SmartMATE: Online Self-Serve Access to State-of-the-Art SMT

Andy Way Kenny Holden Lee Ball Gavin Wheeldon

Applied Language Solutions
Delph, UK
firstname.lastname@appliedlanguage.com

Abstract

Access to good quality Machine Translation (MT) has never been as easy as it is today. Portals such as Google Translate and Bing Translator facilitate huge amounts of translation requests on a daily basis, for an ever increasing spectrum of language pairs. People are finding many uses for the raw MT output provided by these, and other, freely available engines on the web, including gisting, assimilation, first drafts of translations for dissemination, etc. However, each of these systems is a 'one-size-fits-all' solution, where no customization is available to the user. One alternative is to purchase a system, which may be overly expensive, or sub-optimal for the type of documentation required to be translated. Another alternative is to install a freely available system such as Moses, but this may prove unduly onerous for the naïve user. In this paper, we present a portal which facilitates self-serve MT using state-of-the-art statistical MT (SMT). This is currently free for anyone to access and personalise their system by uploading their own Translation Memory(TM) and glossaries. By means of a simple key-press, optimal training, development and test data are created on-the-fly, which are then used for an automatic system build, with the results published in very acceptable amounts of time, together with automatic evaluation scores. According to user trials, this – together with built-in TM and online editing functionality – is a very exciting development, with the potential to vastly expand the user-base for self-serve SMT on a global basis.

1 Introduction

Given the recent severe global financial downturn, economic drivers are increasing the importance of MT: there are pressures on translation costs, and automation is key in driving such costs down. At the same time, the volume of material which is available for translation is increasing; in his keynote address at the AMTA 2010 conference in Denver, Mark Lancaster, CEO of SDL, stated that as much as 90% of what could currently be translated is not being translated. Furthermore, Common Sense Advisory have conducted research which shows that 98% of content is never translated (DePalma and Kuhns, 2006). In the same document, they also note that "of the 1000 websites from the worlds biggest companies and top brands, 45% are still single language sites". For these key reasons, as well as the fact that for some language pairs human translators are in very short supply (or even absent altogether), it is vital that the next generation of MT systems can be created rapidly, in a flexible environment, personalised to the requirements of the user, while at the same time ensuring that translation quality continues to improve.

Corpus-based approaches to MT – especially statistical models (SMT) – currently dominate the landscape. They predominate almost entirely in the academic research and development space, and even in industrial settings, a transition from the rule-based (RBMT) approaches which have been prominent for around 40 years to SMT is becoming ever more evident. SDL have recently acquired Language Weaver, Lionbridge have aligned themselves strategically with the IBM SMT systems (Ittycheriah and Roukos, 2007), GoogleTranslate and the Bing Translator dominate the field of freely available general-purpose online MT – for the time being, at least – and large organisations such as the European Patent Office have started to deploy SMT models over previous RBMT systems.[1] Other main RBMT providers either have already added statistical components (e.g. Systran (Dugast et al., 2007)) or are seeking to replace their old, unwieldy RBMT systems with new, flexible, customizable statistical engines.

Perhaps the clearest reflection of the exploding demand for translation can be seen in the remarkably widespread adoption of Google Translate, which has radically lowered the barrier to entry to use of MT, attracting millions of users worldwide. This worldwide user community is increasing rapidly, and new language pairs continue to be added which attracts still more users for whom no prior MT service existed. No special hardware or software is needed; it is available online, on-demand, and (currently) for free.

Nonetheless, Google Translate is no panacea; it is a 'one-size-fits-all' model, allowing no customization. A user cannot specify the training material to be used by the translation model, add in new lexical entries to reduce out-of-vocabulary items in the target language model, or add in specific termbanks, glossaries and/or in-domain lexicons. While there is a facility for users to upload alternative translations than those output by the system, we are not aware of any studies that show this to be effective for particular end-users when translating similar documents in the future. Furthermore, no confidence measures exist to comfort the user that the translation being suggested by the system is actually of good quality. Security is also a major concern, especially for industrial users with a translation requirement. In sum, while Google Translate clearly is useful for millions of users, it is a 'passive' MT service with major limitations.[2]

At the other end of the SMT spectrum, we have the Moses platform (Koehn et al., 2007).[3] Moses is open-source and free, and it can be installed on a user's local machine and seeded with that user's training data, with translations obtained in a reasonable amount of time (especially compared to building an MT system from scratch). Thus Moses too has lowered the barrier to entry to widespread use of MT, but from a development point of view. Like Google Translate, the size of the Moses' user/development community is increasing, especially among academic sites. But for the naïve user, Moses is not really an option at this stage; a sizeable amount of programming knowledge is required to install it correctly and have it run properly.

Thus while Google Translate is easily accessible to the wide community of users, it cannot be customised, thus failing to offer what is arguably the greatest potential benefit of SMT. Moses on the other hand does allow customization, but only through a cumbersome and limited process, that is out of reach for the wide community of users.

In this paper, we describe SmartMATE, a portal which allows users to subscribe to the platform, upload Translation Memories (TMs) and other appropriate files, and have a phrase-based SMT system built, tuned and tested on a cloud-based architecture, with document-level BLEU[4] scores (Papineni et al., 2002) output with the target-language translations as an indicator of MT performance. Post-processing of the translations is also facilitated via a built-in editing environment. Hiding the internal system-build requirements from the user opens up access to state-of-the-art SMT technology to a wide variety of potential users who for

[1] `http://www.pluto-patenttranslation.eu/index.php?q=espacenet`

[2] While in the main Google Translate has been a boon to all in the MT community, the recent announcement that it would be used for the translation of poetry (Genzel et al., 2010) sends out the wrong signals to the majority of users, who are in the main naïve to the types of documents that are best suited for MT. In the short term, at least, this move has the potential to undo some of the good that Google has undoubtedly brought to our field, as it has been known for quite some time that this type of text is unsuitable for MT (Hutchins and Somers, 1992).

[3] `http://www.statmt.org/moses/`

[4] We expect instead to output TER (Snover et al., 2006) scores, as these give an approximation of the amount of post-editing required, so are likely to be more comprehensible.

many reasons – a perceived lack of understanding of SMT, insufficient computational knowledge to install and customise such systems, or insufficient access to hardware in order to train, tune and test such engines – are currently disenfranchised.

The remainder of this paper is organised as follows. In Section 2, we briefly revisit the state-of-the-art SMT technology which predominates today, together with a discussion of its current suitability to the naïve user. In Section 3, we provide an overview of the tools currently in the translator's armoury, and discuss how access to MT can improve the capacity of such users. In Section 4, we describe our tool, and outline some areas where we intend to extend the functionality of the tool in Section 5. In Section 6, we conclude.

2 Users of State-of-the-Art Statistical Machine Translation

Statistical Machine Translation (SMT) is indisputably the dominant paradigm used today in the field of MT, especially in the research area. From the initial presentation (Brown et al., 1988) in the late-80s through a series of influential papers that appeared in *Computational Linguistics* (Brown et al., 1990; Brown et al., 1993), SMT is clearly much more heavily used today than other approaches such as RBMT (cf. (Hutchins and Somers, 1992)) and Example-Based MT (cf. (Carl and Way, 2003)).

As a prerequisite, SMT needs a parallel corpus of aligned source–target translations. SMT employs three distinct and separate processes: training, tuning (also known as 'parameter estimation') and decoding (or 'testing'). The training phase involves extracting a statistical model of translation from a parallel corpus, and a statistical model of the target language from a (typically much larger) monolingual corpus. In a log-linear framework (Och and Ney, 2002), tuning involves optimising the weights of the various features used in the n-gram-based SMT model, usually via Minimum Error Rate Training (MERT, (Och, 2003)), to try to ensure that the automatic evaluation score given to the translations of the test set are as high as possible, according to a specific metric (usually BLEU).

There are two main ways of availing of state-of-the-art MT technology today: (i) accessing free, online systems such as Google Translate or Bing Translator; or (ii) downloading and installing a freely available open-source toolkit such as Moses on a user's own machine. Most Moses developers are in large academic MT groups, or in large industrial enterprises; the ramp up for building one's own system from scratch may be considered to be inordinately high, so we do not consider this option further in this paper.

Current alternatives involve making available MT system components via Web services, as in the Panacea project (Pecina et al., 2011),[5] or to offer a fully-fledged, customizable MT service, as in our approach. Both approaches take the problem of configuring a system away from the user. In Panacea, users can combine different web services using a graphical user interface in a workflow editor. To date, however, access is available only for a suite of well-known word aligners, and there is no plan to integrate all MT components by the end of the project (December 2012).

In our approach, we allow users to upload their own TMs and glossaries, and with one simple click, an SMT system using Moses running on the Amazon cloud is built, with the results on th extracted test set returned to the user accompanied by BLEU scores (cf. footnote 4). If they require, a limited amount of post-processing is facilitated via regular expressions. This takes away completely the need to know how to install and run Moses, but nevertheless leaves the user in control, with the ability to personalise the system by uploading their own language resources.[6]

In addition, the 'fear factor' is done away with altogether; Way and Hearne (2011) describe how central linguists and translators are to the MT process – after all, where do the parallel resources come from in the first place? – so that SMT developers and researchers may better understand how to include these groups in continuing to advance the state-of-the-art. However, many users do not understand how SMT works, so in an accompa-

[5] http://www.panacea-lr.eu/

[6] Note that other companies, including PangeaMT, Let's MT, Asia Online, and LanguageStudio, claim to have similar functionality, but none are as accessible as ours, and some of the above-mentioned offerings do not allow seamless integration with TM and online editing environments.

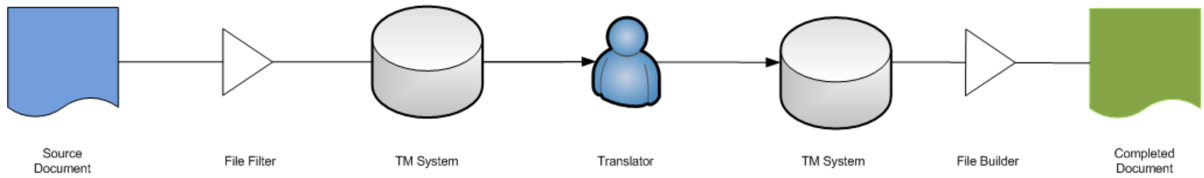

Figure 1: Typical MT + Postediting Workflow

nying paper, Hearne & Way (2011) attempt to provide a description of SMT in layman's terms. While we agree that at least a basic understanding of how the technology works should be striven for by all users, the need to understand how Moses works – which is currently an absolute prerequisite for its installation and use – is done away with here. We posit that in being able to run Moses without knowing any of the computational details, the experience gained by new users will attract them to our field, and break down the current resistance from many in the translator community towards using MT.

3 The Place of SMT in the Translator's Toolkit

A translator's typical workflow is shown in Figure 1. A source document is received in one of a number of different possible formats (.rtf, .xml, .txt, .pdf etc.), which may then be put through a file filter to produce a .tmx file which facilitates seamless integration with the user's TM system, the main tool in the translator's armoury. Translators have long relied on TMs as the main technology to assist translation, having invested huge amounts of time and effort over many years to keep these up-to-date.

Furthermore, translators are used to working with TM fuzzy match scores (Sikes, 2007), as these offer a good approximation of the amount of post-editing effort they will have to put in to amend the fuzzy match(es) to an appropriate target-language sentence. In turn, this is useful when estimating how much a translator should charge for a particular job.

Over the years, despite many efforts to establish the contrary, an old chestnut that raises its ugly head even today is that MT will put translators out of work. As we mentioned earlier, given the current information deluge, only a fraction of material that should be translated actually is translated, and for that material which is to be disseminated publicly, there will always be a human in the loop to ensure that only good quality output is published.[7] Furthermore, MT is just not suited to translating certain types of material (cf. footnote 2), and even if suitable material exists in a particular source language, SMT might still not be good enough (or even possible) given a lack of parallel text for a certain language-pair, and RBMT might be impossible given the lack of appropriately qualified linguists for that pair. In sum, there will always be a need for human translators; in fact, given the current demand for translation, despite the large number of translator training courses available, an insufficient number of translators are leaving Universities with appropriate qualifications.[8]

This latter, together perhaps with the current global economic situation, is leading many observers to conclude that without MT, the gap between what needs to be translated and what actually appears in the target language is set to increase further. Accordingly, many proactive translators already use translation engines to produce first drafts of texts, that then help them to produce the final version. Of course, it goes without saying that all translations produced are accredited by a human, or in some cases, up to three: the translator, and two proofreaders. Many translation companies provide this level of service, albeit at premium rates. Nonetheless, the level of profit made is not huge, given the number of people involved. Greater returns on investment can

[7] For a discussion of the central role played by translators and translations in SMT, see (Way and Hearne, 2011).

[8] cf. the survey of the Canadian translation industry: `http://www.uottawa.ca/associations/csict/represum.pdf`, p.49

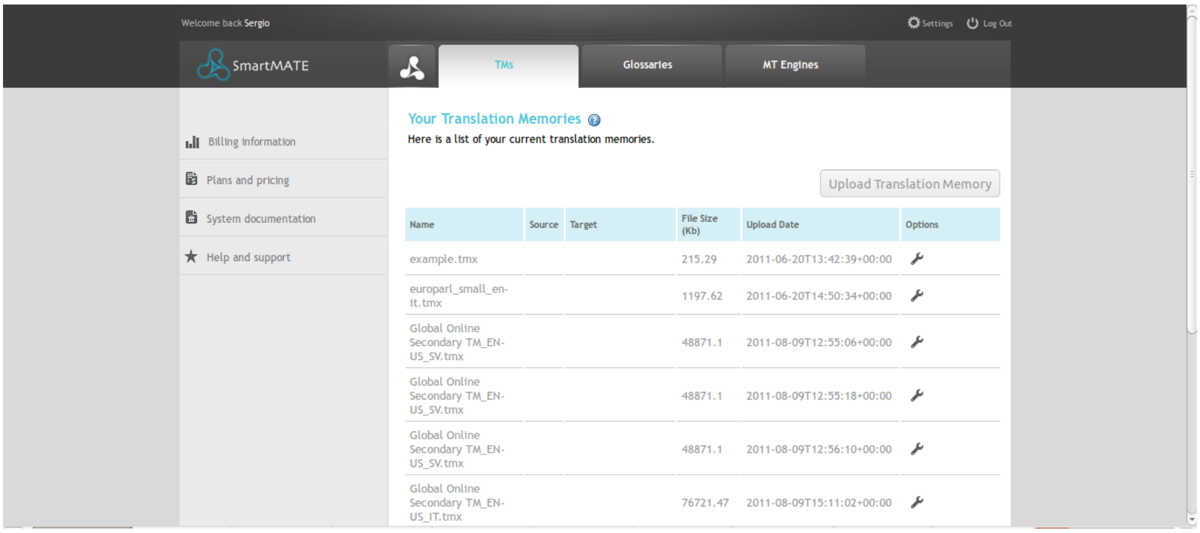

Figure 2: The SmartMATE Translation Assets Environment

be seen for post-edited MT solutions, with even higher profits generated where raw MT suffices.

Nonetheless, even for translators who are positively disposed towards MT, there remains some reluctance to use technology other than TM which could have a potential negative impact on their productivity, in the short term at least. This is clearly much less of a problem when using a free online system such as Google Translate, but is a real concern when attempting to install a purchased RBMT system (most likely, given the proliferation of such systems available for purchase, compared to engines under active development), or when trying to get to grips with an open-source MT toolkit such as Moses. Currently, a significant level of computational expertise is required, and even if this problem can be overcome, a hi-spec laptop or (more likely) access to a reasonably-sized hardware cluster is a prerequisite, given the amount of memory, disk space and cutting-edge CPUs required to build, tune and test such a system. In the next section, we describe how our tool overcomes these issues, and makes state-of-the-art SMT directly available to any user, by delivering a fast, cost-effective, high-quality translation solution, balanced with user control in a live environment.

4 Enabling User Access to State-of-the-Art SMT

SmartMATE is a portal which anyone can currently register with free of charge. Once registered, the user can securely upload TMs and other parallel data and use these translation assets to build an SMT engine (cf. Figure 2). When creating an engine configuration, before the build process the user is able to indicate which TMX files should be considered a glossary and which should be part of the engine build.

Of the hundreds of SmartMATE engine builds to-date, the smallest TM uploaded comprised 87K (source) words, training took 3 minutes and tuning 83 minutes (using 1GB RAM and an Intel Xeon CPU E5430 @ 2.66GHz). The largest TM uploaded comprised 2.45M words, and here training took 29 minutes, with tuning taking a further 98 minutes. Users including Dell, Epson and Caterpillar find these to be very acceptable build times, but we are experimenting with limiting the number of MERT iterations to achieve even faster builds, with little trade-off in translation accuracy.

Once a user has had their engine built (cf. Figure 3, where the user has built several MT engines), s/he can then start/stop the engine on-demand. When they have a running engine, they can send a simple text string to the engine's URL to obtain an instant translation. Alternatively they

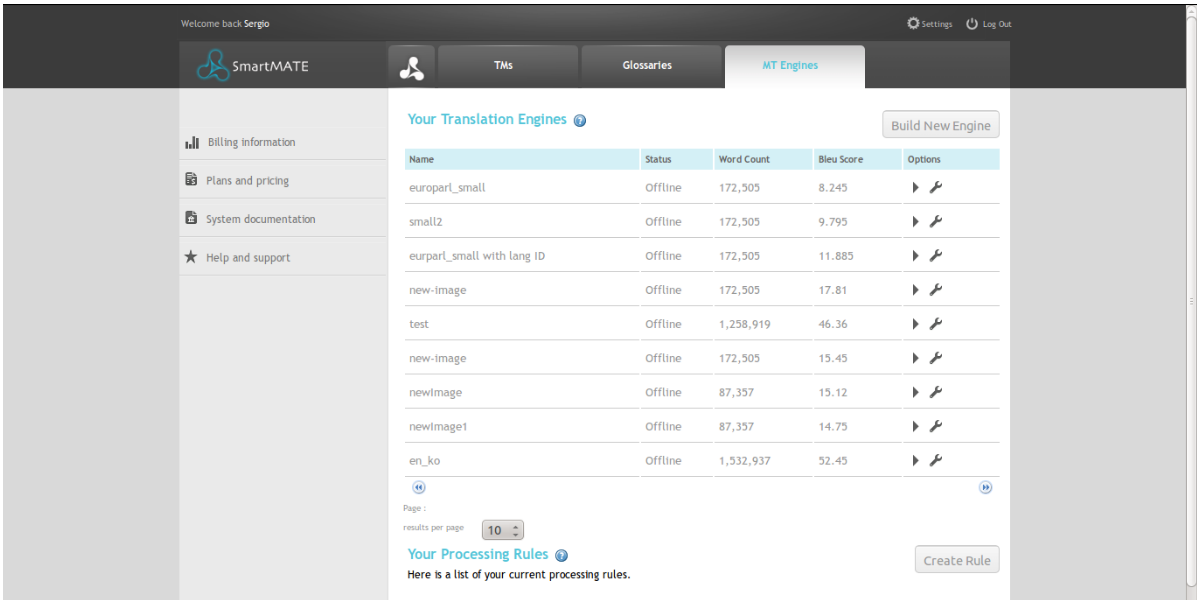

Figure 3: The SmartMATE Translation Engines Environment

can upload other documents for translation. The engine can translate .tmx and XLIFF[9] files. If the user uploads a file in another format, then the file is processed by the Translation Management System (TMS) to create a translation kit (which contains an XLIFF file), which can then be passed to the engine for translation. When a user uploads a file, they can configure a TM (or memories) to use before the SMT engine for matches which fall below the stipulated fuzzy-match level.[10]

The system also allows users to add pre- and post-processing rules of input strings to/from the engine in the form of regular expressions, which can be ordered if required. This permits the user to easily modify inputs and outputs, e.g. changing or removing XML tags, or introducing double-spacing between words in the output (cf. 'Create Rule' button in Figure 3).

Consider the case of an individual freelance translator, who specialises (say) in the translation of legal agreements for a medical device company from English into German. Google Translate does not give good quality translation, because of the technical terms and specialised legal language. The translator logs into the system, uploads his/her TM and specialised glossary (in .tmb (xml), .csv or tbx formats), clicks a button and the system is built, without any further intervention required on behalf of the user.

We have here the best of both worlds: the system can be used without any preparation or special software, just like Google Translate, but it is always customised to the translator's specific translation job, with noticeable and continual improvements in quality. This scenario is simply not possible today; for the vast majority of people doing translations every day, customised SMT is not an option. A useful development in this scenario is the use of rapid retraining/automatic adaptation functionality of an MT system to incorporate the changes that the user applies during post-editing. This will be especially useful in cases where, say, the translation of a technical term has been updated. Here, the post-editor will only need to supply the new translation within the translation of a larger segment for the system to adapt and start using it on its own.

Even where MT has found user acceptance, today's SMT systems are much less user-friendly than they could be. Most users don't care particularly about what's happening behind the scenes, only that the system is usable and produces useful results. Nevertheless, we hope that certain

[9] http://wiki.oasis-open.org/xliff/

[10] We are currently adding a facility to allow users to access MT engines built with generic, freely available data.

users will gain confidence by being permitted for the first time to interact with a state-of-the-art SMT system geared towards their specific requirements, such that while the details of the engine – pre-processing, word and phrase alignment, language modelling, tuning, and decoding – are all hidden, over time they will want to know more about these components so as to empower themselves still further, and customise the engine in a more fine-grained way while also understanding the automatic quality scores output by the system.

Equally importantly, by offering a cloud-based solution in SmartMATE, we eliminate the need for users of state-of-the-art MT to buy specific hardware on which huge translation and language models need to be built and stored. Access to the system is currently available for free, but soon the business model will cover the cost of accessing machines on the Amazon cloud, with an overhead to ensure that offering the service pays off for ALS.

5 Incorporating the Portal in a Translation Toolsuite

Currently, the portal contains two underlying systems: an Instance Management System (IMS), and a TMS. The IMS is a REST web service running on one of our web servers, with the following key responsibilities:

1. Receiving requests from users to start a server instance;
2. Receiving work request from users;
3. Receiving requests from users to stop a server instance;
4. Starting server instances, and linking back to a user;
5. Queuing work for server instances;
6. Giving work to started instances requesting work;
7. Tracking instances every minute to see which are online, so it can be tracked back to a user;
8. Storing engine configurations;
9. Storing files linked to engine configurations;
10. Tracking MT engines running on instances;
11. Rerouting MT web URL requests to the correct instance;
12. Terminating server instances;
13. Terminating server instances when an engine build is complete.

This ensures that our use of server time on the Amazon cloud is optimal, thus keeping our costs to a minimum. It also means that users are continually kept up-to-date with the status of their specific engine builds. The TMS essentially imports and cleans TMX files specified by the user, prior to using this cleaned text to optimally configure training, dev and test sets for the job at hand.

However, our SmartMATE self-serve MT portal is a component of an overall translation and editing environment that can be used at different stages of the translation process. The editing environment is easy to use, simple and should increase the productivity of whoever is using the system – author, translator, post-editor and/or proofreader – so contains various toolsets to help improve (i) the quality of the work being done and (ii) the user's productivity, while at the same time allowing for the introduction of other tools in future iterations.

One of the key assets that we have built is the ability for multiple users to work on a document simultaneously. Currently, the translation cycle is linear; after a job has been authored and committed, a translator receives the set of files pertaining to the project, and as in Fig. 1, pre-processes the documents prior to translation using a TM system. Once the translation job is complete, the files are inspected for conformity to the demands of the client (adherence to glossaries, style guides etc.), and the translation quality is underwritten. Clearly, any sub-standard translations are returned to the translator, and if time permits the loop continues.

However, what is clear from this description is that neither the translator nor the proofreader work on the same document at the same time. In Figure 4, the SmartMATE editing environment allows (i) multiple translators to work on the same

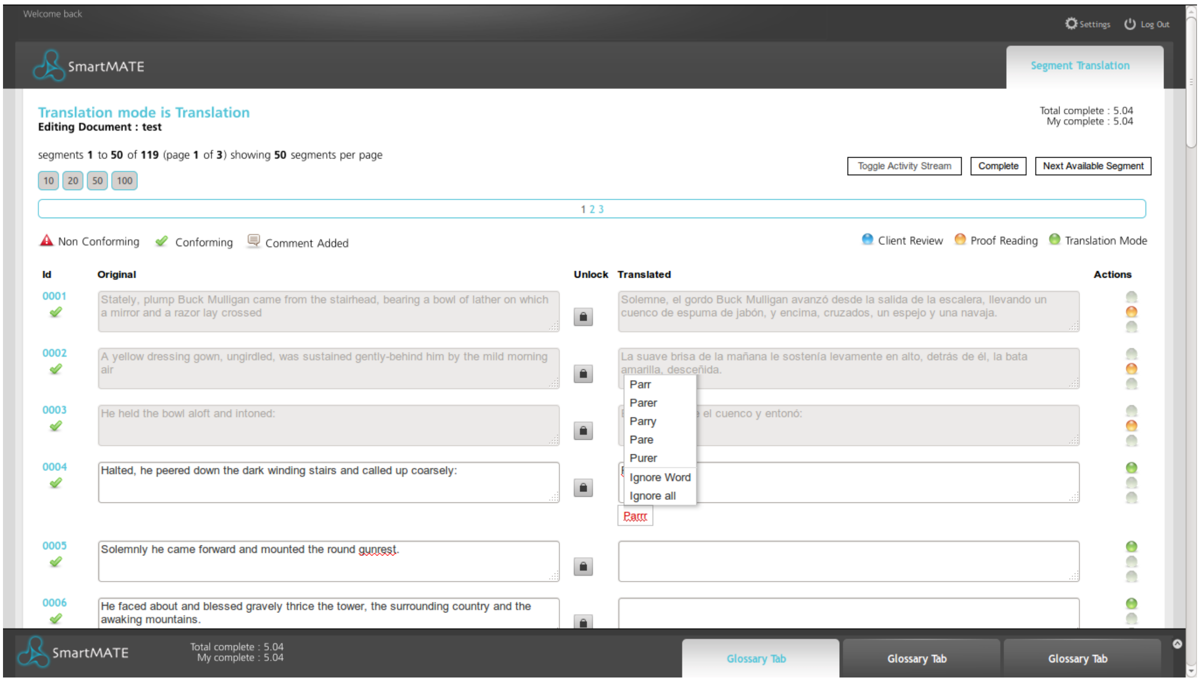

Figure 4: The SmartMATE Editing Environment

document at the same time; and (ii) translator(s) and proofreader(s) to work simultaneously on the same document. While it is clearly preferable for one individual translator to work on a particular job to ensure consistency of style, there are numerous occasions where for reasons of speed, this is not possible. In the first scenario, therefore, as soon as a translator has clicked on a particular source segment (English in Fig. 4), that segment is locked out, and no other translator can take that segment.[11] As soon as s/he commits the translation (Spanish, in our example), the segment is immediately available to a proofreader for approval. If, as is generally agreed, a proofreader can work about three times' faster than a translator, if we had three translators and a proofreader working simultaneously on a document, they should all finish at roughly the same time, with huge savings in overall translation time. In order to help us measure the throughput of translators, the system provides a clear productivity tracking capability so that it is easy to measure the amount of time a linguist is working on a job (cf. Specia (2011), who observed that speed is by far the most informative indicator of a translator's post-editing capability), as it is our intention in the near future to pay post-editors by the hour as opposed to the number of words translated.

In order to facilitate all this, the key features of the system include:

- supporting different usage scenarios or editing types as outlined above;
- recording real-time progress of translation, proofreading and reviewing for use by project managers as well as the client themselves, so that the overall completion percentage of a document is immediately visible;
- recording the user's usage of TM matching capability, i.e. exact, fuzzy, repetition, number of new words, MT edits etc.
- a facility for translators to easily send queries to the project manager and client, moderated by the project manager in the latter scenario, as well as being able to discuss translation options with fellow translators (cf. Karamakis et al., (2011));

[11] That said, if that translator does not start to work on the translation of that segment within 2 minutes, the segment is released for others to select.

- allowing multiple, simultaneous use (cf. Google docs), with visual indication (colour-coding) – of different types, depending on whether an edit, proof or review is being conducted – of locked segments.

From an implementational point of view, underpinning this functionality we use a very fast Ajax UI (with full Unicode support) for live online use, and employ Ajax to save changes when a user moves away from the input box. The configuration of the current job will determine which columns are visible together with their functionality, and which columns the user can edit. Editing shortcuts are built in to ensure optimal performance by all users, and concordance search for terminology and sub-segment matching is facilitated, as well as the ability to filter from MT/TM depending on performance thresholds. For proofreaders, LISA QA scoring[12] is built-in to allow rating of linguists, while clients can review proofreader and/or translator performance.

6 Conclusions and Further Work

In this paper, we have described SmartMATE, a portal which facilitates access to state-of-the art statistical MT with no expert MT or computational knowledge required. Currently, widespread access to such technology has been restricted mostly to large academic or industrial groups, such that translators – the largest potential audience for SMT – have to all intents and purposes been debarred from such systems. Our solution removes the primary barrier for adoption of SMT – personalisation – opening the way for adoption of SMT on a much wider scale than heretofore. By making this available via a web interface and an API for free (currently, but eventually at limited cost and time to build engines), this would be something that would become a real tool especially for the freelance translator community, which is currently largely excluded from availing of the benefits of state-of-the-art MT technology.

Furthermore, the fact that we can develop MT engines customised specifically to a user's requirements means that individual citizens can avail of state-of-the-art translation for their own personalised needs. An increasing number of multinational companies and SMEs are enabling users to personalise how to satisfy their information needs; rather than a 'one-size-fits-all' website which is managed internally within a particular company to support user Help and Doc, more and more users are being empowered to provide and maintain their own content. Again, although some users do manage to bridge language boundaries, the vast majority operate solely within their mother tongue. This is clearly a missed opportunity for both the company and their proliferation of linguistic user communities, as the collaborative spirit of online interaction encourages discussion and exchange of viewpoints, but the linguistic boundaries currently prevent people from different cultural backgrounds from being able to interact with each other.

As far as MT system-building is concerned, the days of one-size-fits-all are gone, so flexible, personalised solutions have to be geared towards specific user needs. Budgets are being cut everywhere, so technology is the only answer if translation capability for new language pairs and domains can be deployed rapidly. It is not a question of if, but rather when large multinational institutions and SMEs fit MT into their workflows. Accordingly, we are confident that all areas of business will benefit from our novel technological solution.

[12] http://producthelp.sdl.com/SDL_TMS_2011/en/Creating_and_Maintaining_Organizations/Managing_QA_Models/LISA_QA_Model.htm

Acknowledgements

Thanks to Jie Jiang, Sergio Penkale and David Boyle for their help in building key components in SmartMATE, and to Sergio for preparing the figures. Thanks also to the anonymous reviewers for their insightful comments which served to improve this paper.

References

Brown, P., J. Cocke, S. Della Pietra, V. Della Pietra, F. Jelinek, R. Mercer, & P. Roossin. 1988a. A statistical approach to French/English translation. *Second International Conference on Theoretical and Methodological Issues in Machine Translation* *of Natural Languages*, Pittsburgh, Pennsylvania; 16pp.

Brown, P.F., J. Cocke, S.A. Della Pietra, V. J. Della Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer, & P. S. Roossin. 1990. A statistical approach to machine translation. *Computational Linguistics* **16**(2):7–85.

Brown, P.F., S.A. Della Pietra, V.J. Della Pietra, & R.L. Mercer. 1993.The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics* **19**(2):26–311.

Carl, M. & A. Way (eds). 2003. *Recent Advances in Example-based Machine Translation*. Dordrecht/Boston/London: Kluwer Academic Publishers, 2003.

DePalma, D.A. & R.J. Kuhns. 2006. *Automated Translation Technology*. Tech. Report, Common Sense Advisory, Lowell, MA. `http://www.commonsenseadvisory.com/AbstractView.aspx?ArticleID=955`

Dugast, L., J. Senellart & P. Koehn. 2007. Statistical post-editing on SYSTRAN's rule-based translation system. *ACL 2007: Proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic; pp.220–223.

Genzel, D., J. Uszkoreit & F. Och. 2010. "Poetic" statistical machine translation: rhyme and meter. *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Massachusetts, USA; pp.158–166.

Hearne, M. & A. Way. 2011. Statistical Machine Translation: A Guide for Linguists and Translators. *Language and Linguistics Compass* **5**:205–226.

Hutchins, W.J. & H.L. Somers. 1992. An Introduction to Machine Translation. London: Academic Press.

Ittycheriah, A. & S. Roukos. 2007 Direct translation model 2. *NAACL-HLT-2007 Human Language Technology: the Conference of the North American Chapter of the Association for Computational Linguistics*, Rochester, NY; pp.57–64.

Karamakis, N., S. Luz & G. Doherty. 2011. Translation practice in the workplace: contextual analysis and implications for machine translation. *Machine Translation* **25**(1):35–52.

Koehn, P., H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin & E. Herbst. 2007. Moses: open source toolkit for statistical machine translation. *ACL 2007: Proceedings of Demo and Poster Sessions*, Prague, Czech Republic; pp.177–180.

Och, F-J. 2003. Minimum error rate training in statistical machine translation. *ACL-2003: 41st Annual Meeting of the Association for Computational Linguistics*, Sapporo, Japan; pp.160–167.

Och, F-J & H. Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. *ACL-2002: 40th Annual meeting of the Association for Computational Linguistics*, Philadelphia, PA; pp.295–302.

Papineni, K., S. Roukos, T. Ward & W-J. Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. *ACL-2002: 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, PA; pp.311–318.

Pecina, P., A. Toral, A. Way, V. Papavassiliou, P. Prokopidis & M. Giagkou. 2011. Towards using web-crawled data for domain adaptation in statistical machine translation. *EAMT 2011: Proceedings of the 15th Conference of the European Association for Machine Translation*, Leuven, Belgium; pp.297–304.

Sikes, R. 2007. Fuzzy matching in theory and practice. Multilingual **18**(6):39-43.

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

Specia, L. 2011. Exploiting objective annotations for measuring translation post-editing effort. *EAMT 2011: Proceedings of the 15th Conference of the European Association for Machine Translation*, Leuven, Belgium; pp.73–80.

Way, A. & M. Hearne, 2011. On the Role of Translations in State-of-the-Art Statistical Machine Translation. *Language and Linguistics Compass* **5**:227–248.