Standard Language Variety Conversion for Content Localisation Via SMT

Federico Fancellu, Andy Way
Centre for Global Intelligent Content
School of Computing
Dublin City University
Dublin, Ireland.
ffancellu@cngl.ie,away@computing.dcu.ie

Morgan O'Brien
McAfee Ireland Inc
Citygate
Mahon
Cork, Ireland.
Morgan_O'Brien@mcafee.com

Abstract

Translation between varieties of the same language is a widespread reality in the localisation industry. However, monolingual statistical machine translation (SMT) is still a solution that has not yet been adequately explored; to the best of our knowledge, previous work in this area has never directly applied SMT to varieties of the same language for the precise purpose of reducing the time and cost of human translation and editing of content that needs to be localised.

In this paper, we start exploring the problem by deploying SMT to translate Brazilian Portuguese into European Portuguese. Our exploration mainly takes into consideration the use of bilingual dictionaries to guide the decoder and modify the translation output. We also consider the option of mining a bilingual dictionary from word alignments obtained after standard SMT training.

On good-quality data provided by Intel, we show that the SMT baseline already constitutes a strong system which in a number of experiments we fail to improve upon. We conjecture that bilingual dictionaries mined from client data would help if more heterogeneous training data were to be added.

1 Introduction

Localising content does not only involve translating across different languages, but often also translating between varieties of the same language. These varieties might differ in different repects, including spelling (e.g. British English *colour* vs. American English *colour*), lexicon (e.g. British English *autumn* vs. American English *fall*), word usage (e.g. British English *I'm pissed off* vs. American English *I'm pissed*), grammar (e.g. Irish English *You're after spilling my pint* vs. British English *You've just spilt my pint*), etc. Considering that often such translation tasks are carried out by humans, monolingual translation becomes costly and time-consuming, especially when one takes into account how much the two languages have in common.

Deploying Statistical Machine Translation (SMT, e.g. Koehn et al. (2003)) would appear to offer a solution to the problem. Given that the two varieties are essentially the same language except for some minor differences, we expect most of the translation variants to be captured by an SMT system. Moreover, we rely on the SMT system to be able to capture those structures that are not only acceptable in a language variety but are also *preferable*; in a rule-based system (RBMT), these could only be handled by complex hand-written rules.

The present study was run as a short-term (3∼4-week) innovation project between CNGL and Intel. The main goal of the present paper is to assess to what extent automatic methods can deliver a good translation between language varieties for localised content. For this reason, in the present study we refrain as much as possible from using any hand-crafted rules. After an initial SMT baseline was generated, we then explored (i) to what extent the system needed to be improved, and (ii) which techniques lead to the biggest improvement in translation quality and hence, decrease in human post-editing cost.

The language pair considered here is Brazilian Portuguese (*BP* henceforth) → European Portuguese (*EP*). Although Portuguese orthography

was standardised in 1990, considerable differences remain between the two varieties in a number of linguistic respects, including pronoun (e.g. 2nd pers. pronoun → BP *você* vs. EP *você/tu*) and verb usage (e.g. BP loss of the pluperfect tense) and other lexical differences (e.g. PB *autocarro* vs. BP *ônibus*).

The remainder of this paper is organised as follows. In Section 2, we demonstrate that while same language translation can be of real benefit in a number of use-cases, at the same time, very little previous work appears to have been carried out. In Section 3, we describe the data used to build the various systems, and provide the results using a variety of techniques in Section 4. In Section 5, we discuss some of the pertinent findings, and conclude in Section 6 with some avenues for future work.

2 Same Language Translation

Despite the number of potential applications for same language translation, there are only a few works which address the problem. To the best of our knowledge, there is no published research which has directly applied SMT to translate from one language variety to another.

However, SMT has been applied for two related tasks in two of our own papers. In patented work described in Cahill et al. (2009), we built an English-to-English system using our in-house MaTrEx system (Tinsley et al. (2008)) to generate an N-best list of outputs that could be used for improved target-language speech synthesis. In Penkale and Way (2012), we addressed the problem of translating a bad version of a language into a 'less poor' one. This was in the context of translating in-game text, where incorrect English – usually written by a non-native game developer – needs to be improved prior to localisation *per se*; translating the poor original English 'as is' would produce completely unintelligible output. Using post-edited data as the target-side of the training data, our SMT system was able to learn how to automatically post-edit some of the errors made by the source authors, in much the same way as Dugast et al. (2007) and Simard et al. (2007) have shown previously.

While we are unaware of any published work on the subject, it is clear that Microsoft have done something similar, albeit for a different purpose. They describe their 'Contextual Thesaurus'[1] as "an English-to-English machine translation system that employs the same architecture that the Microsoft Translator uses when translating different languages". They list a number of applications for this "large-scale paraphrasing system", including document simplification, language learning, plagiarism detection, summarization and question answering, to name but a few.

As to non-statistical approaches, only Zhang (1998) appears to have applied RBMT to translate from Mandarin Chinese to Cantonese. Murakami et al. (2012) adopted instead a two-stage translation pipeline where Japanese is first rendered in English through pattern-based translation, which is in turn translated into more correct English. Formiga et al. (2012) focused on improving the output of an English-to-Spanish SMT system, where correct morphology is generated in a post-translation morphological generalisation stage.

As well as the use-cases presented already, the current paper addresses a number of real-world problems, which are as yet unsolved in the translation and localisation industries. Notwithstanding the need to come up with a proper treatment of terminology, we believe that some of the techniques utilised in our work can be brought to bear in addressing two other crucial problems, namely outdated legacy Translation Memories (TMs) and the introduction of new company terminology. As far as the first of these is concerned, companies typically prune data according to its age; clearly this is a very arbitrary solution. With respect to the second, new terminology presented in company glossaries may not tally with legacy (but still useful) TM data.

3 Data and System Building

The data were provided in the form of Intel TMs – BP-to-EN and EP-to-EN, where the English side was common to both – in the area of software documentation and customer support. As it was translated and validated by human experts, the data provided by Intel was of very good quality. However, before training the engines, any punctuation and markup 'noise' still left in the data was removed via regular expressions.

Two phrase-based SMT systems were built using Moses (Koehn et al. (2007)). The first (referred

[1] http://labs.microsofttranslator.com/thesaurus/

Approaches	BLEU	TER	METEOR
Baseline	**.589**	**0.292**	**0.704**
+ DNT (exclusive)	.588	**0.292**	**0.704**
+ DNT (constraint)	**.589**	**0.292**	**0.704**
+ LEX SUB1	.577	0.301	0.697
+ RUL1 (all)	.260	0.504	0.445
+ RUL1 (freq>5)	.524	0.327	0.658
+ RUL1 (freq>10)	.529	0.324	0.661
+ LEX SUB & dict from aligned data (constraint)	.578	0.30	0.70
+ post-decoding LEX SUB	.588	**0.292**	**0.704**

Table 1: System A: automatic evaluation scores for the different approaches.

to below as *System A*) was trained using 63,137 length-ratio filtered sentences (approx. 687,410 tokens). A devset of 1,498 sentences (approx. 20,286 tokens) was used to tune the weights for the features in the log-linear model using MERT (Och (2003)). In comparison, the second system (*System B*) was trained on a larger set of 75,324 sentences (approx. 828,532 tokens) using a different devset containing 1,499 sentences (approx. 20,174 tokens). For both systems we used a single test set comprising 1,500 sentences.

4 Methodology and Results

The main goal of the present paper is to show which approach (or combination of approaches) leads to the biggest improvement in translation quality. In more detail, we explored the following options:

1. Guiding decoding to ensure technical terms are translated correctly via supplied dictionaries,
2. Using lexical substitution to replace Brazilian Portuguese words remaining in the output,
3. Using data-driven spelling rules to correct the translation output,
4. Using company-internal and data-driven bilingual dictionaries to both guide decoding and correct the translation output.

The results for System A are shown in Table 1, while those for System B are shown in Table 2. Column 1 shows each of the different system variants built, with columns 2–4 showing the BLEU (Papineni et al. (2002)), Translation Edit Rate (TER: Snover et al. (2006)) and METEOR (Lavie and Denkowski (2009)) scores, respectively. Note that for BLEU and METEOR, the higher the score the better, while for TER, a lower score is indicative of better quality.

In the next sections, we describe each experiment conducted with the results achieved.

4.1 Translation of technical terms

Intel provided us with a list of technical and product names that the system should not mistranslate or lose during decoding. In order to adhere to their requirements, we wrapped those terms in xml tags (i.e. ⟨DNT⟩ ... ⟨/DNT⟩) and used both the *exclusive* and *constraint* options implemented in Moses to guide decoding; *exclusive* forces the decoder to use a word input by the user as translation, while *constraint* allows the decoder to use *only* those phrases containing that word.

As seen from both Table 1 and Table 2, neither of the two options (*DNT (exclusive)* and *DNT (constraint)*) outperforms the Baseline; however, in Table 1, we see a small deterioration only for *DNT (exclusive)* in terms of BLEU, although more significant differences are seen in Table 2 for both options.[2] Accordingly, it might be said that these options do not appear to be too harmful, either. Forcing the decoder to select a specific word or phrase is likely to adversely impact the fluency of the translation which is otherwise ensured during phrase-based decoding (i.e. in the Baseline). Of course, in the majority of cases the baseline is able to translate these technical terms, merely by dint of these appearing in the TM from which the (correct) translations are learned; to us, this is not too surprising considering that the human translations on which the systems are trained are produced following strict guidelines. However, for many companies, correct rendering of terminology is of paramount importance and they are willing to sacrifice a small drop in (say) BLEU score as a trade-off; in practice, this deterioration in transla-

[2]Note that while it is surprising that the results in Table 2 are consistently lower, despite being trained on a larget data set, the results in Tables 1 and 2 are not directly comparable given that parameter estimation was performed on different devsets.

Approaches	BLEU	TER	METEOR
Baseline (w/ Intel content)	.583	0.295	0.695
+ DNT (exclusive)	.582	0.295	0.694
+ DNT (constraint)	.583	0.295	0.695
+ LEX SUB & dict from aligned data (constraint)	.571	0.305	0.685
+ post-decoding LEX SUB	.583	0.295	0.695

Table 2: System B: automatic evaluation scores for the different approaches.

tion quality is small enough to be of no real consequence to post-editors.

4.2 Lexical substitution

Here we used lexical substution as an attempt to replace words in the hypothesis translations that are still in Brazilian Portuguese. Here we assumed that the reference contains the correct *EP* variant, being human-translated material. We used an initial list of 982 item pairs provided by Intel. However, as shown in Table 1, this simple lexical substitution does not help translation, as words in the human-provided reference sentences do not tally with words described as 'European Portuguese' in the Intel lexicon. As an example, consider the dictionary items in (1):

(1) a. *mais→maior*
b. *confiança→considerar como fidedigno*

Now consult the behaviour in (2):

(2) a. EP reference: pode fazer compras com *mais confiança* em sites que passam os testes diários do serviço SECURE
b. EP translation baseline: pode efectuar compras com *maior confiança* em sites que passem os testes diários de Serviço SECURE
c. EP translation with lexical substitution: pode efectuar compras com *maior considerar como fidedigno* em sites que passem os testes diários de Serviço SECURE

As we can see, while the Baseline produces the correct form *maior* in (2b), it is penalised when compared to the reference in (2a). Furthermore, when we exercise the rule in (1b) to produce (2c) – as required by the Intel dictionary – we generate a translation which differs still further from (2a). Given this, it is perhaps surprising that this approach does not show large deteriorations in translation quality as measured by the automatic metrics in Table 1 (see line 5 'LEX SUB1'). However, we were convinced enough that relying only on such scores would not bring about translation improvements even on the larger set, so we omitted this experiment for System B.

4.3 Correcting the output using data-driven spelling rules

Another method to improve the quality of translation is to automatically extract spelling rules from the bilingual dictionary provided by Intel. These rules are then transformed into regular expression and applied to the test output *post hoc*. The algorithm takes into consideration each pair in the bilingual dictionary and sees which blocks differ and which operation has to apply in order to transform the source block into the target block. For instance, a *delete* type difference is detected between the pair in (3):

(3) BP:*detecção*→ EP: *deteção*

Consequently, we can extract a rule such as c→ø.

In order to exclude lexical differences (e.g. *assinatura→subscrição*) where block matching would yield rules that are not systematic (because they are not related to spelling differences), string-based similarity Levenshtein (1966) is calculated prior to rule extraction. If the pair has a similarity score greater than .6 (empirically determined), the rule is extracted.

At first we just extracted shallow rules resembling phonological rules which consider whether (i) the preceding or following letter is a vowel, (ii) the preceding or following letter is a consonant, and if so which consonant it is, and (iii) whether it is in sentence-initial or final position. For instance, a rule for *c*-deletion when preceded by a vowel and followed by ç is shown in (4):

(4) Vcç→Vøç.

To calculate improvement we then consider three different conditions: (i) *all*: all rules found are considered (RUL1 (all) in Table 1); (ii) *(freq.>5)*: all rules that were found more than 5 times are considered (RUL1 (freq>5)); and (iii) *(freq.>10)*: all

rules that were found more than 10 times are considered (RUL1 (freq>10)).

Again, the results in Table 1 do not show any improvement across all metrics. What is especially clear (cf. RUL1 (all)) is that it makes sense to limit the application of the rules to those that were found many times if extremely low performance is to be avoided. One problem we detected with this approach was that some rules were *over-generalised* and could have been grouped more wisely.

Given the poor results of the current rule extraction algorithm, we considered a refinement whereby the context is first over-specified and then generalised if a lot of different contexts for the same target block are found. Consider the two rules in (5):

(5) a. (? <=s)ão$ → ø (lit. delete ão when preceded by s)

b. (? <=ç)ão$ → ø (lit. delete ão when preceded by ç)

We found some preceding context in common and so were able to merge both rules in (5) into the rule in (6):

(6) (? <=[sç])ão$ → ø

However, yet again this method did not lead to any further improvement. One of the reasons why poor-quality rules are extracted is because the input comprised misaligned data. For example, the rule in (7) tells us to delete word-final 's' if it is preceded by either a, p, e or o:

(7) (? <=[apeo])s$ → ø

This works correctly for strings such as (8a), but not for (8b), where the form is the same in both EP and BP:

(8) a. *relatório de atividades → relatório de atividade*

b. *log de atividades → log de atividades*

Furthermore, it applies to strings that it shouldn't: *dos→do*).

4.4 Company-internal vs. data-driven bilingual dictionary

As we showed in Section 4.2, using the glossary supplied by Intel didn't help improve translation performance owing to mismatches with the reference translations. While the results in the previous section were disapppointing, we considered it to have some potential. Accordingly, we extracted instead a bilingual dictionary (omitting function words) using alignment information computed during training. This alignment information was filtered *post hoc* using fine-grained POS-tagging and morphological analysis using Freeling (Padró and Stanilovsky (2012)) for Portuguese.

However, again we were again unable to improve over the Baseline. Nonetheless, this approach (LEX SUB & dict from aligned data (constraint)) produces slightly better quality translations according to all three automatic evaluation metrics than the original LEX SUB1 method. Performing this model in a post-decoding phase causes results to improve still further, with results matching the Baseline in Table 1 for both TER and METEOR, although the BLEU score lags behind a little. In Table 2, with the larger training set, we see exactly the same thing: TER and METEOR scores match the Baseline, with BLEU just a little lower.

5 Observations

The fact that no method implemented leads to two different hypotheses.

Firstly, the baseline models are already able to learn very strong translation patterns (i.e. words and phrases), such that there is little need for modifications to be made. All other methods we tried lead either to errors, or to paraphrases that are still correct but which are sufficiently different from the reference translation to be unfairly penalised. For instance, the sentences in (9) are grammatical and almost identical in meaning to the reference, but an *n*-gram overlap-based metric such as BLEU fails to reward the two sentences appropriately.

(9) a. *Reference*: contacto do suporte (online ou telefone)

b. *Baseline*: contacte o suporte (Online ou Telefonico).

c. *Lex sub w/ aligned data (constraint)*: entre em contacto com o suporte (online ou por telefone)

That the baseline already is able to recognise some inter-language patterns can be seen in (10) and (11), where the baseline system is able to translate the *bp* construction *estar + gerundive* vs. *ep estar a + infinitive*:

(10) a. *Source*: [...] descobrimos que ele pode **estar tentando** vender algo que normalmente [...]
b. *Baseline*: [...] verificamos que pode **estar a tentar** vender algo que , [...]
c. *Lex sub w/ aligned data (constraint)*: *same as Baseline*

(11) a. *Source*: [...] este arquivo **esteja sendo** usado [...]
b. *Baseline*: [...] este ficheiro **está a ser utilizado** [...]
c. *Lex sub w/ aligned data (constraint)*: *same as Baseline*

Secondly, the reason why the basic model is able to learn translation patterns to a consistently high level is because all the material is from the same domain and of good quality, seeing as it is human translated and validated. We hypothesise here that if more heterogenous material were to be used for training (out-of-domain, possibly containing some errors, e.g. emanating from a 'light' post-edit), then lexical substitution based on the aligned data is likely to lead to an improvement over the baseline.

6 Conclusion and Future Work

In this paper, we bootstrapped a Brazilian Portuguese-to-European Portuguese SMT system from Intel TMs where the English side was common to both. We demonstrated that the performance of the Baseline engines was so strong that an array of techniques could not bring about any improvement as measured by three mainstream automatic evaluation metrics. Accordingly, what is essential is that a human evaluation be carried out, to see which translations are actually preferred by users. Given that the SMT system is producing a score of nearly 0.6 BLEU points on a large test set, our experience tells us that this may be immediately deployed in Intel with productivity gains for post-editors likely to be of the order of double their human translation throughput. Of course, this too needs to be verified, and the cost savings calculated once the engine is deployed in Intel's translation workflow.

Given that the methods used are language-independent, it can also be extended to other language variety pairs; those of immediate interest to Intel include ES-to-ES-xx and FR-to-FR-CA. Moreover, we have shown that applying language variety conversion can go far beyond simple content localisation, although for a large player like Intel, already helping just this use-case is likely to lead to significant savings.

As well as these topics, we aim to investigate whether deploying similar pre-processing techniques on the training data itself *before* engine building can lead to improved translation output. If successful, this will have important consequences for companies owning large amounts of legacy TM data, who will subsequently be able to curate their data sets in a more informed manner than is currently the case.

Acknowledgements

This work was partially supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of CNGL at Dublin City University. Thanks also to Loic Dufresne de Virel at Intel, and Steve Gotz at CNGL, for their support on this project.

References

Cahill, P., Du, J., Berndsen, J., and Way, A. (2009). Using same-language machine translation to create alternative target sequences for text-to-speech synthesis. In *Proceedings of Interspeech 2009, the 10th Annual Conference of the International Speech Communication Association*, pages 1307–1310, Brighton, UK.

Dugast, L., Senellart, J., and Koehn, P. (2007). Statistical post-editing on SYSTRANs rule-based translation system. In *ACL 2007: Proceedings of the Second Workshop on Statistical Machine Translation*, pages 220–223, Prague, Czech Republic.

Formiga, L., Hernández, A., Mariño, J., and Monte, E. (2012). Improving English to Spanish out-of-domain translations by morphology generalization and generation. In *Proceedings of the Monolingual Machine Translation Workshop – AMTA 2012*, pages 6–16, San Diego, CA.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen,

W., Moran, C., Zens, R., Dyer, C., and Bojar, O. (2007). Moses: Open source toolkit for statistical machine translation. In *ACL 2007: proceedings of demo and poster sessions*, pages 177–180, Prague, Czech Republic.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *HLT-NAACL 2003: conference combining Human Language Technology conference series and the North American Chapter of the Association for Computational Linguistics conference series*, pages 48–54, Edmonton, Canada.

Lavie, A. and Denkowski, M. (2009). The meteor metric for automatic evaluation of machine translation. *Machine Translation*, 23(2-3):105–115.

Levenshtein, V. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10:707–710.

Murakami, J., Nishimura, T., and Tokuhisa, M. (2012). Two stage machine translation system using pattern-based MT and phrase-based SMT. In *Proceedings of the Monolingual Machine Translation Workshop – AMTA 2012*, pages 31–40, San Diego, CA.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *ACL-2003: 41st Annual meeting of the Association for Computational Linguistics*, pages 60–167, Sapporo, Japan.

Padró, L. and Stanilovsky, E. (2012). Freeling 3.0: Towards wider multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012)*, Istanbul, Turkey. ELRA.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Philadelphia, Pennsylvania.

Penkale, S. and Way, A. (2012). SmartMATE: An online end-to-end MT post-editing framework. In *Proceedings of AMTA 2012 Workshop on Post-editing Technology and Practice*, 10pp., San Diego, CA.

Simard, M., Ueffing, N., Isabelle, P. and Kuhn, R. (2007). Rule-based translation with statistical phrase-based post-editing. In *ACL 2007: Proceedings of the Second Workshop on Statistical Machine Translation*, pages 203–206, Prague, Czech Republic.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In *AMTA 2006: Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, Visions for the Future of Machine Translation*, pages 223–231, Cambridge, Massachusetts, USA.

Tinsley, J., Ma, Y., Ozdowska, S., and Way, A. (2008). MaTrEx: the DCU MT system for WMT 2008. In *ACL-08: HLT. Third Workshop on Statistical Machine Translation, Proceedings (ACL WMT-08)*, pages 171-174, Columbus, Ohio, USA.

Zhang, X. (1998). Dialect MT: a case study between Cantonese and Mandarin. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 2*, pages 1460–1464, Montreal, Quebec, Canada.