Translating User-Generated Content in the Social Networking Space

Jie Jiang

Andy Way

Rejwanul Haque

Applied Language Solutions
Delph, OL3 5FZ
UK

{firstname.lastname}@appliedlanguage.com

Abstract

This paper presents a case-study of work done by Applied Language Solutions (ALS) for a large social networking provider who claim to have built the world's first multi-language social network, where Internet users from all over the world can communicate in languages that are available in the system. In an initial phase, the social networking provider contracted ALS to build Machine Translation (MT) engines for twelve language-pairs: Russian⇔English, Russian⇔Turkish, Russian⇔Arabic, Turkish⇔English, Turkish ⇔Arabic and Arabic⇔English. All of the input data is user-generated content, so we faced a number of problems in building large-scale, robust, high-quality engines. Primarily, much of the source-language data is of 'poor' or at least 'non-standard' quality. This comes in many forms: (i) content produced by non-native speakers, (ii) content produced by *native* speakers containing non-deliberate typos, or (iii) content produced by native speakers which *deliberately* departs from spelling norms to bring about some linguistic effect.

Accordingly, in addition to the 'regular' pre-processing techniques used in the building of our statistical MT systems, we needed to develop routines to deal with all these scenarios. In this paper, we describe how we handle shortforms, acronyms, typos, punctuation errors, non-dictionary slang, wordplay, censor avoidance and emoticons. We demonstrate automatic evaluation scores on the social network data, together with insights from the the social networking provider regarding some of the typical errors made by the MT engines, and how we managed to correct these in the engines.

1 Introduction

With the advent of Web 2.0, individual users have been able to actively participate in the generation of online content via community forums or social media. Online publishing is no longer the realm of large software companies and media organisations, with the Web open and accessible to an ever-larger percentage of the world's population.

Increasingly so, the influence of English as an Internet language is declining, so much so that recent data from June 2010 suggests that English-language users comprise just 27% of the overall Web population.[1] Nonetheless, while Carrera et al. (2009) acknowledge that user-generated content is suitable for machine translation (MT), they also state that most such content usually remains untranslated.

Indeed, there is little related work to date on translating such data. This paper presents a case-study in this area where we built a number of statistical MT (SMT) engines for a large Middle East-based social networking provider who claim to have built the world's first multi-language social network where Internet users from all over the world can communicate in languages available in the system.

In an initial phase, in order to facilitate communication between as many of its users as possible, the social networking provider contracted ALS to build MT engines for twelve

[1] http://www.internetworldstats.com/stats7.htm

engines for the language-pairs with most demand: Russian⇔English, Russian⇔Turkish, Russian⇔Arabic, Turkish⇔English, Turkish⇔ Arabic and Arabic⇔English.

All of the input data is user-generated content, which caused a number of problems when it came to building large-scale, robust, high-quality engines for the language-pairs in question. The main problem was where ’poor’ or ‘non-standard’ source-language quality was encountered; translating it ‘as is’ would have been pointless, as we would have suffered from the ‘garbage in garbage out’ problem.

There are essentially two use-case scenarios:

1. Much of the content is produced by non-native speakers, so the source-language data can be of very poor quality. In this case, we need to translate this into ‘good’ English prior to translation. This ‘monolingual translation’ is the theme of one of the workshops at AMTA-2012, and the pre-editing task we are confronted with here can perhaps be seen as the inverse of the statistical post-editing (SPE) solutions proposed a few years ago (Dugast et al., 2007; Simard et al., 2007).

2. In contrast, source content is authored by *native* speakers, where the author:

   (a) either entered the text too fast and so made typographical errors, or

   (b) *deliberately* departed from spelling norms to bring about some linguistic effect.

In the remainder of this paper, we provide an overview of related work in this area in Section 2. In Section 3, we give an overview of how the ALS statistical MT engines are built, focusing specifically on how the ‘regular’ pre-processing techniques needed to be extended to cope with the above problems, including dealing with shortforms, acronyms, typos, punctuation errors, non-dictionary slang, wordplay, censor avoidance and emoticons. In Section 4, we provide automatic evaluation scores on social network data, together with insights from the social network provider regarding some of the typical errors made by the MT systems, and how these were corrected in the engines. We conclude in Section 5, together with plans for future collaboration, especially the development of more engines as further language-pairs come onstream.

2 Related Work

Despite the obvious benefits in translating user-generated content, there do not appear to have been many published attempts at doing so.

Hecht & Gergle (2010) examine how knowledge representation differs in 25 different language editions of Wikipedia. They note that this diversity is greater than has been presumed to date, which will impact on applications that use Wikipedia as a knowledge-source. While they hypothesize how knowledge diversity can be leveraged to create “culturally-aware applications” and “hyperlingual applications”, they do not address MT directly.

Flournoy & Rueppel (2010) state that “Adobe is working to develop richer community-derived resources in many markets, including community translations and user community forums. MT has natural integration with both scenarios,” without stating precisely how they aim to go about this. Nonetheless, they do provide a useful table which summarizes the requirements for community-based translation, namely:

- Quality: Low-medium,
- Purpose: Gisting,
- Customization: Varied subject matter,
- ROI: Difficult to calculate,
- Security: Low,
- Language Pairs: Primarily EN→XX; also XX→XX,
- Input quality: Varied, uncontrolled.

More related to the topic of this paper, Roturier & Bensadoun (2011) and Mitchell & Roturier (2012) discuss ways in which machine-translated user-generated content can be best evaluated. In the former, four MT systems – Microsoft Translator, Systran, “a third-party commercial SMT system that was customized using Symantec translation memories”, and a system called ‘VICTOR’, “a standard phrase-based SMT system trained using Moses" (Koehn et al., 2007) with some extra pre-processing components – were compared using a range of automatic MT evaluation metrics in order to evaluate their suitability in translating user-generated content. In contrast, Mitchell & Roturier (2012) examine the perceived quality of MT among members of an online community forum, finding that albeit with quite a low response rate, the MT output was "comprehensible slightly more often than not" (p.64).

On a related topic, Banerjee et al. (2011, 2012) demonstrate how they customize an MT system to translate Symantec user-generated forum content. Apart from the work we describe in this paper, this appears to be the only research that actually demonstrates how user-generated content may be translated.

In a precursor to Flournoy & Rueppel (2010), Carrera et al. (2009) were probably the first to describe the requirements on translating 'cross-language social media' via MT, albeit in the context of cross-language data mining and social media analysis. They note that an MT system would need to be designed for:

- Large-scale, real-time translation,
- Meaning preservation,
- Robustness, especially in light of
- Errors in linguistic formalization.

Later in the paper, they make the same observation that we do, namely that a writer may "deliberate[ly] inten[d] to break conventional language use for stylistic purposes) currently intractable by MT technology". While they focus specifically on why SMT is particularly poor in this regard – for obvious reasons – and ultimately recommend a hybrid approach, in our paper we show in contrast that SMT is very capable of overcoming such problems.[2]

[2]If we accept that the regular expressions that we write to solve some such problems may be classified as 'rules' (cf. Section 3.2), then our approach too may be regarded as a hybrid solution. See `http://translation-blog.multilizer.com/hybridity-in-translation-overgilding-the-lily/` for our thoughts on the merits of claiming that one's system is 'hybrid' or not.

3 ALS MT Engines

Over the past 15 months, the ALS Language Technology (LT) team has continued the success it enjoyed in its previous guise as a world-leading academic MT group at Dublin City University by delivering major improvements in the speed, quality and usability of today's state-of-the-art statistical models of MT in an industrial environment. Taken together, we believe that these improvements have the potential to transform the current MT landscape.

In this paper we ignore SmartMATE,[3] our self-serve SMT environment (Way et al., 2011), and concentrate instead on describing how we build customized engines. These are SMT systems built offline by experienced ALS LT engineers, and guaranteed to outperform the (very good) SmartMATE baseline, with far greater pre- and post-processing, and the incorporation of a feedback and review phase.

3.1 Customized Engine Builds

The ALS MT technology comprises a number of components, including parallel corpus extraction, pre-processing, corpus cleaning, training data preparation, model training, tuning, translation and post-processing. While many of these processes rely on standard tools such as Giza++ (Och & Ney, 2003), IRSTLM (Federico & Cettolo, 2007), MERT (Och, 2003) and Moses (Koehn et al., 2007) – all areas where we have a proven track record, as demonstrated by our publications on word and phrase alignment, language modelling, tuning and decoding[4] – much of the success gained by our offering relies on the large number of pre- and post-processing routines that we have developed.

While we will not go into too much detail for obvious reasons, despite competitors' claims to own the 'clean data' space, we are confident that no suppliers pre-process our clients' data to the extent that we do.

3.1.1 Data Cleaning

In order to prepare good-quality training material, we perform three consecutive cleaning techniques

[3]`http://smartmate.co`

[4]`http://www.computing.dcu.ie/~away/pubs.html`

misspelled	soundex code	right format	soundex code
speling	S1452	spelling	S1452
c@@l	C400	cool	C400
how r	H600	how are you	H600
h r u	H600	how are you	H600
LOL	L000	laugh out loud	L2343
tmrw	T560	tomorrow	T560
wtf	W310	what the fuck	W312
sayin	S500	saying	S520
whatdoyouwant	W312	what do you want	W312
iirc	I620	if i remember correctly	I1651626234

Table 1: Using soundex on the phonetic level to cope with non-standard input

on the data: (i) cleaning based on empty source and target sentences, (ii) removing duplicate sentence pairs, and (iii) cleaning based on source and target sentence-length ratio. Typically, anything from 10–25% of the data supplied by users is deleted in pre-processing. This may vary on a per language-pair basis, even for the same client.

All of this aggressive pruning demonstrates that high-quality translation results may be achieved, despite reducing the amount of training data. What is quite clear from our engine development is that, contrary to the often heard mantra that 'more data is better data', it is more important to do better with less data.

3.1.2 Data Pre-processing

For the social network scenario, initially there was no client-specific parallel data for us to use,[5] but they were able to provide us with a fair quantity of monolingual data in a range of languages which we were able to use to help improve our language models (LMs). Initially we used mainly OPUS[6] sub-corpora (Tiedemann, 2012) as parallel training data of a 'similar' type, since we extracted sentence-pairs that were constrained by length. This required a large amount of corpus cleaning to remove badly aligned sentences. Target LMs improved greatly by mining data from tweets and similar sources. In order to ensure fast runtime performance, we massively pruned the phrase-tables with only a small degradation in translation quality (cf. Johnson et al., 2007).

Once we have clean parallel data, we perform seven further stages of pre-processing, which includes ensuring the correct text encoding, handling URLs and other special characters (e.g. pipes, quotation marks, brackets), as well as the usual MT processes of tokenisation and lowercasing.

3.2 Adjustments to Pre-Processing

In the introduction, we described three main scenarios where we encounter wrong or 'non-standard' user-generated source-language content.

Where typos have been made owing to authors entering the text too fast, we can use a spellchecker if what has been typed is 'not too far away' from what was intended (as measured by edit-distance: Levenshtein, 1966).

For the more general case of poor non-native competence, in another application scenario (Penkale & Way, 2012), we were provided with a reasonably large collection of original mangled English and the edited versions, and we treated this as an MT task in its own right, in the same way that SPE works, i.e. the 'bad' original English was the source language, and the 'good' post-edited English was the target, and the MT system learnt how to correct many of the errors in the source automatically. This dramatically cut down on the costs for our client. Of course, as we receive more and

[5] Prior to coming to ALS, similar related work involved the SMT systems we built for the 2010 World Cup – `http://www.computing.dcu.ie/news/cngl-launch-world-cup-twanslation-service` – where in just a few days, we built 12 engines from scratch to translate in real time online tweets with the WC2010 hash tag. See Lewis (2010) for a similar time-constrained application.

[6] `http://opus.lingfil.uu.se`

Bilingual training corpora			
language-pairs	Sentences	Words (S)	Words (T)
English⇔Russian	3,409,848	17,954,459	16,628,155
English⇔Arabic	3,290,746	18,874,069	17,649,748
English⇔Turkish	9,982,554	45,394,473	39,685,143
Arabic⇔Russian	4,439,740	41,925,655	47,497,922
Arabic⇔Turkish	1,845,990	9,337,707	8,849,296
Russian⇔Turkish	1,517,034	7,365,201	6,817,511
Mined Data (Monolingual)			
Language	Sentences	Language	Sentences
English	4,365,917	Russian	493,257
Arabic	876,137	Turkish	323,662

Table 2: Corpus Statistics

more post-edited MT output, we will naturally obtain more training data, which is likely to cause MT quality to improve still further, and incrementally cut down more on post-editing.

Note also that the nature of the errors differed substantially from those in the 'native speaker' scenario, so in order to further improve performance, we used a soundex-like algorithm (Odell & Russell, 1922) to operate on the phonetic level; using edit-distance does not work here as too many edits are required, so that the intended inputs are 'too far away' from the actual input. We provide some examples of this in Table 1.

Regarding the 'non-standard' input, some previous interesting work already exists on text normalization for Internet data (Clark & Araki, 2011). They classified the range of phenomena to be dealt with as shortforms (nite (*night*), sayin (*saying*), gr8 (*great*)), acronyms (lol (*laugh out loud*), iirc (*if I remember correctly*)), typing errors/misspellings (wouls (*would*), rediculous (*ridiculous*)), punctuation omissions/errors (im (*I'm*), dont (*don't*)), non-dictionary slang (that was well mint (*that was very good*)), wordplay (that was soooooo great (*that was so great*)), censor avoidance (sh1t, f***), and emoticons (:) (smileys), <3 (heart)). We have already spoken about typographical errors, but some shortforms, punctuation errors and attempts at censor avoidance can be dealt with in the same way, while our soundex-like algorithm deals with many acronyms. We wrote a set of regular expressions to handle the issue of 'wordplay' (e.g. 'cooooooool').

In addition to the phenomena classified by Clark & Araki (2011), we also handle named entities as part of our pre-processing routines, and in addition identified the case where foreign words are used intentionally, and which need to be kept intact in MT (e.g. 'al dente').

As can be seen, our ability to handle all these phenomena has added considerably to the amount of pre-processing that we carry out. Nonetheless, given the ever-increasing amount of data generated by users, as opposed to large multinational organisations, this puts us in a very good position to identifying ALS as *the* MT provider of choice when it comes to translating user-generated content.

4 Translation Performance

In this section, we provide an overview of the data used in our task. We then report automatic evaluation scores using our MT engines together with an analysis of the performance of the systems. We give some typical errors made by the engines, and explain how we corrected these errors exploiting feedback provided by the social network provider.

4.1 Data Statistics

As stated in Section 3.1.2, we initially developed MT systems using bilingual corpora mainly sourced from OPUS sub-corpora, given that the characteristics of this data resemble that of the social network provider's user-generated content to a certain extent. However, this bilingual data contains a lot of noise, including errors from automatic sentence-alignment as well as wrong encoding. Special routines were needed to clean up this corpus (cf. Section 3.1.2). In Table 2, we provide the statistics of the training data with the number of sentences and source (S) and target (T) running words. Note that as anticipated by Carrera et al. (2009), in the context of the social network at the centre of this study, three quarters of the language-pairs do not feature English.

The last two rows in Table 2 provide statistics of the data mined from tweets and other similar data. This was used to build additional language models for each language. We found that when used in a 2-LM set-up with the 'regular' LM built from the target-sides of the social network provider's user-generated content, the fluency of the MT output was improved, and we were better able to generate translations more similar to the client's user-generated content. We also used this additional LM to help select a 1500-sentence development set, using perplexity values to select the most similar source–target pairs to the customer's user-generated content.

Apart from the parallel and monolingual data just described, we also produced a slang dictionary, containing more than 5,000 entries of Internet slang with English explanations. We manually translated all entries into the other three languages to obtain a multilingual slang dictionary. A small sample of the English slang dictionary is presented in Table 3.

slang	English
atb	all the best
brbs	be right back soon
cut3	cute
dtb	don't text back

Table 3: Samples of slang dictionary with English explanations

This dictionary is used in the training stage to enhance our source text correction method (cf. Section 3.2), and part of this dictionary is also used to generate a slang glossary to ensure that colloquialisms are translated correctly. We explain the use of glossaries in Section 4.3, as well as discussing some of the translation edits suggested by the users of the featured social network.

4.2 Automatic Evaluation Results

Once we had built excellent quality MT systems for the social network provider (cf. Section 3.1), we deployed these inside our SmartMATE toolkit so that they could be accessed by the client via our API.

Table 4 reports BLEU (Papineni et al., 2002) scores for forward (F) and reversed (R) MT systems for each language-pair obtained on the test sets, i.e. for English⇔Russian in the first line in Table 4, English→Russian would be the 'forward' system, with Russian→English being the 'reversed' engine.

Systems	BLEU (F)	BLEU (R)
English⇔Russian	86.49	91.01
English⇔Arabic	71.10	88.39
English⇔Turkish	79.65	80.78
Arabic⇔Russian	78.29	72.30
Arabic⇔Turkish	72.07	68.06
Russian⇔Turkish	90.54	88.72

Table 4: BLEU scores of the forward (F) and reversed (R) MT systems.

We can see from Table 4 that the BLEU scores on 1000 held-out sentences from the data described in Table 2 are very high for all systems (max. 91.01, min. 68.06), demonstrating clearly the utility of the engines for the task at hand. We also observe that those systems with English as target produce higher BLEU scores than when English is the source language: around 5 points higher (5% relative) for Russian, 17 points higher (24% relative) for Arabic, but just over 1 point higher (1.4% relative) for Turkish.

Other observations are that engines involving Russian give a better score when Russian is the source (except for Arabic); Turkish as target gives better scores than when it is the source (except for English), while Arabic is always better when used as source. Of course, Arabic has a very morphologically rich lexicon, whereas English exhibits much lesser lexical variation, which explains some of the differences in BLEU in Table 4.

4.3 Incorporating Suggestions from our Users

We are gradually improving our MT systems by incorporating suggestions from users of the social network. At the moment, most feedback concerns wrong lexical selection by the engines. One amus-

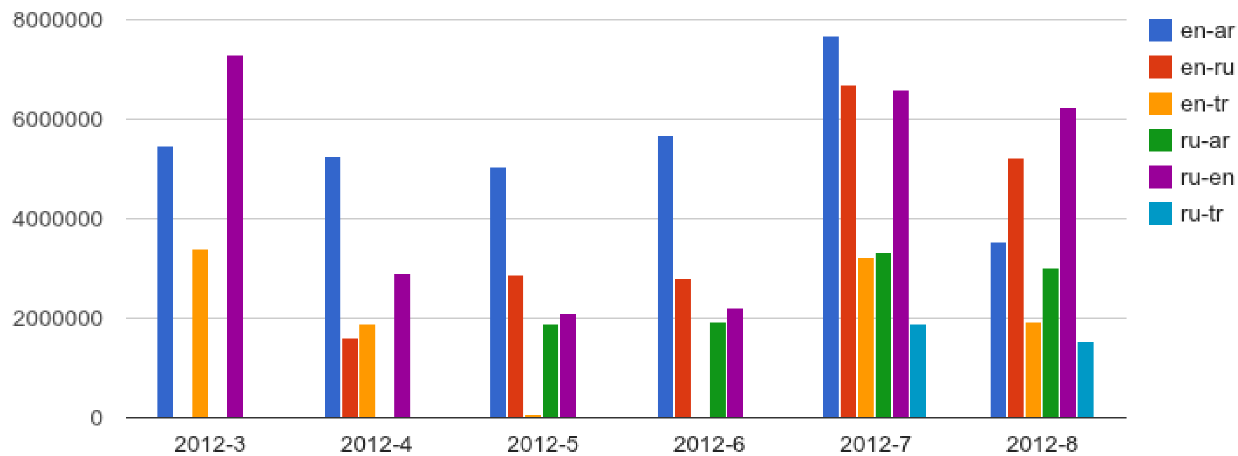

Figure 1: Translated word counts per engine from English and Russian

ing example concerned the translation of the English word ‘nice’ into Russian, where it was always translated as the name of the French city ‘Nice’, which is understandable given that we lowercase all input strings.

To incorporate the editing requests from users of the social network, ALS used the following two methods to improve translation quality:

- *Corpus editing and engine retraining*. Regular expressions were created to modify the original training corpus, and a new training corpus was generated by combining the modified training corpus and editing requests as extra data to retrain the engine.

- *Glossaries for MT*. If the previous method was unsuccessful in dealing satisfactorily with the editing requests from the social network provider, we added source–target equivalents into the glossaries (where the slang dictionary (cf. Section 4.1) already resided) to force the MT engines to translate the words in the glossary.

These two methods solved most of the problems. However, there are still some limitations: the first method cannot always guarantee that the words will be correctly translated after rebuilding the engine, and the second ‘forced replacement’ method may be sub-optimal when all editing requests are considered. Accordingly, in the near future, we aim to address this problem by introducing context-sensitive decoding, as described in Haque et al. (2011).

4.4 Translation Statistics

The service provided by ALS for the social network provider is an example of ‘always on’ online translation. All communication is passed through RESTful APIs. Typical translation speed is 2000 words per minute. From February (when basic API testing only took place) to mid-August 2012, a total of over 135 *million* words have been translated, broken down per month in Table 5.

Time	Translated Words
2012-02	71,779
2012-03	16,182,075
2012-04	16,608,694
2012-05	23,298,287
2012-06	18,843,487
2012-07	36,952,204
2012-08	23,301,706

Table 5: Translated word counts in each month

From Table 5, we can see that there is a huge demand for translation on this social network, with the number of translation requests growing on a monthly basis. For engines translating from English and Russian to other languages – the most heavily queried systems – we separately illustrate the translated word counts from March to August in Figure 1. Note that this graph indicates which order the MT engines came onstream.

As we pointed out earlier, Flournoy & Rueppel (2010) predicted that when building MT engines to handle user-generated content, the demand would primarily be EN→XX. However, as Figure 1 demonstrates, demand for translation *into* English remains strong, especially from Russian (27 million words) and (not shown in the graphic) Arabic (16 million words). The most heavily used language pair is English-to-Arabic, with a total of 32.6 million words translated in a five and a half-month period (around 7 million words/month, currently).

Flournoy & Rueppel (2010) also expected demand for XX→XX pairs, where English does not feature. The Russian-to-Arabic engine has translated over 10 million words, with over 8 million words having been translated for the reverse language direction. The Russian-to-Turkish engine has translated 3.5 million words in just over a month and a half, with 2.8 million words translated for Turkish-to-Russian over 4.5 months, albeit at an increasing rate on a monthly basis (on schedule to translate 3 million words in August 2012).

5 Conclusions and Future Developments

In this paper, we described a partnership between ALS and a large social network provider to provide SMT engines for 12 language-pairs to allow Internet users to communicate with one another in a large multi-language social network.

We described the MT system-building process in ALS, focusing specifically on how the 'regular' pre-processing techniques needed to be extended to cope with the particulars of the user-generated content at hand.

We provided scores for each of the language-pairs in this study on the initial (mostly) OPUS training data, as well as following the incorporation of additional training material and feedback from the client. We showed the huge translation requirements on our engines servicing communication in the featured multilingual social network. We demonstrated that this demand is increasing on a monthly basis, and provided details on throughput for the language-pairs in question.

Plans for future collaboration centre on the development of more engines as further language-pairs come onstream. In the near future, the addition of Chinese will lead to an additional 8 language pairs. Discussions regarding even tighter cooperation between ALS and the social network provider are ongoing, which may soon lead to further exciting announcements leading to more improvements to and functionality in the social networking space.

References

Pratyush Banerjee, Sudip Kumar Naskar, Johann Roturier, Andy Way and Josef van Genabith. 2011. Domain Adaptation in Statistical Machine Translation of User-Forum Data using Component Level Mixture Modelling. In *Proceedings of Machine Translation Summit XIII*, Xiamen, China, pp. 285–292.

Pratyush Banerjee, Sudip Kumar Naskar, Johann Roturier, Andy Way and Josef van Genabith. 2012. Domain Adaptation in SMT of User-Generated Forum Content Guided by OOV Word Reduction: Normalization and/or Supplementary Data? In *Proceedings of the 16th Annual Meeting of the European Association for Machine Translation*, Trento, Italy, pp. 169–176.

Jordi Carrera, Olga Beregovaya and Alex Yanishevsky. 2009. Machine Translation for Cross-Language Social Media. AAAI. `http://www.promt.com/company/technology/pdf/machine_translation_for_cross_language_social_media.pdf`

Eleanor Clark and Kenji Araki. 2011. Text normalization in social media: progress, problems and applications for a pre-processing system of casual English. *Procedia - Social and Behavioral Sciences* **27**(2): 2–11.

Loïc Dugast, Jean Senellart and Philipp Koehn. 2007. Statistical post-editing on SYSTRAN's rule-based translation system. In *ACL 2007: proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic, pp. 220–223.

Marcello Federico and Mauro Cettolo. 2007. Efficient handling of n-gram language models for statistical machine translation. In *ACL 2007: proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic; pp. 88–95.

Raymond Flournoy and Jeff Rueppel. 2010. One Technology: Many Solutions. In *AMTA 2010: the Ninth conference of the Association for Machine Translation in the Americas*, Denver, CO, 6pp.

Rejwanul Haque, Sudip Kumar Naskar, Antal van den Bosch and Andy Way. 2011. Integrating Source-Language Context into Phrase-Based Statistical Machine Translation. *Machine Translation* **25**(3): 239–285.

Brent Hecht and Darren Gergle. 2010. The tower of Babel meets web 2.0: user-generated content and its applications in a multilingual context. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*, ACM New York, NY, pp. 291–300.

Howard Johnson, Joel Martin, George Foster and Roland Kuhn. 2007. Improving Translation Quality by Discarding most of the Phrasetable. In *EMNLP-CoNLL 2007: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Prague, Czech Republic, pp.967–975.

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 *ACL 2007: proceedings of demo and poster sessions*, Prague, Czech Republic, pp.177–180.

Vladimir Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady* **10**: 707–710.

Will Lewis. 2010. Haitian Creole: how to build and ship an MT engine from scratch in 4 days, 17 hours, & 30 minutes. In *EAMT 2010: Proceedings of the 14th Annual conference of the European Association for Machine Translation*, Saint-Raphaël, France. 8pp.

Linda Mitchell and Johann Roturier. 2012. Evaluation of Machine-Translated User Generated Content: A pilot study based on User Ratings. In *Proceedings of the 16th Annual Meeting of the European Association for Machine Translation*, Trento, Italy, pp. 61–64.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *41st Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, Sapporo, Japan, pp. 160–167.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics* **29**(1): 19–51.

Margaret Odell and Robert Russell. 1922. *U.S. Patent Number 1,435,663*. U.S. Patent Office, Washington, DC.

Kishore Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *40th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, Philadelphia, PA, pp. 311–318.

Sergio Penkale and Andy Way. 2012. An Online End-To-End MT Postediting Framework. In *Proceedings of AMTA Workshop on Post-editing*, San Diego, CA. (under review).

Johann Roturier and Anthony Bensadoun. 2011. Evaluation of MT Systems to Translate User Generated Content. In *Proceedings of Machine Translation Summit XIII*, Xiamen, China, pp. 244–251.

Michel Simard, Nicola Ueffing, Pierre Isabelle and Roland Kuhn. 2007. Rule-based translation with statistical phrase-based post-editing. In *ACL 2007: proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic, pp. 203–206.

Jörg Tiedemann. 2012. Parallel Data, Tools and Interfaces in OPUS. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'2012)*, Istanbul, Turkey, pp. 2214–2218.

Andy Way, Kenny Holden, Lee Ball and Gavin Wheeldon. 2011. SmartMATE: Online Self-Serve Access to State-of-the-Art SMT. In *Proceedings of the Third Joint EM+/CNGL Workshop "Bringing MT to the User: Research Meets Translators", JEC 2011*, Luxembourg, pp. 43–52.