Corpus development for machine translation between standard and dialectal varieties

Barry Haddow[1], Adolfo Hernández Huerta[2], Friedrich Neubarth[2], Harald Trost[3]

[1)] ILCC, School of Informatics, University of Edinburgh
bhaddow@staffmail.ed.ac.uk
[2)] Austrian Research Institute f. Artificial Intelligence (OFAI)
{adolfo.hernandez, friedrich.neubarth}@ofai.at
[3)] Institute for Artificial Intelligence, Medical University of Vienna
harald.trost@meduniwien.ac.at

Abstract

In this paper we describe the construction of a parallel corpus between the standard and a non-standard language variety, specifically standard Austrian German and Viennese dialect. The resulting parallel corpus is used for statistical machine translation (SMT) from the standard to the non-standard variety. The main challenges to our task are data scarcity and the lack of an authoritative orthography. We started with the generation of a base corpus of manually transcribed and translated data from spoken text encoded in a specifically developed orthography. This data is used to train a first phrase-based SMT. To deal with out-of-vocabulary items we exploit the strong proximity between source and target variety with a backoff strategy that uses character-level models. To arrive at the necessary size for a corpus to be used for SMT, we employ a boot-strapping approach. Integrating additional available sources (comparable corpora, such as Wikipedia) necessitates to identify parallel sentences out of substantially differing parallel documents. As an additional task, the spelling of the texts has to be transformed into the above mentioned orthography of the target variety.

1 Introduction

Statistical machine translation between dialectal varieties and their cognate standard variety is a challenge quite different from translation between major languages with large resources on both sides. Instead of having huge corpora at hand that offer themselves for machine learning techniques, substantial written corpora of dialectal language varieties are rare. In addition, there is no authoritative orthography, which calls for methods to normalize the spelling of existing written texts. Parallel resources for a standard language and a dialectal variety thereof are even less common. But such parallel data is the workhorse of modern machine translation systems and key to producing sufficiently natural utterances. On the positive side, the relative proximity between a standard language and its varieties opens up new possibilities to gather parallel data, despite data sparsity.

In this paper we will outline methods to acquire such data, developed for a specific pair of varieties, Austrian German (AG), the standard variety, and a dialectal variety spoken in the capital, Viennese dialect (VD) (Schikola, 1954), (Hornung, 1998).[1] From a linguistic perspective, it has to be noted that dialects generally are not really homogenous. Lacking standardization initiatives, reinforcement by education or public media and predominantly being confined to oral usage, dialects most often form a dynamic continuum between different varieties and speaker groups. Being defined by social group rather than geographical regions, the Viennese variety is a sociolect in the strict sense, where dialects in urban regions are generally associated with lower social classes (Labov, 2001). Also, speakers with native competence usually adapt the register to the communicative situation as well as to the content of the utterances in a very dynamic way. Switching between varieties and subtle gradual shifts are a very natural phenomenon in such a linguistic situation.

While being aware that the linguistic conception of a dialect is not uncontroversial, we still think that it is feasible and appropriate to model a dialectal variety that conforms to a stereotype of that dialect.

The paper focuses on the generation of the resources necessary for statistical machine translation between a standard variety with rich resources (AG) and a dialectal variety (VD) with almost no resources. The strategy is to create a minimal base corpus comprising bilingual data in a standardized orthography for VD, and in a second step applying a bootstrapping strategy in order to gain a suf-

[1]The work presented in this paper is based on the project 'Machine Learning Techniques for Modeling of Language Varieties' (MLT4MLV - ICT10-049) funded by the Vienna Science and Technology Fund (WWTF).

Proceedings of the Adaptation of Language Resources and Tools for Closely Related Languages and Language Variants, pages 7–14,
Hissar, Bulgaria, 13 September 2013.

ficient amount of bilingual lexical resources and to increase the data on the basis of automatically generated translations. As proximity between the varieties works on our side, we give detailed descriptions of how the linguistic closeness can be exploited to bootstrap the required resources.

2 Background

Pairs of closely related languages (or language varieties) offer themselves to exploit the linguistic proximity in order to overcome the usual scarcity of parallel data. Nakov and Tiedemann (2012) take advantage of the great overlap in vocabulary and the strong syntactic and lexical similarity between Bulgarian and Macedonian. They develop an SMT system for this language pair by employing a combination of character and word level translation models, outperforming a phrase-based word-level baseline. Regarding MT of dialects, Zbib et al. (2012) use crowdsourcing to build Levantine-English and Egyptian-English parallel corpora; while Sawaf (2010) normalizes non-standard, spontaneous and dialect Arabic into Modern Standard Arabic to achieve translations into English.

A considerable amount of work has been done on extracting parallel sentences from comparable corpora, i.e. a set of documents in different languages that contains similar information. Munteanu and Marcu (2005) use a Maximum Entropy classifier trained on parallel sentences to determine if a sentence pair is parallel or not. Based on techniques of Information Retrieval, Abdul-Rauf and Schwenk (2011) use the translations of a SMT system in order to find the corresponding parallel sentences from the target-language side of the comparable corpus. Smith et al. (2010) explore Wikipedia to extract parallel sentences where, once they achieve an alignment at the document level by taking advantage of the structure of this online encyclopedia, they train Conditional Random Fields to tackle the task of sentence alignment. Tillmann and Xu (2009) extract sentence pairs by a model based on the IBM Model-1 (Brown et al., 1993) and perform training on parallel data. With the exception of Munteanu and Marcu (2005), where bootstrapping techniques find application, these methods require (and presuppose the existence of) a certain amount of resources (i.e. parallel data or lexicon coverage) not available for some languages or varieties.

3 Constructing a Parallel Corpus

For Standard German to Viennese dialect, there were no existing parallel data sets and, moreover, most monolingual text sources that exist are written in an inconsistent way, oscillating between standard conventions and free attempts to encode the phonetic realization in the dialect. The first step was to design an orthographic standard for the target language that would be consistent, unambiguous and phonologically transparent. In the light of applicability in language technology, accuracy towards phonological properties seemed the most important criterion, on a par with the necessity to minimize lexical ambiguities. This is different from producing literary texts, where readability might be a more prominent issue, and the orientation towards the standard orthography may have a higher priority.

A second problem with initial data acquisition is the fact that dialect speakers in Vienna very often switch between the dialect and the standard variety, depending on the communicative situation, but also on the content that may invite to use a higher register. Text data with a bias towards the standard by virtue of standard orthography quite often also reflects such switching processes. In order to circumvent such biases, we carefully selected colloquial data of VD that are as authentic to the dialect as possible. The basic material consists of transcripts of TV documentaries and free interview recordings of dialect speakers. The transcripts were manually translated into both AG and VD, the latter being vacuous in most cases. This way we could ensure that (rarely occurring) switchings into the standard would not end up in the target model. A typical example looks as follows, where AG and VD refer to the standard and the Viennese orthography of a sentence from our corpus.

(1) AG: `Ja, ich weiß es doch.`
VD: `Jå, i waas s e.`
'yes, I know it anyway.'

In an early stage, we were interested in finding a way to align these parallel sentences on a word-by-word basis, in order to simultaneously generate lexical resources comprising morphology and morpho-syntactic features (PoS tags, grammatical features, such as gender, case, person, number etc.). Given that usually the two translations are syntactically very similar, with little re-

ordering and/or n-to-n correspondences, and also that many corresponding words are 'cognates', meaning that they are lexically (and morphologically) the same in both varieties, with different phonology and spelling (e.g., AG *'weiß'* corresponds to VD *'waas'* "(I) know"), we bootstrapped a word-alignment routine that very soon provided promising results.

The core idea was to use the string edit distance (Levenshtein algorithm) to determine whether two words should be aligned or not. Because it matters if one or more editing steps (errors) occur in a short or in a long word, we normalized the string edit distance by a factor consisting of the logarithm of the average string length or a special penalty factor for very short strings). However, the orthographic forms may differ substantially while referring to identical words. So, the second ingredient was to train a character based translation model between AG and VD, using the data-driven grapheme-to-phoneme converter Sequitur G2P (Bisani and Ney, 2008). These automatically generated strings of dialect words (VD*) are then compared to the words of the target (VD). Given that the initial data is very limited, the results of the G2P translation are not reliable as a translation, but still very useful to determine the distance measure. Since the full set of extracted word pairs (after validation) is used to re-train the models in an iterative way, the word alignment gets better the more data is added. In a way, over-fitting, generally carefully avoided in statistical modeling, works to our advantage.

The alignment algorithm in a first step linearly searches for the best path of matches. If the score provided by the string edit distance is above a given threshold, insertions and deletions are the less costly options, and the words will not be aligned. By this method, we would only align cognates and miss the more interesting cases where words of AG are translated into different words that may be typical for the dialect (e.g., VD has a special word for AG *'Polizist'* "policeman": VD *'kibara'*). Therefore two more iterations over the set of aligned pairs try to find these non-cognate pairs. First, adjacent insertions and deletions are aligned regardless of the distance measure. This guarantees that word pairs that are not cognates (with a high degree of similarity), but different lexical items, are also captured by the word alignment, given that the syntactic structure of the source and the target sentence are approximately the same. Second, non-adjacent insertion-deletion pairs with a distance measure below the threshold are marked as valid alignments. That way the algorithm that by itself provides only linear alignments is also capable to capture some non-local alignments resulting from syntactic re-ordering.

With regard to SMT and contemplating the immanent problem of data sparsity, it seems obvious that a factorized translation model (Koehn and Hoang, 2007) will have certain advantages over a translation model that only considers full word forms. This, however, requires the generation of lexical resources for both language varieties. For the source language (AG) such resources already exist. The question is, if and how the lexical information stemming from the source language can be transferred onto the target language.

Our word alignment is capable of identifying cognates. However, these cognates will only cover certain word forms out of more complex morphological paradigms. Given that for AG, the lemma and the information about the paradigm can be automatically retrieved from the word form, the task is to identify lemma and the paradigm from the VD word form. In many cases it will suffice to strip off the inflectional endings and to transfer the morphological information from the AG entry. However, there are many deviations (from AG to VD) as well as exceptions, also only real cognates can be treated that way, so there has to be done some manual validation in order to create a VD lexicon that in the end covers all word forms.

(2)

INPUT:	`haus NN Neut . -I-a`	
OUTPUT:	haus	haus+NN+Neut+Sg+NDA
	heisa	haus+NN+Neut+Pl+NDA

sg./pl. forms of VD 'haus' (AG 'Haus' *'house'*)

When the lemma, the major category and the relevant morphological information are identified, this is sufficient to generate all word forms together with morphological features in a given language variety.

4 Machine Translation Experiments

In this section we report on some experiments using the data set described in the previous section to build statistical machine translation systems, using Moses (Koehn et al., 2007).

4.1 Corpus

The corpus was split into four sections, TRAIN, DEV, DEVTEST and TEST, where the first was used

for estimation of phrase tables and language models, the second for tuning the MT system parameters and the third for testing during system development. The last was reserved for final testing. The relative sizes of the three section is shown in Table 1.

Section	Sentences	Tokens	
		AG	VD
TRAIN	4909	39108	40031
DEV	600	4775	4882
DEVTEST	600	4712	4803
TEST	600	4841	4943

Table 1: Corpus sizes (untokenised)

4.2 Word-level Models

The word-level models are standard phrase-based models built using Moses. The parallel text is tokenised using the Moses tokeniser for German, then it is all lowercased. This parallel text is then aligned in both directions using GIZA++ (Och and Ney, 2000) and the alignments are symmetrised using the "grow-diag-final-and" heuristic. The aligned parallel text is then used to estimate a translation table using the standard Moses heuristics, and a 3-gram language model built on the target side of the parallel text using SRILM with Kneser-Ney smoothing. The translation and language models are then combined with a distance-based reordering model and their weights optimised for BLEU using MERT on the DEV corpus.

4.3 Character-level Models

In earlier work on MT for closely-related languages (Vilar et al., 2007; Tiedemann, 2009; Nakov and Tiedemann, 2012), it has been shown that character-level translation models can be effective. These character-level models are also built using phrase-based Moses, but allowing it to treat single characters or groups of characters as "tokens". In the unigram character-level model, we treat each character as a separate token by inserting a space between each of them, and using a special character (||) to indicate word boundaries. For the bigram character-level model, the "tokens" are pairs of adjacent characters, with the same word boundary character as in the unigram model. Table 1 shows examples of a German sentence converted into suitable formats for the character-level unigram and bigram models.

After decoding with one of the character-level models, converting back to word-level text is straightforward in the unigram case; it is just a matter of removing spaces then replacing the special word-boundary character with a space. For the bigram-level model, we remove the first character in each bigram then proceed as for the unigram-level models.

Other than the word-to-character conversion of all data, the character-based models are trained using the standard Moses training pipeline. We use the default maximum phrase-length of 7, and a 7-gram language model, parameters that were observed to work best in early experiments. During tuning, we maximise word-level BLEU with respect to the reference.

4.4 Backoff Models

After observing the performance of word and character-level models, we decided to try to combine them into a *backoff* model, which would use the word-level translation wherever possible, but apply the character-level model for unknown words. In (Nakov and Tiedemann, 2012), they found that a similar model combination gave the best results when translating between closely related languages.

Firstly, we experimented with different variations of the character-level model for the unknown words (OOVs). Each of these models is trained and tuned on the TRAIN and DEV sets, and we report accuracies on the OOVs in DEVTEST (OOV according to the phrase-table built on DEV). The translations of the OOVs were extracted from the word alignments of the base corpus, and out of 330 OOVs, 325 have gold translations.

The first two character-level models are just the unigram and bigram baseline models from Section 4.3. We then built further models by attempting to extract the *cognates* from the training set. The idea here is that the character-level models are built from "noisy" training data, containing many German-Viennese word-pairs which either represent lexical differences, or are the result of bad alignments. In order to extract the cognates we ran GIZA++ alignment on the combined TRAIN and DEV corpora, extracted all source-target token pairs that were aligned, converted the pairs to the BARSUBST representation (see section 5.1), and filtered using the log-normalised Levenshtein distance.

word-level:	und für die tipps
character-level (unigram):	|| u n d || f ü r || d i e || t i p p s ||
character-level (bigram):	||u un nd d|| ||f fü ür r|| ||d di ie e|| ||t ti ip pp ps s||

Figure 1: Conversion of a German sentence into forms suitable for training the character-level models

Model	Correct	Accuracy (%)
Pass-through	21	6.5
Unigram	154	47.4
Bigram	150	46.2
Unigram cognate	154	47.4
Bigram cognate	150	46.2
Unigram cognate (freq)	160	49.2
Bigram cognate (freq)	145	44.6

Table 2: Comparison of accuracy of character-level models on the OOVs in DEVTEST. The plain unigram/bigram models are trained on complete sentences, whereas the cognate models are trained on cognate pairs (unique or frequency weighted) extracted from these sentences.

With this list of cognate pairs, we trained both unigram and bigram models, firstly from a list of the unique cognate pairs and secondly from the same list with frequencies adjusted to match their corpus frequencies. These models were trained using the usual Moses pipeline, estimating phrase tables and language models from 90% of the cognate pairs and tuning on the other 10%.

The OOV accuracies (on DEVTEST) of all 6 character-level models, as well as a pass-through baseline are shown in Table 2. We can see that, in general, the cognate models offer small improvements on the models trained on the whole sentences, and the unigram models are slightly better than the bigram models.

Finally, we show a comparison of the word-level and character-level systems, with the backoff system (using the unigram cognate frequency adjusted model) in Table 3. The backoff systems are implemented by first examining the tuning and test data for OOVs, then translating these using the character-level model, and creating a second phrase-table with the character-level model. This second phrase table is used in Moses as a backoff table.

For both test sets, the character-level translation outperforms the word-level translation, but the backoff offers the best performance of all. The BLEU scores are relatively high compared to the typical values reported in the MT literature, reflecting the restricted vocabulary of the data set.

Model	DEVTEST	TEST
Word-level	63.28	60.04
Character-level (unigram)	65.00	63.17
Character-level (bigram)	64.98	63.43
Backoff to char-level	68.30	66.13

Table 3: BLEU scores for all translation systems

5 Comparable Corpora

Wikipedia is a multilingual free online encyclopedia with currently 285 language versions. Adafre and de Rijke (2006) investigated the potential of this resource to generate parallel corpora by applying different methods for identifying similar texts across multiple languages. We explore this resource as it contains a relatively large bilingual corpus of articles in (Standard) German (DE) and Bavarian dialects (BAR). There are 5135 parallel articles (status from July 2012), of which 219 are explicitly tagged as "Viennese dialect". It can be assumed that the parallel articles refer to the same content, but texts often differ substantially in style and detail. Articles in Bavarian are generally shorter, containing less information than the corresponding German ones, with an average ratio of about 1:6. The challenge of finding corresponding sentence pairs is met by a sentence alignment method that crucially exploits the phonetic similarity between the German standard and Bavarian dialects, specifically Viennese.

5.1 Sentence Extraction

Our sentence alignment algorithm is primarily based on string-edit distance measures. There exist several open-source alignment tools for extracting parallel sentences from bilingual corpora. However, none of them is applicable to our data because they either require a substantial amount of data to reliably estimate statistical models, i.e. at least 10k sentence pairs, such as the Microsoft Bilingual Aligner (Moore, 2002). But also the number of sentences to be aligned must be almost

equal – with a ratio of 1:6 it was not possible to achieve any reliable results at all. Additionally, the sentences in the parallel texts are presupposed to occur in the same order, which does not apply to the Wikipedia articles under consideration. Similar requirements hold for the Hunalign tool (Varga et al., 2005). Finally, LEXACC (Stefanescu et al., 2012) is a parallel sentence extractor for comparable corpora based on Information Retrieval, but again, certain resources are required beforehand, such as a GIZA++ dictionary created from existing parallel documents. The main obstacle to using any of these algorithms is that the texts in the BAR Wikipedia obey widely differing and mostly ad-hoc orthographic conventions, which are not consistent for a given dialectal variety, even within a single article. In our situation, we had to develop an alignment method that relies only on the linguistic proximity between the two varieties.

Comparing strings of DE that occur in standard orthography with strings of BAR in varying non-standard orthography directly does not make sense, unless both forms are transformed into a phonetically based common form. Inspired by Soundex and the Kölner Phonetik algorithm (Postel, 1969), we developed an algorithm (henceforth BARSUBST) that takes into account some characteristics of the Bavarian dialect family (liquid vocalization: i.e., DE *'viel'* corresponds to VD: *'fü'*; vowels are retained as one class of characters; the character for 'dark a' <å> had to be included). This ensures that cognate words will have a very low string edit distance. Just to give an impression, we calculated the average values of Levenshtein string edit distance and the average ambiguity of particular word forms of AG and VD from the data of the word aligned base corpus. As ambiguity we counted the number of occurrences of a given word in a distinct word pair. The baseline value of 1.26/1.27 relates to the fact that for a given word there may be more than one valid translations. When the ambiguity is much higher this indicates that the distance measure is less reliable (words that should not relate turn out to be identical).

The average LD significantly drops down from 3.47 of the baseline (lowercase word forms) to approx. 1.0 for both, Kölner Phonetik and BARSUBST. The average ambiguity is almost equal for both BAR and DE - slightly below a value of 2 with BARSUBST; the Kölner Phonetik algorithm fares better with DE word forms, but worse with BAR word forms, which shows that it is justified to adapt the Kölner Phonetik algorithm to our purposes.

	LD	amb.DE	amb.VD
Baseline (lcase)	3.47	1.26	1.27
Soundex	1.35	4.53	5.69
Kölner Phonetik	1.00	1.87	2.25
BARSUBST	0.99	1.94	1.96

Table 4: Average distance and ambiguity values

Applying this algorithm to words of both DE and BAR, we defined a scoring function that evaluates possible word alignments against each other in order to find the optimal sentence pairs from related articles. The alignment algorithm works as follows: after creating a matrix of all sentence pairs, each potential alignment is evaluated by the scoring function that takes into account the sum of (positive and negative) scores resulting from a non-linear word alignment based on the transformed character sequences (best matches aligned first), the number of not-aligned words (negative scores) and a penalty for crossing alignments and extra short word sequences. We selected a set of approx. 50 sentences to manually test the effects of the different parameters of the scoring function. After fine-tuning the parameters, a threshold of above zero proved to be a good indicator for a correct alignment between two sentences. From this matrix, sentence pairs are extracted in the order of their scores (best scores aligned first) until a defined threshold is reached.

We used only articles that are explicitly tagged as 'Viennese' (approx. 200). From these we extracted and aligned 4414 sentences with 40.1k word tokens that correspond to 12.9k word types. Unlike the texts extracted from spontaneous speech recordings, the Wikipedia texts seem to contain many more word types, which is due to the fact that Wikipedia texts tend to contain a large number of named entities. Unfortunately, these are not very useful for SMT by themselves, but still the amount of parallel data can be significantly increased.

5.2 Orthography Normalization

One problem still to be solved in a satisfactory way is how to deal with non-standardized, inconsistent orthography in dialectal texts. The cor-

pus of parallel sentences from Wikipedia articles can in principle provide ample training data for a character-level translation algorithm between non-standard orthography of BAR and our specifically designed, standardized orthography for VD. Given a 1-to-1 word alignment based on the Levenshtein distance of BARSUBST transformed word forms of sufficient quality, we can extract a list of BAR-DE word pairs, but the target, words in VD orthography, is missing.

To tackle this problem, we used the data from our speech-based corpus of aligned AG-VD word pairs. (We take Austrian German (AG) and German Standard (DE) to refer to the same variety). We filtered the list of word pairs gained from BAR-DE word alignment for only those DE-BAR pairs where we have an AG-VD word pair in our base corpus. That way, the AG/DE standard is used as an anchor to link non-standardized BAR orthography to our standard of DE orthography.

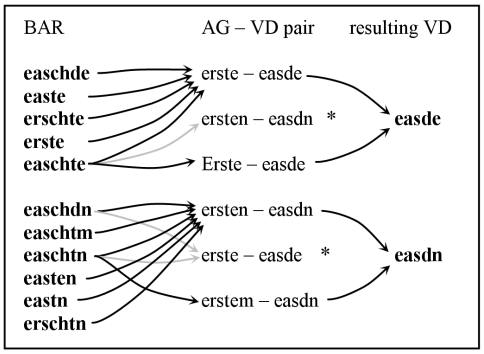

Figure 2: Correspondences between BAR orthographic forms, AG-VD pairs and VD forms.

As can be seen in Figure 2, the correspondences can be manifold. In order to decide which VD form is the correct one to be associated with certain BAR variants, we apply a weighted Levenshtein distance measure, where the weights are chosen in such a way that plausible and frequent substitutions are assigned less costs than others. When more data is available, these weights can be re-estimated on a statistical basis, for a start we just stipulated them based on the linguistic knowledge about the two varieties. The matches are not symmetrical under this approach, for example BAR <m> matching with VD <n> (which often occurs when dative endings in BAR are written according to the standard of DE, while they are pronounced and written as /n/ in VD) is assigned a cost of 0.6, while the reverse match is not defined and receives the default cost value of 1.)

Having gathered some initial training data this way, we experimented to train a character level translation using again Sequitur G2P. Of all the pairs, we spared 25% for testing and used the rest for training, which proved to be very little – approx. 1500 instances of BAR-VD pairs. To increase this number in a sensible way, we created two more sets by adding the set of AG-VD pairs from the base corpus and adding the set of VD-VD pairs, simulating a situation where the BAR input is already in the correct orthography. The results are not fully compelling (50 % correct spellings in the optimal case). This may be due to the rather small amount of training data, but also to the high degree of variance in the input data. To enhance the quality of orthography normalization we foresee a combination of modelling character-level BAR-VD correspondences with the character-level translation models of AG to VD that hopefully will make it possible to achieve a automatically normalized parallel corpus from the Wikipedia data that conforms to the same standards as the base corpus.

6 Discussion and Outlook

Starting from a base corpus of parallel AG and VD sentences generated by manual transcription of spoken text and translation into the two varieties, we applied various methods to iteratively enhance the word alignment and the generation of lexical resources in the target variety. Using this corpus for SMT provided good preliminary results given that we employed a backoff strategy for OOV words building on character level models. To enlarge the corpus with automatic methods, we extracted sentence pairs from corresponding articles from the Bavarian and the German Wikipedia, where the identification of corresponding sentences was based on the similarity of the two varieties. Still, the normalization of Bavarian/Viennese dialectal spelling to our orthography is work in progress. However, methods for normalization of spelling are crucial for the acquisition of monolingual data from texts in dialects, generally. Another line will be the bootstrapping of parallel data by generating automatic translations of sentences that are selected by an active learning algorithm, in order to gain maximal information for the system.

References

Sadaf Abdul-Rauf and Holger Schwenk. 2011. Parallel sentence generation from comparable corpora for improved SMT. *Machine Transl.*, 25(4):341–375.

Sisay Fissaha Adafre and Maarten de Rijke. 2006. Finding similar sentences across multiple languages in wikipedia. In *Proceedings of the 11th Conference of the European Association for Computational Linguistics (EACL)*, pages 62–69.

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

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Comp. Ling.*, 19(2):263–311.

Maria Hornung. 1998. *Wörterbuch der Wiener Mundart*. ÖBV, Pädagogischer Verlag.

Philipp Koehn and Hieu Hoang. 2007. Factored translation models. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 868–876, Prague, Czech Republic, June.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL), demonstration session*, pages 177–180, Prague, Czech Republic.

William Labov. 2001. *Principles of linguistic change (ii): social factors*. Blackwell, Massachusetts.

Robert C. Moore. 2002. Fast and accurate sentence alignment of bilingual corpora. In *Proceedings of the 5th Conference of the Association for Machine Translation in the Americas (AMTA)*, pages 135–144, Tiburon, CA.

Dragos Stefan Munteanu and Daniel Marcu. 2005. Improving machine translation performance by exploiting non-parallel corpora. *Computational Linguistics*, 31(4):477–504.

Preslav Nakov and Jörg Tiedemann. 2012. Combining word-level and character-level models for machine translation between closely-related languages. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 301–305, Jeju, Korea.

Franz Josef Och and Hermann Ney. 2000. Improved Statistical Alignment Models. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL00)*, pages 440–447, Hongkong, China.

Hans Joachim Postel. 1969. Die Kölner Phonetik. Ein Verfahren zur Identifizierung von Personennamen auf der Grundlage der Gestaltanalyse. *IBM-Nachrichten*, 19:925–931.

Hassan Sawaf. 2010. Arabic dialect handling in hybrid machine translation. In *Proceedings of the 9th Conference of the Association for Machine Translation in the Americas (AMTA)*, Denver, Colorado.

Hans Schikola. 1954. *Schriftdeutsch und Wienerisch*. Österr. Bundesverlag für Unterricht, Wissenschaft und Kunst.

Jason R. Smith, Chris Quirk, and Kristina Toutanova. 2010. Extracting parallel sentences from comparable corpora using document level alignment. In *Proceedings of Human Language Technologies: The 2010 Annual Conference of the North American Chapter*, pages 403–411, Los Angeles, California.

Dan Stefanescu, Radu Ion, and Sabine Hunsicker. 2012. Hybrid parallel sentence mining from comparable corpora. In *Proceedings of the 16th Annual Conference of the European Association for Machine Translation (EMAT)*, pages 137–144, Trento, Italy.

Jörg Tiedemann. 2009. Character-based {PSMT} for closely related languages. In Lluís Marqués and Harold Somers, editors, *Proceedings of 13th Annual Conference of the European Association for Machine Translation (EAMT'09)*, pages 12–19, Barcelona, Spain.

Christoph Tillmann and Jian-Ming Xu. 2009. A simple sentence-level extraction algorithm for comparable data. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 93–96, Boulder, Colorado.

Dániel Varga, Péter Halácsy, András Kornai, Viktor Nagy, László Németh, and Viktor Trón. 2005. Parallel corpora for medium density languages. In *Proceedings of the RANLP*, pages 590–596.

David Vilar, Jan-Thorsten Peter, and Hermann Ney. 2007. Can we translate letters? In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 33–39, Prague, Czech Republic. Association for Computational Linguistics.

Rabih Zbib, Erika Malchiodi, Jacob Devlin, David Stallard, Spyros Matsoukas, Richard Schwartz, John Makhoul, Omar F. Zaidan, and Chris Callison-Burch. 2012. Machine translation of Arabic dialects. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT)*, pages 49–59, Montreal, Canada.