1936. *The Psychobiology of Language*. Routledge.