

Towards desktop-based CAT tool instrumentation

John Moran
Christian Saam
Dave Lewis

moranj3@cs.tcd.ie
saamc@cs.tcd.ie
dave.lewis@cs.tcd.ie

Department of Computer Science and Statistics, Trinity College Dublin, Ireland

Abstract

Though a number of web-based CAT tools have emerged over recent years, to date the most common form of CAT tool used by translators remains the desktop-based CAT tool. However, currently none of the most commonly used desktop-based CAT tools provide a means of measuring translation speed at a segment level. This metric is important, as previous work on MT productivity testing has shown that edit distance can be a misleading measure of MT post-editing effort. In this paper we present iOmegaT, an instrumented version of a popular desktop-based open-source CAT tool called OmegaT. We survey a number of similar applications and outline some of the weaknesses of web-based CAT tools for experienced professional translators. On the basis of a two productivity test carried out using iOmegaT we show why it is important to be able to identify fast good post-editors to maximize MT utility and how this is problematic using only edit-distance measures. Finally, we argue how and why instrumentation could be added to more commonly used desktop-based CAT tools that are paid for by freelance translators if their privacy is respected.

1. Introduction

To measure the utility of machine translation (MT) in a post-editing context two dimensions must be measured - translation quality and speed. In large commercial post-editing (PE) projects weighted scorecards methods for quality control like the LISA QA Model and SAE J2450¹ can be adapted to assess the quality of post-edited machine translation (MT) without changing the underlying software (often spreadsheets). In particular to renegotiate word prices for post-editing it is important to know how fast a translator works in an MT post-editing context and when translating from scratch as human translation (HT). Unfortunately, measuring speed presents a challenge. It is possible to measure and validate translation and post-editing speed for in-house translators where conditions can be controlled. However, 74% of translators in Europe are freelance translators (Pym, Grin, Sfreddo, & Chan, 2013) and it is likely the percentage is high in other territories also.

The majority of these translators use sophisticated desktop-based Computer-Aided Translation (CAT) tools they have purchased themselves like Trados², MemoQ³ and Wordfast⁴. However, none of these CAT tools record translation speed (Aziz et al., 2012). As a result, a common practice in translation agencies is to use edit distance between raw and post-

¹ <http://www.sae.org/standardsdev/j2450p1.htm>

² www.sdl.com

³ www.kilgray.com

⁴ www.wordfast.net

edited MT^{5,6} to infer the utility of MT. In a paper that describes how post-editing was introduced in a Spanish translation agency over a period of years Silva (2014) neatly summarises this problem:

“One of the problems faced when introducing MT was that measuring its benefits on productivity on a representative number of translators over long periods of time was very difficult to achieve. Almost all translators employed by the company were freelancers (97%), mostly working remotely from home (88%)....Although online CAT tools are becoming more popular which may allow for easier productivity rate and post-editing effort tracking, most translators work on a locally installed CAT tool that does not offer such functionalities. The only information available was their feedback. As valuable as it is, it did not really provide valid post-editing effort figures. Internal translator productivity, on the other hand, was easily measured as there was complete control over their working environment and number of hours employed.”

In this paper we will survey a number of web-based and desktop-based systems that can be used to measure translation speed. We will then discuss some of the weakness of current web-based CAT tools from a translator perspective and how OmegaT⁷, a free open-source desktop-based CAT tool that is used by many translators worldwide addresses some of these weaknesses. We will then present iOmegaT, an adapted version of OmegaT to which we have added instrumentation to log User Activity Data in a format we call CAT-UAD and describe the workflows for the system. We describe how MT utility can be measured by means of Segment Level A/B (SLAB) testing, in which CAT-UAD is recorded unobtrusively, and hence cheaply, as a translator processes segments in (A) HT segments and (B) MT segments. We will illustrate how data gathered by this means can be useful for translator selection for post-editing (PE) and discuss some other uses of temporal PE data. Finally, we will discuss some of the issues that may need to be addressed if instrumentation is to be adopted by other more commonly used desktop-based CAT tools normally purchased by translators.

Related Work

In general we will categorise related work on measuring translation and post-editing speed into applications that are obtrusive to the task of translation and those that are unobtrusive. We define an obtrusive translation environment as one in which a translator is asked to work in manner that is not business-as-usual. For example, in an obtrusive environment a translator may be asked to carry out an annotation task to score MT in terms of adequacy and fluency, the environment may use a constrained method for segment navigation or it may not be possible to leverage translation memory. Though some obtrusive tasks like annotation are useful, obtrusive environments reduce the quantity of translation speed data that can be gathered in a typical commercial translation scenario due to cost considerations (Lewis & Moran, 2010).

1.1. Obtrusive applications that can measure translation speed

The first use of UAD to record how translators interact with an editing environment can be traced back to TransLog (Jakobsen & Schou, 1999). In this application and its successor TransLogII (Carl, 2012) UAD is recorded in XML for subsequent analysis and replay. TransLog is designed for lab use. It is impractical for large-scale or multi-day/multi-language MT

⁵ <http://www.yamagata-europe.com/en-gb/blog/item/909/quality-metrics-for-machine-translation-output>

⁶ http://www.alphacrc.com/news/newsitem_29-11.php

⁷ www.omegat.org

productivity testing. This lab or experimental focus makes it particularly suitable for use with eye-tracking software, which is used to examine cognitive aspects of the translation process. Another desktop-based free and open-source application used to carry out translation research is PET (Aziz et al., 2012). The focus here is on MT post-editing. Translators can annotate MT proposals, e.g. in terms of quality using Likert scores. It also records translation speed and reports on edit distance. Another, more standards-based free open-source system that can be used to annotate post-edited MT is Ocelot⁸ and two web-based MT rating systems that can also report on PE time are TransCenter (Denkowski & Lavie, 2012) and ACCEPT⁹.

A number of web-applications to measure post-editing speed in a SLAB testing scenario have been described in the post-editing literature. CrossLang is a company that provides MT consulting and software services. They have developed a web application that can be used by their clients to measure post-editing speed and compare it to HT speed (Depraetere, De Sutter, & Tezcan, 2014). Autodesk also describe a similar internal system (Plitt & Masselot, 2010). A similar commercially available system that can be used to measure translation speed (and annotate segments) is the TAUS DQF platform¹⁰. This application is available for a monthly fee.

Features these three applications have in common are that translators can press a Pause button when they chose to take a break. In the Autodesk and TAUS DQF applications translators navigate through a text by means of a Previous and Next button. Thus, for example, if a translator is 200 sentences into a translation and decides to return to the 50th sentence to make a change based on new context, he must press the Previous button 149 times. There is no mention of self-review in publications that use this design so presumably translators carrying out productivity tests navigate the text from start to finish in a single pass. In the Crosslang application only a Next button is available so self-review is technically impossible. This kind of linked-list navigation can impact negatively on translation consistency, as the translator is not working in a business-as-usual manner in which they are free to navigate freely between segments. Also, no application listed above allows the translator to leverage translation memory. As such they do not work well in a typical translation agency workflow.

1.2. Unobtrusive applications that can measure translation speed

CASMACAT¹¹ and MateCAT¹² are two cooperating web-based CAT tool projects still in progress. While CASMACAT provides a more constrained working environment and is focused on research use, MateCAT is designed to be used by working translators, albeit “*volunteer translators and professional translators without advanced technical skills*” (Alabau et al., 2013).

Finally, IBM TM/2 is an offline CAT tool used internally in IBM¹³. A number of years ago a feature called MTLog was added to the tool to measure translation speed at a segment level. It is both desktop-based and also has a logging feature that has been added to an existing CAT tool. While results from this system have been presented orally¹⁴ no published re-

⁸ <http://open.vistatec.com/ocelot>

⁹ <http://www.accept.unige.ch>

¹⁰ <https://www.taus.net>

¹¹ www.casmacat.eu

¹² www.matecat.com

¹³ www.ibm.com

¹⁴ www.localizationworld.com/lwbar2011/presentations/files/E6.ppt

search is available on the topic. To the best of our knowledge, the system is only available for use internally within IBM and to its suppliers. Unfortunately, the MTLog feature is not available in the open-source version of TM/2 called Open TM2¹⁵. Table 1 presents a summary of this survey.

Name	Obtrusive	Architecture	Internal to company
Translog I&II	Yes	Desktop	No
PET	Yes	Desktop	No
TransCenter	Yes	Web	No
Autodesk tool	Yes	Web	Yes
Crosslang	Yes	Web	No
TAUS DQF	Yes	Web	No
IBM TM/2+MTLog	No	Desktop	Yes
CasmaCAT/ Mate-CAT	No	Web	No
ACCEPT	Yes	Web	No
iOmegaT Workbench	No	Desktop	No

Table 1. Survey of applications that can measure translation speed

As iOmegaT is a logging feature added to an existing desktop-based CAT tool functionally it is most like IBM TM/2 + MTLog. However, this system is not available outside of IBM. In terms of goals and functionality MateCAT is currently the most similar *available* application to iOmegaT. For this reason we will now examine some of the differences between web-based and desktop-based CAT tools as they appeal to quite different translator demographics.

2. Browser-based CAT tools – a discussion

In this section of the paper we take a step back from MT post-editing productivity testing and examine the purported trend towards web-based CAT tools in the previous quote by Silva. In a paper entitled *Power-shifts in web-based translation memory* Garcia (2008) notes that while online or web-based workflow systems may serve other stakeholders in the translation supply chain, they can involve drawbacks for freelancer translators. Amongst other issues he argues that web-based translation memory tools do not allow translators to access their own private locally stored translation memories and terminology databases. Access to locally stored terminology files does not just improve translation quality by ensuring consistency and reusing previous terminology research. Much anecdotal evidence suggests auto-complete using well-populated termbases can also aid translation speed.

Though this paper was published several years ago, we are aware of no browser-based CAT tool that facilitates access to local files. This is unfortunate as the Google Chrome browser allows access to local files¹⁶ - so from a programming perspective it should be possible. Access to the local file system by the web-application seems necessary as we feel at least some translators would be unwilling to upload private termbases¹⁷ to any third party. Aside

¹⁵ www.opentm2.org

¹⁶ <https://developer.chrome.com/apps/fileSystem>

¹⁷ Private translation memory is less of a concern as a translator can always have a second desktop-based CAT tool open for concordance searches and full-sentence matches across clients are rare.

from the competitive advantage private termbases represent in terms of quality and translation speed, the contents of these databases may originate from various clients. Uploading them to any third party could be a breach of client confidentiality as defined in non-disclosure agreements commonly signed by translators.

Garcia also points to the fact that most browser-based systems do not let the translator retain bilingual assets generated while translating in the CAT tool. This means previous work cannot be referenced. In his summary he refers to web-based CAT tools as a “*looming reality*” that will “*alienate the most able section of its workforce and discourage promising newcomers from entering*”.

However, Garcia’s pessimism may have been premature. According to recent surveys of CAT tools used by professional translators^{18,19}, by far the dominant architecture remains the desktop-based CAT tool. The 8 most used CAT tools in Figure 1 are desktop-based CAT tools. 3000 translators responded to the survey on proz.com, a translator website. Only 4 of the 16 systems listed feature a browser-based CAT tool. These are XTM²⁰, MemSource²¹, Wordbee²² and Google Translator’s Toolkit²³. Interestingly, though they started out as applications that include a web-based CAT tool, XTM and MemSource now provide their own desktop-based CAT tools. Only Google Translators Toolkit, a free online service provided by Google, does not provide an export facility for linguistic assets (XLIFF/TMX/TBX files) or TIPP²⁴ export facility. Unfortunately, this export means the translator can no longer collaborate using a shared translation memory in parallel with other translators so the benefits of being online are lost.

A concern not addressed by Garcia is that translators who are used to working with a CAT tool for many clients may suffer a loss of productivity if they are asked to work in a new unfamiliar environment intermittently.

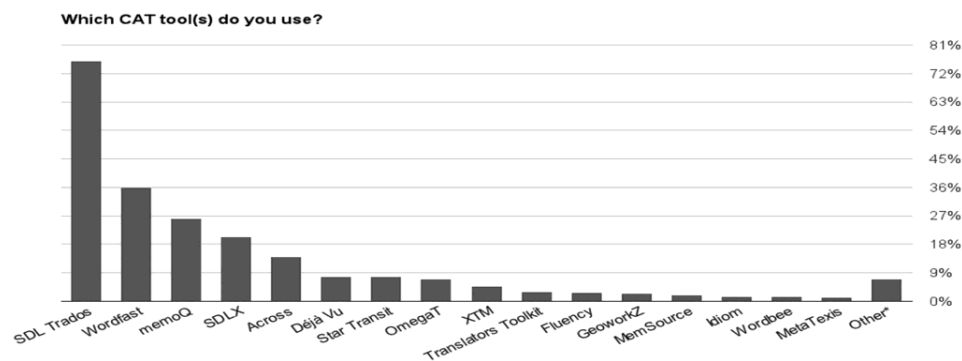


Figure 1. Results of a survey on CAT tool use (reproduced with permission).

¹⁸ <http://www.translationtribulations.com/2014/01/the-2013-translation-environment-tools.html>

¹⁹ <http://prozcomblog.com/2013/03/28/cat-tool-use-by-translators-what-are-they-using>

²⁰ <http://xtm-intl.com>

²¹ <http://www.memsource.com>

²² <http://www.wordbee.com>

²³ <https://translate.google.com/toolkit>

²⁴ <https://code.google.com/p/interoperability-now>

Finally, a shortcoming of the web-based tools surveyed above, including MateCAT is that translators cannot work if the connection to the Internet is lost. iOmegaT was developed in close collaboration with a large translation agency called Welocalize²⁵. Feedback from translators working for the agency identified this as a frustration with existing web-based CAT tools provided by other agencies.

In short, web-based CAT tools provided for free by agencies or translation buyers and used intermittently by translators can be costly to those translators in terms of lost productivity or translation quality.

2.1. OmegaT

OmegaT addresses most of these concerns. It is a well-featured desktop-based CAT tool already used by many experienced professional translators. It can be used for offline and collaboratively using the team function and translators can use their own translation memories and terminology databases safe in the knowledge that the contents will not be uploaded to a third party. A feature of OmegaT's GPLv3 license is that any user of the software has the right to the source code for the software. This means assurance can be sought in this regard by reading the code or consulting with someone else who can. This is not possible for proprietary software where oftentimes complex legal agreements must be taken on face value. However, integration of instrumentation does not solve the problem of productivity loss due to a lack of familiarity for users of CAT tools. We will return to this problem in Section 5.

3. The iOmegaT Translator Productivity Workbench

The iOmegaT Translator Productivity Workbench is a fully-offline suite of software applications used to record and analyze how translators interact with a CAT tool, in particular when post-editing MT. Unlike web-based environments its offline nature means that test results cannot be affected by local network conditions or high server load. Static logging has no impact on the performance of OmegaT. The suite of applications is composed of four main components shown in Figure 2. These are:

- 1) The instrumented CAT tool (iOmegaT)
- 2) Middleware, used to transfer files to and from SDL TMS and SDL WorldServer
- 3) A collection of utilities to prepare a productivity test.
- 4) The analysis application and prototype replayer component.

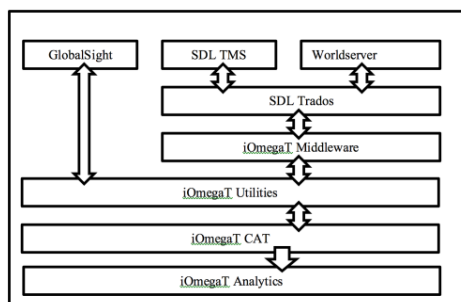


Figure 2. The iOmegaT Translator Productivity Workbench workflow

²⁵ www.welocalize.com

The software is designed to work in a typical large-scale or enterprise post-editing project workflow where there is some degree of content homogeneity within work packages that are handed off to translators or across multiple work packages on a single account. The workflow is outlined in Figure 2. In a typical scenario an engineer responsible for MT or a project manager selects a project from a TMS to use for productivity testing. If the TMS is GlobalSight only one step is required to prepare the project for productivity testing, namely deleting segments at random from the TMX file used to store the machine translation. This creates the HT segments for the HT/MT SLAB tests. If the files originate from an SDL TMS or SDL WorldServer system we use the Middleware applications to import the various dialects of XLIFF into OmegaT via Trados (sdlxliff) before deleting the MT segments using the iOmegaT Utilities. Initially, MT productivity tests were one way. Files could not be returned to the TMS they came from so they could not be used. This massively reduces the cost of productivity tests as prior to this translations were discarded. We expect the cost of acquiring PE speed data to fall further where OmegaT is used as a production CAT tool. This means post-editing productivity can be measured on an ongoing basis rather than in a shunted test workflow.

The code we have added to OmegaT writes a stream of events corresponding to user activity and application context to XML log files stored on a translators PC as the translator works in the CAT tool. We call these XML log files instrumentation files to distinguish them from application log files. As opposed to application log files, instrumentation files are repayable in the Replayer. Currently, this component is an early prototype. It can only reply CAT-UAD at a segment level. Each time a source file is opened a new instrumentation file is created. CAT-UAD is mainly expressed as events. These events contain timestamps in milliseconds and contextual information. For example, in the case of a keystroke event the time of the event is recorded along with the key in question and the cursor position in the segment. Segment editing sessions are used to distinguish between the first and subsequent times a translator visits a particular segment. Figure 3 shows an example of a segment in which an MT segment is post-editing in 42 seconds. The project name, file name, sourceIndex and translatorID combine to uniquely reference a Source segment and this segment may be accessed or visited many times. In productivity tests we find on average translators visit each segment between two and three times.

```

<SegmentSession sourceIndex="14">
  <SourceText>This section describes the options you can use with the <code>cmdjob</code> command.</SourceText>
  <PreeditText></PreeditText>
  <Events>
    <LogEvent Action="segmentOpen" Time="1363874598949"/>
    <LogEvent Action="mtMatchPlacement" In diesem Abschnitt werden die Optionen beschrieben, können Sie mit dem Befehl cmdjob."
      MTSYSTEM="123456789" Time="1363874599029"/>
    <LogEvent Action="segmentClose" Time="1363874641632"/>
  </Events>
  <PostEditTarget>In diesem Abschnitt werden die Optionen beschrieben, die Sie mit dem Befehl cmdjob verwenden können.</PostEditTarget>
  <Comment postEditTime="42"/>
</SegmentSession>

```

Figure 3. A shortened example of a 42 second segment editing session

Once the translation is ready the translator zips and return that package by e-mail or FTP. The by the MT engineer or project manager runs the analysis software across the instrumentation files for all returned packages in a single batch operation. This populates a database that contains data on each editing session. On the basis of that database a summary of results for the current test is output as a spreadsheet. The report contains information on the number of words in each segment and the number of words that were discarded due to inactivity on the translator's part. Some translations are also discarded for other reasons, e.g. if a translator saves an empty target segment to the project translation memory. Cut-offs due to inactivity are defined in the analytics configuration file. For example, a cut-off of 5 minutes was used

for the productivity tests discussed below. For patent translation with much longer sentences 15 minutes is used. At an early stage in the design we decided against a Pause button, as it is not possible to verify if it was used and it conflicts with the unobtrusive philosophy underlying the design of the software.

4. Some results from two large scale productivity tests

Economically, from a translator’s perspective temporal post-editing effort is the more important measure of effort (Krings, 2001). In this section we will discuss the results of two MT productivity tests carried out using iOmegaT, with a focus on temporal data. We will show that reliance on edit-distance alone is not advisable when measuring the utility of MT or selecting translators to work on post-editing projects.

	Dell	Autodesk
Total Number of Segments	14686	13145
HT	4920	1672
100% match	412	20
Already translated	2855	2249
Fuzzy Match	2854	2070
MT Changed Segments	3353	5703
MT Unchanged Segments	292	1431
MT Changed WPH	476	560
MT Unchanged WPH	942	837
MT Utility	+5%	+54% ²⁶

Table 2. Summary of data from two similar productivity tests

Table 2 shows a summary of statistics from two similar productivity tests carried out using iOmegaT for Autodesk and Dell, two companies that already use MT on a large scale. Translators were all experienced professional translators and MT had been in use for some time on the accounts in question so translators had passed many successive QA cycles. Spot-checking was applied to ensure basic quality levels were maintained in the productivity tests. In general segments were sentences. In both projects the mean sentence length was nine words and the mean number of visits per segment was just under two. Whether a segment was MT or not had no meaningful impact on the mean visit count.

100% match segments were segments in the translation memory provided to the translators before the project was started. “Already translated” segments are those the translator has translated or post-edited during the project and which appear again as full matches. “Fuzzy Match” segments were fuzzy or partial matches from the translation memory provided to the translator or the project translation memory that is populated as the translator works. “MT Changed” segments are segments that are changed by the translator and “MT Unchanged” segments were judged to require no changes. “MT Utility” is the overall utility of the MT in terms of speed relative to HT translation speed. Unsurprisingly a high MT utility corresponds to a high number of unchanged MT segments as unchanged MT segments required much less time to process. This can be seen in the Words per Hour (WPH) values for each category. All WPH values also account for self-review time across multiple segment visits or sessions. This is discussed in more Moran, Lewis, & Saam (2014).

²⁶ French, Italian, German, Spanish only

A feature of these two early productivity tests was that the two-day translation projects were handed off to two translators per language to be redundantly translated. However, they were free to translate normally using OmegaT, so although all translators started at the start of the job there was no stipulation that they should translate the same segments and some translators translated non-contiguously. Also, due to multithreading in OmegaT a segment may be presented to translator A as MT while the same segment may be shown to translator B as a translation memory match. This was recorded correctly by the instrumentation but was only apparent upon analysis. For some visualisations of segment progression and segment category distribution see Moran et al. (2014). In general faster translators translated more segments over the two days. Figure 3 shows PE speed compared against translators for both tests. There was a low positive correlation between translators in terms of PE speed. The Pearson correlation coefficient comparing speed between Translator A and B for Dell is $r = .19$. For Autodesk it is slightly higher, $r = .29$. It appears the translators post-edited at quite different speeds even when initially presented with the same MT and source text.

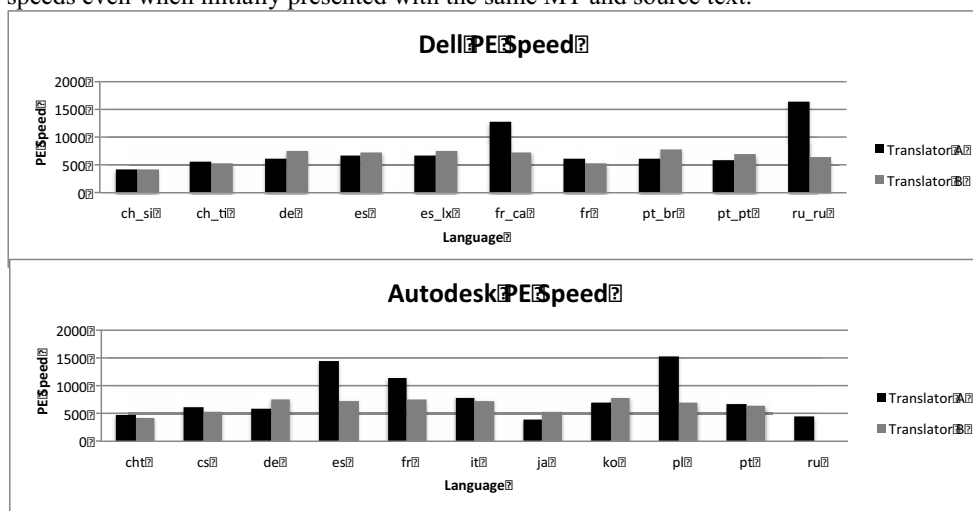


Figure 4 Autodesk and Dell PE speed per language for the same files.

The scatterplot in Figure 5 below shows the correlation between PE similarities as calculated using a character-based Levenshtein algorithm averaged across MT segments for each translator and PE speed in WPH in which self-review time was accounted for.

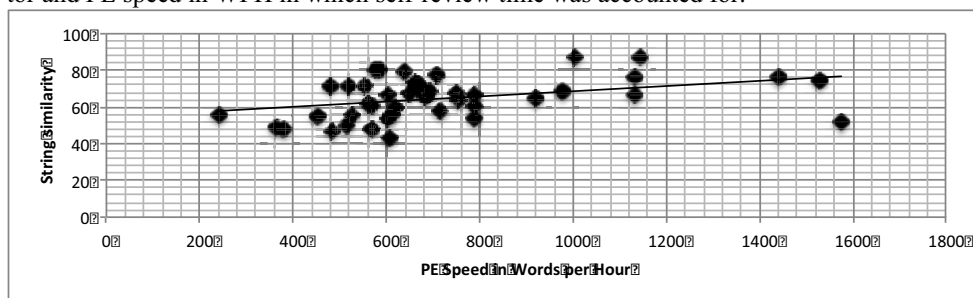


Figure 5. PE speed versus string similarity

44 translators across the Autodesk and Dell tests were bundled together in the analysis. A moderate correlation of Pearson $r = 0.37$ was found. Thus far we have only discussed post-

editing speed. However, SLAB testing accounts for multiple segment categories so we turn our attention to HT.

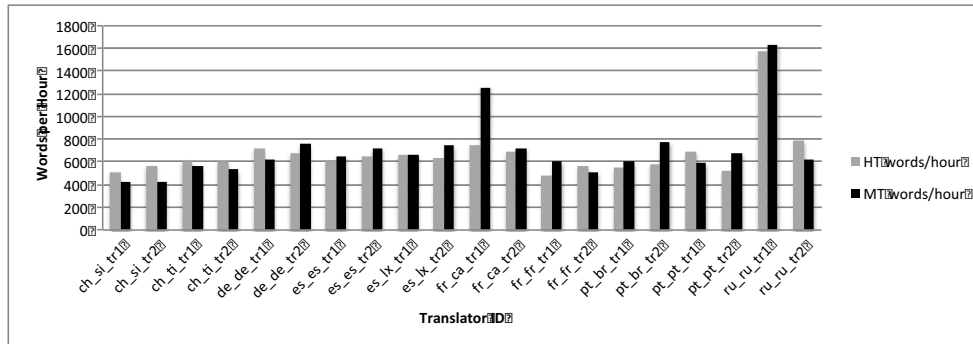


Figure 6. HT/MT speed ratios and absolute speed values for a Dell productivity test in 2012.

Figure 6 shows absolute HT and MT post-editing speeds for the Dell productivity test. As Autodesk had low HT sample sizes for languages except for French, Italian, German, Spanish (FIGS) it is not shown here. It should be noted that the +5% MT speed delta was positive mainly as a result of four efficient post-editors, fr_fr_tr1, pt_br_tr2 and pt_pt_tr2 and one very efficient post-editor fr_ca_tr1. Two years after this snapshot was taken, the current HT/MT speed ratio on the Dell account is around 40%. This improvement is due to improved MT but also translator selection weighted in favour of translators like these four translators and against their language counterparts. However, using string distance measures alone it is not always possible to identify efficient post-editors.

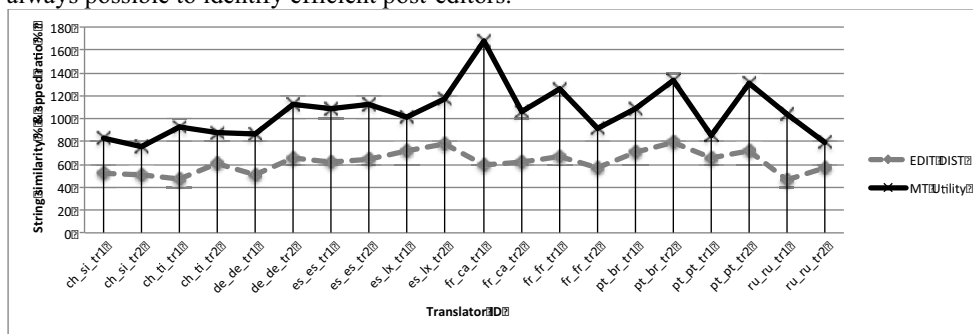


Figure 7. String similarity and MT utility (HT/MT speed ratio)

This can be seen in Figure 7 where the average string similarity across MT segments using character-based Levenshtein is shown per translator along with the HT/MT ratio. Both are represented as a line graph. For ease of visual comparison both string distance percentages (up to 80%) and PE speed ratio percentage (80% to 160%) are both shown on the X-axis. The three or four efficient posteditors are clearly seen as peaks above the 100% line, the breakeven point for MT productivity. However, they are barely visible on the lower dotted line. In several instances edit distance is misleading as a predictor for PE speed relative to HT. For example, fr_ca_tr1 and fr_ca_tr2 have about the same edit distance value of 60% but very different PE speeds relative to HT. fr_ca_tr1 is twice as fast using MT relative to HT com-

pared to fr_ca_tr2. This is consistent with key findings from a similar large scale productivity test carried out by Plitt & Masselot (2010). They found the translator who post-edited fastest had the best quality review scores and also made the most changes. It seems some translators are more efficient than others at using MT to improve their working speed. More information on the MT system used by Autodesk can be found in Zhechev (2012) and a discussion of trends across multiple productivity tests at Welocalize is presented in by Casanellas & Marg (2014).

We conclude this discussion with a note on sample sizes. Though work remains to be done here, we have found that analyzing CAT-UAD over periods shorter than a day per main category (HT and MT) leads to unreliable statistics due to problems associated with small sample sizes and hence sparse data in each segment category. For this reason, it is important that productivity test costs are kept to a minimum. Once reliable speed data has been established, edit distance metrics can be used to monitor MT quality. However, a better solution would be to be able to measure post-editing speed at all times or at least intermittently in a translator's preferred desktop-based CAT tool.

5. Towards a standard for CAT-UAD in paid CAT tools

Though we are unaware of any research on the topic intuitively, translators are most familiar and hence very likely most productive in the one or two CAT tools they use most frequently. For the translators who responded to the questionnaire in Figure 1, in most cases this is Trados, Wordfast, MemoQ or DejaVu. OmegaT is only a standard CAT tool for a minority of translators.

However, HT/MT SLAB testing is useful to translators. In our experience the discounts for post-editing it facilitates are generally considered fair. A portion of the cost saved because of the time saved by MT can be passed on to the client. In turn this saving can be used to improve MT. As long as translator hourly earnings for MT post-editing are above hourly earnings for HT, payment for post-editing should not be a complaint. However, there are other reasons why temporal PE data is useful. Along with translator feedback, it also makes it possible to assign priority to problems reported by translators that impact most on working speed, e.g. over translation of URLs.

However, from a translator's perspective there is a darker side to measuring translation speed. It could be used to negotiate discounts where MT is not in play. For example, a translator who translates at 2,000 words per day when he is new to a regular translation account might translate at 3,000 words per day after a few months. Although they are paid by the word translators typically invest more time in terminology research when they are new to an account. Were a client to approach the translator with speed data to ask for a discount, at this point the translator is likely to become unhappy that translation speed data was being recorded as this time investment would be wasted.

In web-based CAT tools translation speed data can be recorded on a server and a translator cannot delete or view that data. In iOmegaT it is recorded on a local disk and can be deleted from the /instrumentation folder within the OmegaT project folder. In this respect desktop-based CAT tools provide a level of privacy web-based tools do not. However, a translator does not have to make a buying decision to use iOmegaT. Agencies are within their rights to ask freelance translators to use specific tools if they provide the tool at no cost. Equally, freelancer translators are within their rights to turn down the work.

The situation is different for paid desktop-based CAT tools like Trados, MemoQ or DejaVu. Freelance translators typically pay for this software. Were these applications to implement instrumentation of the kind used in iOmegaT, in our opinion it is important that speed

data is only shared at the discretion of the translator so it can be limited to post-editing or other scenarios in which technology can improve a translator's per hour earnings. Otherwise, it seems likely translators will pay for CAT tools that do not automatically share speed data. We do not see SLAB tests putting an end to per word pricing. However, we do feel that where an evolving technology like MT is concerned it is important for buyers who supply that MT and translators who use it to be able to monitor relative per hour earnings.

Finally, CAT-UAD may be useful beyond MT. It could be used by translators to communicate delays caused by slow servers (terminology and translation memory), time wasted fixing false positives generated by automatic translation quality checks or to measure the impact of technologies like intelligent fragment assembly, Automatic Speech Recognition (ASR) or predictive typing /auto-complete. It might also encourage training in the use of these technologies by measuring the impact of that training. As desktop-based and some web-based CAT tools become more complex it will become increasingly important to be able to record and analyze the impact of individual technologies on translation speed. We feel that if paid CAT tool companies address translator privacy concerns many translators will see the benefit of CAT-UAD and those that do not can ignore the feature.

To this end we have offered to provide early access to the iOmegaT CAT tool to a number of commercial desktop-based CAT tool publishers with a view to encouraging a discussion on CAT-UAD as a standard data format (Moran, 2014). Discussions are progressing in that regard.

6. Summary

In this paper we surveyed a number of applications that can be used to measure translation speed. We showed on the basis of a questionnaire that the desktop-architecture remains dominant in the CAT tool market and outlined some advantages of OmegaT, a free open-source CAT tool over web-based CAT tools. We briefly introduced the iOmegaT Translator Productivity Workbench and described how it is used in a typical enterprise translation workflow. On the basis of two HT/MT SLAB tests we showed that because translators vary greatly in how much MT speeds up their work it is important to be able to spot post-editors who are faster using MT and, conversely, recognize translators who are not aided by MT proposals. Finally, we made a case for instrumentation in existing paid CAT tools while drawing attention to the importance of a translator-centric privacy model.

Acknowledgements

This research is supported by the Science Foundation Ireland (Grant 12/CE/I2267) as part of CNGL, the Centre for Global Intelligent Content (www.cngl.ie) at Trinity College Dublin and as part of Technology and an Innovation Development Award Feasibility Grant (12/TIDA/I2424) titled "iOmegaT – An Instrumented Replayable Computer-Aided-Translation Tool". The iOmegaT Workbench developed under this grant has been licensed to a small number of companies. Companies interested in licensing it or researchers interested in using it at no cost should contact John Moran at the e-mail address in the title of this paper.

References

- Alabau, V., Bonk, R., Buck, C., Carl, M., Casacuberta, F., Garcia-Martinez, M., ... others. (2013). Advanced Computer Aided Translation with a Web-Based Workbench. *Machine Translation Summit XIV*, 55.

- Aziz, W., Castilho, S., & Specia, L. (2012). PET: a Tool for Post-editing and Assessing Machine Translation. In N. C. (Conference Chair), K. Choukri, T. Declerck, M. U. Doğan, B. Maegaard, J. Mariani, ... S. Piperidis (Eds.), *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. Istanbul, Turkey: European Language Resources Association (ELRA).
- Bota, L., Schneider, C., & Way, A. (2013). COACH: Designing a new CAT Tool with Translator Interaction. In *Machine Translation Summit XIV, Main Conference Proceedings* (pp. 313–320).
- Carl, M. (2012). Translog-II: a Program for Recording User Activity Data for Empirical Reading and Writing Research. In *LREC* (pp. 4108–4112).
- Casanellas, L. & Marg, L. (2014). Assumptions, Expectations and Outliers in Post-Editing. In *EAMT 2014: Proceedings of the 17th Annual conference of the European Association for Machine Translation*.
- Denkowski, M., & Lavie, A. (2012). TransCenter: Web-Based Translation Research Suite. In *Workshop on Post-Editing Technology and Practice Demo Session*. San Diego. Retrieved from <http://www.cs.cmu.edu/~mdenkows/transcenter/>
- Depraetere, I., De Sutter, N., & Tezcan, A. (2014). Post-edited quality, post-editing behaviour and human evaluation: a case study. In S. O'Brien, M. Carl, L. Specia, & L. Balling (Eds.), *Post-editing of Machine Translation* (pp. 78–109). Newcastle upon Tyne.
- Garcia, I. (2008). Power shifts in web-based translation memory. *Machine Translation*, 21(1), 55–68. doi:10.1007/s10590-008-9033-6
- Jakobsen, A. L., & Schou, L. (1999). Translog documentation. *Copenhagen Studies in Language*, 24, 149–184.
- Krings, H. (2001). *Repairing Texts: Empirical Investigations of Machine Translation Post-Editing Processes*. The Kent State University Press.
- Lewis, D., & Moran, J. (2010). Unobtrusive methods for low-cost manual evaluation of machine translation. In *Tralogy*. Paris.
- Lossner, K. (2012). *memoQ 6 in Quick Steps*. Self published.
- Moran, J. (2014). User Activity Data in a CAT tool - An industrial use case and a proposal for an open standard - oral presentation at FEISGILTT 2014. Dublin. Retrieved from <http://www.localizationworld.com/lwdub2014/feisgiltt/accepted.html>
- Moran, J., Lewis, D., & Saam, C. (2014). Analysis of Post-editing Data: A Productivity Field Test using an Instrumented CAT Tool. In S. O'Brien, M. Carl, L. Specia, & L. Balling (Eds.), *Post-editing of Machine Translation* (pp. 126–146). Newcastle upon Tyne.

- Plitt, M., & Masselot, F. (2010). A Productivity Test of Statistical Machine Translation Post-Editing in a Typical Localisation Context. *The Prague Bulletin of Mathematical Linguistics*, (93), 7–16.
- Pym, A., Grin, F., Sfreddo, C., & Chan, A. L. J. (2013). *The status of the translation profession in the European Union*. London: Anthem.
- Silva, R. (2014). Integrating Post-Editing MT in a Professional Translation Workflow. In S. O'Brien, M. Carl, L. Specia, & L. Balling (Eds.), *Post-editing of Machine Translation* (pp. 24–51). Newcastle upon Tyne.
- Way, A., Holden, K., Ball, L., & Wheeldon, G. (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* (pp. 43–52).
- Zhechev, V. (2012). Machine Translation Infrastructure and Post-editing Performance at Autodesk. In *AMTA 2012 Workshop on Post-Editing Technology and Practice (WPTP 2012)*. San Diego.