This document is part of the Project "Machine Translation Enhanced Computer Assisted Translation (MateCat)", funded by the 7th Framework Programme of the European Commission through grant agreement no.: 287688.

**Machine Translation Enhanced**
**Computer Assisted Translation**

# D3.2 – Second Report on Informative MT

**Authors:** Matteo Negri, Marco Turchi, Philipp Koehn, Christian Buck, Ulrich Germann

**Dissemination Level:** Public
**Date:** 26 September 2014

| | |
|---|---|
| Grant agreement no. | 287 688 |
| Project acronym | MateCat |
| Project full title | Machine Translation Enhanced Computer Assisted Translation |
| Funding scheme | Collaborative project |
| Coordinator | Marcello Federico (FBK) |
| Start date, duration | November 1st 2011, 36 months |
| Dissemination level | Public |
| Contractual date of delivery | August 31st, 2014 (M34) |
| Actual date of delivery | September 26th, 2014 |
| Deliverable number | 3.2 |
| Deliverable title | Second Report on Informative MT |
| Type | Report |
| Status and version | Final, V1.0 |
| Number of pages | 48 |
| Contributing partners | UEDIN, FBK |
| WP leader | UEDIN |
| Task leader | FBK |
| Authors | Matteo Negri, Marco Turchi, Philipp Koehn, Christian Buck, Ulrich Germann |
| Reviewer | Nicola Bertoldi |
| EC project officer | Aleksandra Wesolowska |
| The partners in MateCat are: | Fondazione Bruno Kessler (FBK), Italy<br>Université Le Mans (LE MANS), France<br>The University of Edinburgh (UEDIN)<br>Translated S.r.l. (TRANSLATED) |

For copies of reports, updates on project activities and other MateCat-related information, contact:

FBK
MateCat
Marcello Federico — federico@fbk.eu
Povo - Via Sommarive 18 — Phone: +39 0461 314 521
I-38123 Trento, Italy — Fax: +39 0461 314 591

Copies of reports and other material can also be accessed via http://www.matecat.com

# Executive Summary

The purpose of Informative Machine Translation is to provide post-editors of MT output with additional information beyond simple translation suggestions in order to boost their productivity. Examples of such additional help are assistance with bilingual terminology management, and mark-up of untrustworthy parts of the translation suggestion, i.e., an indication of how sure the system is in general about the quality of its suggestion, and specific parts thereof in particular. Access to such information should ideally be integrated seamlessly and unobstrusively into the post-editor's or translator's work interface, e.g., the MATECAT workbench.

Automatic assessment of machine translation quality, especially if there is no gold standard reference translation to compare against, is an active area of research. There is still no definitive answer as to what constitutes a good translation suggestion with respect to its helpfulness to the human post-editor. Part of the research within the scope of the MATECAT project addresses this question.

This report details our continued work on terminology assistance and quality estimation of MT output with a focus on practical applicability to make these technologies available to professionals within a Computer-assisted Translation (CAT) environment. Software and datasets developed within the scope of this work package have been released as open-source resources. The published papers (11 in total during the third year of the project) are listed in the Appendix to this deliverable.

# Contents

# 1 Introduction

The goal of the MATECAT project is to develop a Computer-assisted Translation (CAT) framework that supports translators[1] in post-editing raw machine translation (MT) output. In addition to improvements to the overall quality of MT output in general, it is also possible to assist translators in their task by providing them with information beyond the raw MT output. This report summarizes the research activities within Workpackage 3 in the third and last year of the MATECAT project. It gives an update on the activities in Tasks 3.1 (Terminology Help) and 3.2 (Sentence- and Word-level Confidence Estimation), and presents the results of our efforts in integrating our findings into the MATECAT interface (Task 3.3 — Enriched MT Output).

**Terminology management** is a crucial aspect of the work of professional translators. In Section 2, we present the results of our work on supporting translators in this respect in two ways: first, by automatically acquiring bilingual terms from existing data, either the results of the post-editors' own (or their colleagues') work, or from monolingual data; and second, by using the acquired terms to improve the performance of an SMT system at the back-end of the system.

**Automatic estimation of translation quality**, or *quality estimation* (QE) for short, can boost not only post-editor productivity, but also their confidence in the MT tools they rely on for support. While it has been shown that, *on average*, post-editing MT output is more efficient than translation from scratch (Plitt and Masselot, 2010; Federico *et al.*, 2012), this is not necessarily true for each and every *individual* segment. If the MT output for a particular segment is of poor quality, it may be more efficient to simply translate it from scratch. Reading, processing, and ultimately discarding as useless poor translations is cognitively demanding and thus puts additional strain on the translator. Therefore, it is highly desirable to be able to identify and suppress such translations automatically, so that they do not get into the translator's way. Automatic identification of weak spots in overall acceptable MT output can also guide the post-editor's attention towards them and thus improve the overall efficiency of the post-editing process.

Section 3 reports on our activity in the realm of automatically judging translation quality. This includes:

1. the development of an automatic translation quality estimation (QE) system
2. participation in a shared task on QE at WMT 2014
3. the development of adaptive QE solutions based on online and multitask learning

[1] We use the terms *translator* and *post-editor* interchangeably in this deliverable, in full acknowledgement of the fact that many translation professionals insist that (proper) translation and post-editing are different tasks requiring different skill-sets.

4. the development of technology to automatically annotate QE corpora, in order to derive binary labels that show whether a suggestion is suitable for post-editing
5. analysis of the impact of different translation errors on MT output quality
6. the release of open source QE tools and freely available resources
7. integration of the solutions we developed with the MATECAT tool.

Our work in these research areas spanned a nine months period, from November 2013 to August 2014, during which 11 scientific papers were published or accepted for publication in major venues in the field, including the *Annual Meeting of the Association for Computational Linguistics* (ACL), the *International Conference on Computational Linguistics* (COLING), the *Conference on Empirical Methods in Natural Language Processing* (EMNLP), the annual *Workshop on Statistical Machine Translation* (WMT), the *Biennial Conference of the Association for Machine Translation in the Americas* (AMTA), the *Language Resources and Evaluation Conference* (LREC), as well as the *Machine Translation Journal*. Their first pages, including abstracts, are enclosed in the appendix.

# 2 Task 3.1: Terminology Help

## 2.1 Acquisition of Terminology

The preliminary approach to terminology acquisition developed during Year 2 has been extended and improved by addressing two major challenges for its deployment in a CAT tool.

The first challenge (Sec. 2.2) is the identification and collection of bilingual terms from human post-edited data (*i.e.* ⟨*source, target, post-edit*⟩ triples), and their injection into the SMT system at runtime for continuous improvement. The second challenge (Sec. 2.3) is the extraction of bilingual terms from monolingual documents and their use in improving the SMT system in offline mode. Our work in these areas resulted in two publications (Arcan *et al.*, 2014a,b).

## 2.2 Bilingual Term Extraction from Post-Edited Data (Arcan *et al.*, 2014b)

In Arcan *et al.* (2014b) we propose a framework for extracting bilingual terms from a post-edited corpus and using them to enhance the performance of an SMT system embedded in a collaborative CAT environment, where a large translation project is split across different translators, and where each translator post-edits a limited amount of sentences per day. Our approach takes advantage of such post-edited data to gather bilingual terms specific to the domain. The parallel data produced each day is then used to continuously improve a generic machine translation system by: *i)* injecting the bilingual terms into the SMT system, and *ii)* re-optimising the system's parameters on this specific data.

Bilingual term extraction is performed in two steps. First, the source and the target sides of the corpus are processed by a keyword extractor in order to identify the most relevant terms in each language. Taking advantage of the parallelism of the data, each monolingual term in the source language is then paired with a term in the target language. We compare different techniques to perform this step and show that simple approaches based on word alignment and term translation are more robust and more efficient than the state-of-the-art method of Aker *et al.* (2013), which relies on supervised classification.

Previous work on integrating terminology into SMT (Bouamor *et al.*, 2012) assumed a batch translation scenario. Newly acquired terms were added to the training data or appended at the end of the phrase table. Neither route is feasible in our scenario, because we cannot stop the translation service and let translators wait for the update to be carried out.

Instead, we investigated for the first time the integration of cache-based translation and language models (Bertoldi *et al.*, 2013) specifically in the context of embedding terminology. The cache-based model makes it possible to add bilingual terms into a running SMT system, without the need to stop and restart it. We compared this approach to two alternatives. The first

is translation pegging, an technique that dates back to the *ReWrite* decoder, the first publicly available statistical machine translation system.[2] In this approach (made possible by the Moses Toolkit)), MT-specific attributes in XML markup of the translation input provide a mechanism to ask the MT engine to choose a particular translation for a specific span of input text. The second alternative is a general mechanism to learn from post-edited data in the context of hierarchical phrase-based translation (Denkowski *et al.*, 2014). It relies on lexicalized synchronous context-free grammars, takes as input the whole source and post-edited sentences and updates the models based on the new training instances. The latter of the two comparisons aimed at measuring how MT performance differs if we add to the cache-based model only bilingual terms instead of the whole sentence. MT performance was measured by the BLEU score (Papineni *et al.*, 2002).

Our framework was evaluated on an English to Italian translation task in two domains: medical and information technology (IT).

To evaluate the intermediate steps of the process (*e.g.* keyword extraction, term alignment) we focussed on the IT domain. We used freely available data which were manually annotated. These include a portion of the GNOME project data (4.3K tokens[3]) and the KDE data (9.5K tokens[4]).

The entire framework (including machine translation) was then tested on a subset of the EMEA corpus (Tiedemann, 2009) for the medical domain (18K tokens), and on an IT corpus (18K tokens) extracted from a software user manual. Each corpus was split into parts of around 3K tokens, which the estimated daily workload of a professional translator in post-editing, resulting in 6 person days worth of post-editing in each domain. Overall, our results suggest that:

1. An SMT model enriched with the identified bilingual terms substantially improves translation quality in terms of BLEU score over a generic SMT system. Incrementally tuned systems always outperformed those whose weights were left fixed after the initial pre-deployment parameter tuning.

2. Strategies to integrate terminology also need to consider the context of a translated term. Hard translation pegging in its straightforward implementation forces a particular translation regardless of context. For proper lexical choice, the cache-based model offers a better integration of the acquired terms into the final translation.

[2] http://www.isi.edu/natural-language/software/decoder/manual.html
[3] https://l10n.gnome.org/
[4] http://i18n.kde.org/

## 2.3 Bilingual Term Extraction from Monolingual Documents (Arcan *et al.*, 2014a)

The work we published in Arcan *et al.* (2014a) addresses the problem of automatically identifying terms in a source language document, retrieving translations from monolingual Wikipedia articles that are cross-lingually linked, and embedding them into the SMT system.

This work builds and improves on *The Wiki Machine*[5], a tool for identifying monolingual terminology in a text, disambiguating ambiguous terms and linking them to the corresponding page in WikiPedia. The original annotation process consists of a three-step pipeline based on statistical and machine learning methods which exclusively rely on Wikipedia data to train the models. The first step identifies and ranks the terms by relevance using a simple statistical approach based on *tf-idf* weighting, where all the n-grams, for n from 1 to 10, are generated and the *idf* is directly calculated from Wikipedia pages. The second annotation step links the extracted terms to Wikipedia pages. The linking problem is cast as a supervised word sense disambiguation problem. Again, Wikipedia is used, this time to provide the sense inventory and the training data, i.e. for each sense, a list of phrases where the term appears. This method was first introduced in Mihalcea (2007). The final step enriches the linked terms using information extracted from Wikipedia and Linked Open Data (LOD - http://lod-cloud.net/) resources. The additional information relative to the pair term or Wikipedia page consists of alternative terms (*i.e.* orthographical and morphological variants, synonyms, and related terms), images, topic, type, cross language links, etc. For example, in the text "*click right mouse key to pop up menu and Gnome panel*", The Wiki Machine identifies the terms *mouse*, *key*, *pop up menu*, and *Gnome panel*. For the ambiguous term *mouse*, the linking algorithm returns the Wikipedia page 'Mouse_(computing)', and the other terms used to link that page in Wikipedia with their frequency, *i.e. computer mouse*, *mice*, and *Mouse*.

Our extensions to the Wiki Machine are the following:

- a domain filter that removes terms from the list returned by the Wiki Machine that are not domain-specific;
- a cross-lingual linking algorithm that associates the extracted terms with their corresponding translations

### 2.3.1 Domain filtering of term lists

To identify specific terms, we first assign a domain to each linked term in a text, then obtain the most frequent domain, and finally filter out the terms that are out of scope. The large

[5] https://bitbucket.org/fbk/thewikimachine/

number of languages and domains have to be covered in a real CAT application prevents us from using standard text classification techniques to categorize each document. For this reason, we implement an approach based on a mapping from Wikipedia categories to *WordNet* domains (Bentivogli *et al.*, 2004). The domain for a term is obtained as follows: First, for each term, we extract its set of categories, $C$, from the Wikipedia page linked to it. Second, by means of a recursive procedure, all possible outgoing paths (usually a large number) from each category in $C$ are followed in the graph of Wikipedia categories. When one of categories which are mapped to a WordNet domain is found, the algorithm stops and assigns the WordNet domain to the term. In this way, potentially many domains are assigned to a single term. Third, to isolate the most relevant one, these domains are ranked according to the number of times they have been found following all the paths. The most frequent domain is assigned to the term. Although this process needs the human intervention for the manual mapping, this is done once and it is less demanding than annotating large amounts of training documents for text classification, because it does not require the reading of the document for topic identification.

To revisit the example given earlier: the term *mouse* is accepted because it belongs to the same domain *(computer_science)* as the majority of terms (*mouse, pop up menu, and Gnome panel*), while the term *key* in the (incorrectly assigned) domain *music* is rejected.

### 2.3.2 Obtaining term translations

The translations of the extracted domain-specific terms are then obtained by following cross-language links in Wikipedia. These links, however, provide an alignment at page level and not at term level. To deal with this issue we apply the following heuristic: if the term is equal to the source page title (ignoring case) we return the target page; otherwise, we return the most frequent alternative form of the term in the target language.From the previous example, the system is able to return the Italian page *Mouse* and all terms used in the Italian Wikipedia to express this concept of *Mouse* in *computer_science*. Using this information, the term *mouse* is paired with its translations into Italian.

To evaluate the quality of our bilingual term extraction algorithm, we compared the term–translation pairs to bilingual term lists obtained from the online service TaaS.[6] The evaluation criterion was the improvement in MT quality after feeding the information into the MT pipeline. We used the same English-Italian data for evaluation as in Arcan *et al.* (2014b; cf. Sec. 2.2). Two strategies for terminology embedding were compared: the Fill-up model (Bisazza *et al.*, 2011), which emphasizes phrase pairs extracted from the bilingual terms, and the XML markup approach (translation pegging, see Section 2.2), which uses the terms as preferred translation

[6]TaaS – https://demo.taas-project.eu/ – is a cloud-based platform for terminology services based on the state-of-the-art terminology extraction and bilingual terminology alignment methods.

candidates at run time.

Both the sources of terminology lead to significant improvements (up to 2 points in BLEU) over a baseline SMT system without embedded terminology. Neither of the two term identification methods clearly outperformed the other across all test sets, which suggests some complementarity of the two approaches.

# 3 Task 3.2: Sentence and Word Level Confidence

This section reports on the activity carried out on *Task 3.2 "Sentence and Word Level Confidence"* during the third year of MATECAT. Along this strand of research, we consolidated the work carried out during Year 2 and extended it as follows:

- We improved the Quality Estimation (QE) system developed during Year 2 by adding new features that capture word-level information. This led to a successful participation in the shared task on QE organized within the $9^{th}$ Workshop on Machine Translation. The joint participation, a collaboration of FBK, the University of Edinburgh, and the Universitat Politècnica de València (a member of the CasmaCat[7] project), achieved the best results in two out of the three tasks that we participated in. Details are reported in Section 3.1.
- We developed methods for adaptive QE that address the specific challenges posed by the integration of this functionality into a CAT environment, in particular the dearth of representative training data and the need for domain adaptation. The outcomes of this activity are reported in Section 3.2.
- We conducted experiments on binary `OK` / `BAD` QE with real users operating with the MATECAT tool to explore the impact of human subjectivity on data annotation. The outcomes of this activity are reported in Section 3.3.
- We developed an effective method to assess the impact of different types of MT errors on translation quality. Details are reported in Section 3.4.
- Tools and data sets that we developed in the course of this work were released as open-source resources. They are listed in Section 3.5.

## 3.1 Quality Estimation System (C. de Souza *et al.*, 2014a)

The possibility to reduce the cost of translation by post-editing good-quality MT output calls for methods to automatically decide *what* to present as a suggestion, and *how* to do it in the

[7] http://www.casmacat.eu/; also funded under the EU's FP7 programme.

most effective way. These issues, which are particularly relevant to MATECAT, have sparked interest and research on automatic QE, which addresses the problem of estimating the quality of a translated sentence without access to reference translations (Blatz *et al.*, 2003; Specia *et al.*, 2009; Mehdad *et al.*, 2012). QE is generally cast as a supervised machine learning task, where a model trained from a collection of $\langle source, target, label \rangle$ triples is used to predict labels for new unseen test items, i.e. pairs of $\langle source, target \rangle$ (Specia *et al.*, 2010). These labels can refer to post-editing effort (e.g., the percentage of MT output tokens that have to be corrected, quantized to a Likert-style scale from 1 to 5), *Human-mediated Translation Edit Rate* (HTER; Snover *et al.*, 2006), or post-editing time.

In the past three years, the shared tasks on QE organized within the WMT workshop series[8] have served as a major venue to measure improvements on QE. Building on the positive results we achieved in the 2013 evaluation campaign (C. de Souza *et al.*, 2013a), we developed and submitted an improved system for 2014. Besides representing an ideal scenario to evaluate our progress, the WMT 2014 QE task provided the MATECAT consortium with a good opportunity for collaboration between partners of the project (FBK and University of Edinburgh), and also with a team from the CasmaCat project (Universitat Politècnica de València).

The FBK-UPV-UEdin QE system (C. de Souza *et al.*, 2014a) participated in the following WMT 2014 QE tasks/sub-tasks (all for English-Spanish), ranking at the top in two (T1.2, T2) of thm:

- Task T1 – Sentence-level QE
  - Subtask T1.2. – Prediction of the Human-mediated Translation Edit Rate (HTER) between a suggestion generated by a machine translation system and its manually post-edited version.
  - Subtask T1.3. – Prediction of post-editing time, *i.e.* the time required to post-edit a suggestion given by a machine translation system.
- Task T2 – Word-level QE.
  This task requires participants to classify each word in a translation hypothesis according to one of the following classification schemes:
  - Binary: `OK/BAD`, where `BAD` indicates that the word in question requires editing.
  - Level-1 classification: `OK`, `Accuracy`, or `Fluency`, splitting the class `BAD` into words that need to be edited because of translational inaccuracy and words that require editing because of target-language disfluency.

[8] http://www.statmt.org/wmt14/

– Multi-class with 20 distinct error classes as described in the shared-task description[9] plus `OK` for words with no error.

In Task T2, we participated in the English–Spanish track.

### 3.1.1 Sentence-level QE

**Features.** We added 18 new features based on word-level quality classifications to the widely used "baseline" features implemented in the open source tool QUEST[10], and to the other[11] features used in our system from last year. These are listed in Table 1.

**Algorithms.** QE models for both sentence-level tasks (T1.2 and T1.3) are trained using extremely randomized trees (ET) (Geurts *et al.*, 2006).

**Results.** For Task 1.2 (HTER), our submission performed best out of 10, with a statistically significant improvement in the Mean Absolute Error (MAE) over the official baseline (12.89 vs. 15.23). Our second contrastive baseline submission (MAE: 14.38), using only last year's features, also outperformed the official baseline. The two results confirm the effectiveness of last year's features as well as the value that the new features add.

In Task 1.3 (post-editing time), the better of our two submissions ranked $6^{th}$ out of 9 with an MAE of 17.48, but the difference to the winning system (16.77) is not statistically significant.

### 3.1.2 Word-level QE

**Features.** The feature set used for the *Word-level QE* task comprises:

- **Word Posterior Probabilities (WPP).** Unlike in previous years, in 2014 the MT system that produced the translations for the shared task was **not** made available to the participants. Thus, to compute WPPs, we approximated the decoder's search space as well as an N-best list of possible translations by re-translating the source using the system that was made available for the 2013 WMT QE Shared Task (Bojar *et al.*, 2013). As proposed by previous work (Blatz *et al.*, 2004; Ueffing and Ney, 2007; Sanchis *et al.*, 2007), posterior probabilities were computed over n-best lists, in our case 100k-best lists.

- **Confusion Networks (CN).** The same N-best lists serve to compute features based on the graph topology of confusion networks (Luong *et al.*, 2014). This network is obtained through Levenshtein alignments of all the translations in the N-best list. Word edges are

[9] http://www.statmt.org/wmt14/quality-estimation-task.html
[10] http://www.quest.dcs.shef.ac.uk/
[11] Last year's system (C. de Souza *et al.*, 2013a) exploited the following information for quality prediction: *(i)* word alignment information (C. de Souza *et al.*, 2013b), *(ii)* the diversity of n-best translations, *(iii)* word posterior probabilities, *(iv)* pseudo-references and *(v)* back-translations.

Table 1: New features for sentence-level quality estimation.

1. the proportion of OK words among **all** the words in the sentence
2. the proportion of OK words among all the **function words** in the sentence
3. the proportion of OK words among all **content words** in the sentence
4. the proportion of OK words among all **nouns** in the sentence
5. the proportion of OK words among all **verbs** in the sentence
6. the proportion of OK words among all **proper nouns** in the sentence
7. the proportion of OK words among all **adjectives** in the sentence
8. the proportion of OK words among all **pronouns** in the sentence
9. the length of the **longest consecutive sequence of** OK **words** divided by the total number of OK words in the sentence
10. the length of the **longest consecutive sequence of** BAD **words** divided by the total number of BAD words in the sentence
11. the proportion of OK **bigrams** among all the bigrams in the sentence in the sentence
12. the proportion of OK **trigrams** among all the trigrams in the sentence in the sentence
13. the proportion of OK **4-grams** among all the 4-grams in the sentence in the sentence
14. the proportion of OK **5-grams** among all the 5-grams in the sentence in the sentence
15. the proportion of OK words in the **first half** of the sentence.
16. the proportion of OK words in the **second half** half of the sentence.
17. the proportion of OK words in the **first quarter** of the sentence.
18. the proportion of OK words in the **second quarter** of the sentence.

then weighted with the respective posterior probabilities. For each word in the hypothesis, we extract from the respective confusion set: maximum and minimum probability in the set (2 features); the number of alternatives in the set (1 feature); and the entropy of the alternatives in the set (1 feature).

- **Language Models (LM).** As language model features, we compute for each word in the sentence its conditional probability as well as the maximum length of the matching n-gram. Since not all n-grams are observed this value is often different from $n$. This feature is commonly referred to as *backoff behaviour*. We employ both an interpolated LM taken from the MT system made available for the 2013 WMT QE Shared Task (Bojar *et al.*, 2013), and a very large LM (Buck *et al.*, 2014) which we build on 62 billion tokens of monolingual data extracted from the public web crawl Common Crawl.[12]

- **Word Lexicons (WL).** For every target word $e_i$, we compute two different features based on statistical word lexicons (Blatz *et al.*, 2004):

  Avg. probability: $\frac{1}{|\mathbf{f}|+1} \sum_{j=0}^{|\mathbf{f}|} \mathrm{P}(e_i \mid f_j)$

[12]This language model and others are available for download at http://statmt.org/ngrams.

Max. probability: $\max_{0 \leq j \leq |\mathbf{f}|} \mathrm{P}(e_i \mid f_j)$

where $\mathrm{P}(e \mid f)$ is a probabilistic lexicon, $e_i$ is the target word, $f_1 \ldots f_{|f|}$ are the source words, and $f_0$ is the "NULL" word (Brown *et al.*, 1993).

- **POS tags (POS).** Source and target sentence are part-of-speech tagged with *TreeTagger* (Schmid, 1995). The POS tags allows us to obtain a vector that represents the probability distribution of source POS tags for each target word. Additionally, we extract a binary feature that indicates whether the word is a stop word or a content word.

- **Stacking of predictions (S).** The confidence classes for the Level1 and Multi-class conditions are fine-grained versions of the Binary annotation, *i.e.* the `OK` examples are the same for all cases. This allowed us to re-use our binary predictions as an additional feature for the finer-grained classes.

**Algorithms.** For word-level QE, we use bidirectional long short-term memory recurrent neural networks (BLSTM-RNNs) as implemented in the *RNNLib* package (Graves, 2008), and conditional random field (CRF) as implemented in *Pocket CRF*.[13]
**Results.** In the word-level task, our best submission ranked top in the Binary and Level1 settings (out of 7 and 6 submissions respectively), and $2^{nd}$ out of 6 in the Multi-Class setting.

Further details about the features and the parameter optimization strategies, as well as a discussion of the results can be found in our submission description (C. de Souza *et al.*, 2014a) in the workshop proceedings.

## 3.2 Adaptive Quality Estimation

So far, despite its many possible applications, QE research has been conducted mainly under controlled laboratory conditions that disregard some of the challenges posed by real-life scenarios. The large body of research resulting from three editions of the shared QE task organized within the yearly Workshop on Machine Translation (WMT; Callison-Burch *et al.*, 2012; Bojar *et al.*, 2013, 2014) has indeed relied on assumptions that do not always hold in real life. Among these assumptions is the idea that the data available to train QE models is *(i) large* (WMT QE systems are usually trained over datasets of 800/1000 instances, the remaining instances being reserved for development and testing) and *(ii) representative* (WMT QE training and test sets are always drawn from the same domain and are uniformly distributed). However, moving to real working environments calls for solutions to cope with the variable conditions of a translation job. Such variability is due to two main reasons:

[13] http://pocket-crf-1.sourceforge.net/

1. **The notion of MT output quality is highly subjective** (Koponen, 2012; Turchi *et al.*, 2013; Turchi and Negri, 2014). Since the quality standards of individual users may vary considerably (*e.g.* according to their knowledge of the source and target languages), the estimates of a static QE model trained with data collected from a group of post-editors might not fit with the actual judgements of a new user;

2. **Each translation job has its own specificities** (domain, complexity of the source text, average target quality). Since data from a new job may differ from those used to train the QE model, its estimates on the new instances might be biased or uninformative.

The need for QE systems to adapt to the behaviour of specific users and across domain changes is an aspect of the QE problem that has been mostly ignored so far. A common trait of all current approaches is the reliance on batch learning techniques, which assume a "static" world where new, unseen instances that are encountered will be similar to the training data. In order to develop QE models for realistic scenarios where such assumptions might not hold (first of all the CAT framework), part of the MATECAT QE activities during Year 3 focussed on tackling the task in situations where training datasets are small and/or not representative of the actual testing domain. For these situations, which are particularly challenging from the machine learning perspective, we investigated the potential of *online* (Sec. 3.2.1), and *multitask* (Sec. 3.2.2) learning in comparison to the batch learning algorithms used currently. These research activities resulted in two publications in major NLP conferences: Turchi *et al.* (2014a; ACL) and C. de Souza *et al.* (2014c; COLING).

### 3.2.1 Online Learning for Adaptive QE (Turchi *et al.*, 2014a)

In contrast to batch learning scenarios, where models are trained once and then applied to unseen data, online learning algorithms update their models with every single training instance. This allows them to respond to user feedback and to changes in the overall structure of the data that they encounter. Incorporation of online learning into the QE system requires the adaptation of its standard batch learning workflow to:

1. perform feature extraction from a $\langle source, target \rangle$ pair one instance at a time instead of all at once for an entire training set;

2. produce a quality prediction for the input instance;

3. gather user feedback for the instance (*i.e.*, calculate a "true label" based on the amount of user post-edits);

4. send the true label back to the model to update its predictions for future instances.

For adaptation to user and domain changes, we experimented with:

- different online learning algorithms, specifically OnlineSVR[14] (Parrella, 2007) and the Passive-Aggressive Perceptron[15] (Crammer *et al.*, 2006).

- different learning strategies. We compared *adaptive* and *empty* models against a system trained in *batch* mode using the Scikit-learn implementation of Support Vector Regression (SVR).[16] The *adaptive* model is built on top of an existing model created from the training data, and exploits the new test instances to refine its predictions in a stepwise manner. The *empty* model only learns from the test set, simulating the worst condition where training data is not available at all. The *batch* model is built by learning only from the training data and is evaluated on the test set without exploiting information from the test instances.

- different datasets for different language combinations and domains. We used the WMT12 English-Spanish corpus and two English-Italian corpora created with the MATECAT tool (Federico *et al.*, 2014b). The former contains 2,254 source-target pairs from the news domain; the latter covers the legal and information technology domains (164 and 280 sentences respectively).

- different testing conditions. These range from the easiest situation, where training and test data feature homogeneous label distributions (adaptation capabilities are not required and batch methods operate in the ideal conditions), to an intermediate situation (user changes within the same domain) and the most difficult one (user and domain changes at the same time) where the differences between training and test call for adaptive solutions to the learning problem. In the hardest condition (user and domain changes) we also consider the users' behaviour, selecting training/test data from post editors featuring either similar or dissimilar post-editing attitudes (*i.e.* radical or conservative). On each test set, featuring different degrees of similarity with the data used for training, evaluation is carried out by measuring the global error (MAE), similar to the evaluation framework of the WMT QE shared tasks.

Our results, always obtained with the same feature set,[17] show that the sensitivity of online QE models to different distributions of training and test instances makes them more suitable than batch methods for integration in a CAT framework. Despite slight variations across the different testing conditions (*e.g.* , neither of the the two online learning algorithms performs consistently better than the other), global MAE scores for the online algorithms in both training

[14] http://www2.imperial.ac.uk/~gmontana/onlinesvr.htm
[15] https://code.google.com/p/sofia-ml/
[16] http://scikit-learn.org/
[17] The 17 "baseline" features implemented in QUEST.

regimes — *adaptive* and *empty* — significantly outperform the results achieved by the batch models. Interestingly, in many cases the best results are achieved by the *empty* models, with MAE reductions up to 10 points when tested in the most challenging scenarios. These results suggest that, when dealing with datasets with very different label distributions, the evident limitations of batch methods are more easily overcome by learning from scratch from post-editors' feedback. This also holds when the amount of test points to learn from is limited, as in the legal domain where the test set contains only 80 instances.

From the application-oriented perspective that motivated our work, these findings are particularly important results in the light of the high costs of acquiring large representative QE training data and the need of methods to quickly adapt to user and/or domain changes.

### 3.2.2 Multitask Learning for Adaptive QE (C. de Souza *et al.*, 2014c,b)

A possible alternative to cope with data heterogeneity is to share information across domains. This would allow to learn a QE model for a specific target domain by exploiting training instances from different domains. This research direction was explored in C. de Souza *et al.* (2014c). Our goal was to develop approaches that allow not only learning from one single source domain, but also from multiple source domains simultaneously, leveraging the labels from all available data to improve results in a target domain. We thus approach this problem as a *Multitask learning* (MTL) problem, which uses domain-specific training signals of related tasks to improve model generalization on a target task (Pan and Yang, 2010).

An important assumption in MTL is that different tasks (domains in our case) are correlated via a common structure, which allows for knowledge transfer among tasks. MTL has been demonstrated to improve model generalization over single task learning (STL) for various problems in several areas. Under this scenario, several assumptions can be made about the relatedness among the tasks, leading to different transfer structures. Focusing on the capability to learn from scarce amounts of training data from different domains/genres with different label distributions, we experimented with:

- three approaches to MTL that deal with task relatedness in different ways: "dirty" (Jalali *et al.*, 2010), *Sparse Trace* (Chen *et al.*, 2012) MTL and *Robust* (Chen *et al.*, 2011). The three approaches use different regularization techniques that capture task relatedness using norms over the weights of the features.

- three very different domains for the language pair English–French: newswire text (News), transcriptions of Technology Entertainment Design talks (TED) and Information Technology manuals (IT). These domains are a challenging combination for adaptive systems since they come from very different sources spanning speech and written discourse (TED

and News/IT, respectively) as well as a well-defined and controlled vocabulary in the case of IT.

- the 17 "baseline" features implemented in the QUEST tool.

The News corpus contains newswire text used in WMT translation campaigns and covers topics different from those covered in the TED corpus.

The TED talk dataset (Cettolo *et al.*, 2012) is used for MT and automatic speech recognition systems evaluation within the International Workshop on Spoken Language Translation (IWSLT). The dataset consists of subtitles of several talks in a range of topics presented in the TED conferences.

The sentence tuples for the first two domains were randomly sampled from the TRACE corpus,[18] a corpus of professionally post-edited MT output. The translations were generated by two different MT systems: a state-of-the-art phrase-based statistical MT system, and a commercial rule-based system, and then post-edited by up to four translators (Wisniewski *et al.*, 2013).

The IT texts come from a software user manual. They were translated with a phrase-based SMT system on the basis of *Moses* (Koehn *et al.*, 2007) and post-edited by a single professional translator working with a recent version of MATECAT (Federico *et al.*, 2014b).

Each dataset consists of 363 tuples formed by the source sentence in English, the French translation produced by an MT system and a human post-edit of the translated sentence. Each pair of translation and post-edit was labelled with the respective HTER score. Half of the data in each domain was used for training (181 instances); the other half for testing (182 instances). The reduced amount of instances for training contrasts with the 800 or more instances of the WMT evaluation campaigns and is closer to real-world scenarios where the availability of large training sets is far from being guaranteed.

Each MTL model is evaluated on its mean absolute error (MAE) and compared against three baselines. The first is simple to implement but difficult to beat when dealing with regression on tasks with different distributions. It consists of computing the mean of the training labels and using it as the prediction for each testing point (Rubino *et al.*, 2013). Since supervised domain adaptation techniques should outperform models that are trained only on the available in-domain data, we use as a second baseline the regressor built only on the available in-domain data (SVR in-domain). Finally, as a third baseline, we train a regressor by pooling together training data of all domains, combining source and target data without any kind of task relationship notion (SVR Pooling). The baselines are trained on the same feature set, with an SVM regression (SVR) method using the implementation of Scikit-learn (Pedregosa *et al.*, 2011).

[18] http://anrtrace.limsi.fr/trace_postedit.tar.bz2

Results are calculated on varying amounts of training data, ranging from 18 (10%) to the full amount of 181 instances. The motivation is to determine how much training data is required by the MTL methods to outperform the baselines for a target domain. Each method was run on 50 different train/test splits of the data in order to account for the variability of points in each split. Overall and in all three domains, Robust MTL (RMTL) tends to perform best among all models under investigation. It always outperforms the baseline significantly.[19] Performance generally improves as the amount of training data is increased. For example, RMTL has an improvement in performance of 5.13% for TED, 4% for News and 1.85% for IT when trained on 100% of the training data in comparison to the model trained on 30% of training data. Overall, our experiments produced encouraging results with respect to coping with QE data coming from different domains/genres, translated by different MT systems and post-edited by different translators. Even in such difficult conditions, the methods investigated are capable of outperforming competitive baselines on different domains.

In a very recent work (C. de Souza *et al.*, 2014b). we further investigated the effectiveness of the multitask and online learning approaches to QE. In order to estimate and compare the potential of the two approaches, experiments were carried out in the same experimental setting (English-French newswire texts, Information Technology manuals, and TED talks transcriptions). Our results confirm the effectiveness of both methods (they both produce better models than the single task learning and pooling strategies) and suggest the possibility of an integration of the two strategies as a future research direction (*i.e.* an online MTL method that combines the capability of MTL to transfer knowledge from different domains with the capability of online learning to train incremental models that can leverage also the test data).

## 3.3 Binary quality estimation (Turchi and Negri, 2014; Turchi *et al.*, 2014b)

In the second year of the project we investigated binary QE as an alternative MT quality estimation method. We performed a task-oriented analysis of the usefulness of the available human-annotated datasets for binary quality estimation, and developed an automatic method to re-annotate with binary judgements the English-Spanish QE corpus that was used for the WMT 2012 shared task on QE (Turchi *et al.*, 2013).

These accomplishments of the second year were extended and improved upon in the third year, leading to a journal article (Turchi *et al.*, 2014b) and a conference paper (Turchi and Negri, 2014). The activities and research results of the third year can be summarized as follows:

1. The automatic annotation procedure was experimentally evaluated for new language pairs,

[19] as measured by the Friedman test (Friedman, 1937, 1940) and a post-hoc analysis based on the Holm's procedure (Holm, 1979) to perform pairwise comparisons between regressors when the null hypothesis is rejected.

covering different domains and produced by different post-editors.

2. We were able to show that thresholds separating useful from useless MT suggestions can be empirically estimated, even from a relatively small amount of data and under various conditions (language pairs, domains).

3. Such thresholds are always significantly lower than the values proposed by existing QE data annotation guidelines.

4. Our empirical findings have been confirmed by the results of a verification involving several human post-editors operating with the MATECAT tool.

5. The automatic annotation method has been applied to release a freely available binary QE corpus for three language pairs.

### 3.3.1 Experiments with English-Italian: automatic annotation and threshold analysis

**Experimental setup.** The experiments reported in Turchi *et al.* (2014b) used an English-Italian dataset (CAT henceforth) from the legal domain created using the MATECAT tool. The dataset consists of 1,155 ⟨*source, target, post-edit*⟩ triples (946 for training, 209 for testing). Source and target were extracted from four parallel documents of a European Parliament resolution published on the EUR-Lex platform. The source sentences were translated by a *Moses* system trained on 1.5M parallel sentences extracted from the Acquis corpus (Steinberger *et al.*, 2006). Post-edits by professional translators using the MATECAT tool under real-life conditions were collected by MATECAT partner Translated.

Similar to our previous work (Turchi *et al.*, 2013), the CAT training set is partitioned in different ways to obtain suitable data for binary QE, reserving a fraction of the data for evaluation. The arbitrary partitions are obtained according to eleven different HTER thresholds ranging from *0.75* to *0.3* (including the value of *0.3013*, which leads to the most balanced separation into positive and negative instances). Then, based on the method proposed in (Turchi *et al.*, 2013), the classification performance of the binary models obtained from these arbitrary partitions is compared with the results of a model trained on the automatic re-annotation of the same dataset. The classifiers (SVM) and feature sets remain the same. CAT test sets were partitioned according to ten HTER thresholds ranging from *0.75* to *0.3*. Performance was measured by average of the *false positive rate* (FPR) and *false discovery rate* (FDR), which is sensitive to the number of false positives also with unbalanced test sets.

**Results.** Evaluation results confirm, also on this new language pair and domain, our previous findings (Turchi *et al.*, 2013). On all the test sets, the classifier trained on the automatically annotated data (*AA*) shows the lowest error rates. The effectiveness of the automatic annotation

is confirmed by the fact that also the classifiers which are trained on more balanced training sets (including the one resulting from the threshold value set at *0.3013*) achieve worse results than the *AA* classifier. Even for the most unbalanced test set (*0.75 HTER*), an error smaller than 10% indicates the high capability of the *AA* classifier to control the number of false positives (only 4) keeping a good level of true positives. It is worth noting that such coherent results are obtained despite the fact that: *(i)* the WMT and CAT data cover different domains and language pairs, and *(ii)* the two datasets considerably differ in size (the classifier obtained for the CAT data is trained on an automatically annotated corpus whose size is half of the corpus used to learn the WMT models). The good classification results achieved in such different conditions demonstrate that our re-annotation method is not only more reliable than the strategies based on arbitrary data partition, but also robust and portable across different scenarios. The classification improvements observed also with the smaller CAT dataset suggest that high-quality training data for binary QE can be obtained from a relatively small amount of data.

**Threshold analysis.** An interesting finding of this research concerns the HTER threshold used to separate good/useful MT output (*i.e.* worth to post-edit) from bad/useless suggestions (*i.e.* which need complete rewriting). The resulting thresholds, automatically identified by applying our method, are similar across different datasets and surprisingly lower than the value generally assumed as a "reasonable" boundary. While the WMT-12 annotation guidelines indicate an HTER of *0.7* as a good boundary, our automatically-estimated values for the WMT and the CAT data are *0.4* (Turchi *et al.*, 2013) and *0.32*, respectively. To follow up on this observation, we compared in a final set of experiments the automatically estimated thresholds with those indicated by the analysis of data collected from human translators working in a CAT environment. To this end, four expert translators (Italian native speakers) were asked to translate the same document (270 sentences) in controlled conditions with the MATECAT tool. Based on their work, two thresholds for each post-editor are estimated. The first one is calculated with our automatic annotation method. The second one is obtained by considering the time required to translate each sentence of the document in different conditions (either from scratch, or by post-editing suggestions of different quality produced by two MT systems). Our validation experiment is based on the following hypotheses: *(i)* post-editing good-quality suggestions requires the shortest time, *(ii)* post-editing low-quality suggestions requires more time than translating from scratch, *(iii)* the time required to translate from scratch varies according to the complexity of the source sentence, *(iv)* depending on the difficulty of the task, a threshold $t$ exists such that the effort required for post-editing is similar to the effort required to translate from scratch. If such hypotheses hold true, $t$ represents the boundary between good suggestions suitable for post-editing, and bad suggestions whose correction requires more effort than translating from scratch.

The collected data allows us to measure the difficulty of the task (both for post-editing and translating from scratch). The HTER computed over the collected (*target, post-edit*) pairs provides us with a direct estimation of the difficulty to post-edit. The TER computed over the translations from scratch and the corresponding references provides us with an indirect estimation of the complexity of the source sentences. High TER values between a human translation and a human reference (both supposed to be correct) indicate that the same source can be translated in very different, but acceptable ways. In turn, a large space of possible correct translations can be considered as an indicator of the difficulty to translate. In light of these considerations, TER and HTER (both measuring the number of editing operations needed to transform a sentence into another) are homogeneous and can be plotted against time.

Based on these intuitions we analyse the work of the four translators involved in our experiments. Interestingly, for all the translators involved in our experiment we obtain similar results that lead to the following observations. First, the difference between the empirically estimated and the validated thresholds is always minimal, thus confirming the plausibility of the thresholds obtained by applying our annotation method. Second, regarding such thresholds, the small variations observed across different human translators are confirmed by our verification. This indicates that, by applying our approach, differences between post editors which cannot be captured by arbitrary data partition strategies indeed exist and can be learned from data. Overall, despite such small variations across translators, the validated plausibility of our empirically estimated thresholds indicates that reasonable thresholds separating useful from useless MT suggestions fall in the [*0.36 - 0.42*] HTER range and are significantly lower than the value of *0.7* proposed by the existing QE data annotation guidelines.

As a final remark, it is worth observing that the reliability of our approach is minimally affected by the amount of data available. Even with the few translations that we collected in about four working days (270 sentences), we are able to empirically estimate translator-specific thresholds that are very close to the values observed in a controlled verification experiment.

## 3.4 Assessing the impact of translation errors on MT output quality (Federico *et al.*, 2014a)

In the work reported in (Federico *et al.*, 2014a), we analyzed the impact of different error types on MT output quality. We investigated the following questions:

- Which types of MT errors have the highest impact on human perception of translation quality? Surprisingly, there is little prior work that focuses on this issue. Error annotations have been considered to highlight strengths and weaknesses of MT engines or to investigate the influence of different error types on post-editors' work. However, the

direct connection between errors and users' preferences has been only partially understood, mainly from a descriptive standpoint and through rudimentary techniques unsuitable to draw clear-cut conclusions or reliable inferences (Popović and Ney, 2011; Kirchhoff *et al.*, 2013).

- To which types of errors are different MT evaluation metrics more sensitive? This problem has been even less explored. For instance, little has been done to understand which automatic metric is more suitable to assess system improvements with respect to a specific issue (*e.g.* word order or morphology) or to shed light on the joint impact of different error types on performance results calculated with different metrics.

**Method.** To answer these questions, we developed a robust statistical framework to analyse the impact of different error types, alone and in combination, both on human perception of quality and on MT evaluation metrics' results. Our analysis is based on *linear mixed-effects models* (Baayen *et al.*, 2008), a generalization of linear regression models suited to model responses with fixed and random effects. Mixed-effects models, like any regression model, express the relationship between a *response variable* and some *covariates* and/or *contrast factors*. They enhance conventional models by complementing *fixed effects* with so-called *random effects*. Random effects are introduced to absorb random variability inherent to the specific experimental setting from which the observations are drawn. In general, random effects correspond to covariates that are not - or cannot be - exhaustively observed in an experiment, *e.g.* the human annotators and the evaluated systems. Hence, mixed models permit to elegantly cope with experimental design aspects that hinder the applicability of conventional regression models. These are, in particular, the use of repeated and/or clustered observations that introduce correlations in the response variable that clearly violate the independence and homoscedasticity assumptions of conventional linear, ANOVA, and logistic regression models. In our work, we employed mixed *linear* models to measure the influence of different MT error types, expressed as continuous fixed effects, on quality judgements or on automatic quality metrics.

**Experimental setting.** Experiments were performed on data covering three translation directions (English to Chinese, Arabic and Russian). For each direction, two automatic translations were collected for around 400 sentences and manually evaluated by expert native translators through absolute quality judgements (1-5 Likert scores) and error annotation (in terms of reordering errors, lexical errors, missing words and morphology errors). Regarding the automatic metrics, we computed sentence-level BLEU (Lin and Och, 2004), TER (Snover *et al.*, 2006), and GTM (Turian *et al.*, 2003) by relying on single references and by means of standard packages.

**Results and findings**. To assess the impact of translation errors on MT quality we performed two sets of experiments. The first set addressed the relation between errors and human quality

judgements. The second set focused on the relation between errors and automatic metrics.

In both cases, before measuring the impact of different errors on the response variable (respectively quality judgements or metrics), we validated the effectiveness of mixed linear models by comparing their prediction capability with other methods (simpler linear models including only fixed effects). Prediction performance was computed in terms of Mean Absolute Error (MAE) In this preliminary experiment we observed that, on all language combinations, mixed linear models achieve the lowest MAE significantly outperforming all the other models. These improvements are due to the addition of random effects, the possibility to cope with the erratic factors that might influence empirical observations in a given setting (*e.g.* the low agreement between human annotators), and the possibility to identify and model the contribution of the different error types.

Based on these positive outcomes, we then analysed the impact of different error types (alone and in combination) on human quality scores. This leads us to the following interesting findings. First, the impact varies across the different language combinations. For instance, while for English-Chinese and English-Russian *missing words* have the highest impact, the most problematic issue for English-Arabic is represented by *lexical errors*. Second, we observe that in some cases the combined impact of different error types is lower that the cumulative impact of the single errors. The existence of such "discount" effects of various magnitude associated to the different error combinations is a novel finding made possible by the adoption of mixed-effect models. A third interesting observation is that, in contrast with the common belief that the most frequent errors have the highest impact on human quality judgements, our experiments do not reveal such strict correlation (at least for the examined language pairs). For instance, for English-Chinese and English-Russian the impact of *missing word* errors is much higher than the impact of other more frequent issues.

Finally, we analysed the impact of different error types (alone and in combination) on automatic metrics' scores. Also in this case, mixed linear models allow to get interesting insights about the influence of each specific error types and the impact of error combinations on the performance results measured with different automatic metrics. Remarkably, for some translation directions, some of the metrics show a sensitivity to errors that is very similar to that of human judges. In particular, BLEU for English-Chinese and English-Arabic, and GTM for English-Chinese show a very high correlation with the human sensitivity to translation errors, with Pearson correlation coefficient $\geq 0.97$.

In three cases (BLEU for English-Chinese, GTM for English-Chinese and English-Arabic) the analysed metrics are most sensitive to the same error type that has the highest influence on human judgements (these are *missing* for English-Chinese and English-Russian, *lexical* for English-Arabic). On the contrary, in one case (TER for English-Chinese) the analysed met-

ric is insensitive to the error type (*missing*) which has the highest impact on human quality scores. From a practical point of view, these remarks provide useful indications about the appropriateness of each metric to highlight the deficiencies of a specific system and to measure improvements targeting specific issues. As a rule of thumb, for instance, to measure improvements of an English-Chinese system with respect to *missing words* errors, it would be more advisable to use BLEU or GTM instead of TER.

Similar considerations also apply to the analysis of the impact of error combinations. The same discount effects observed when analysing the impact of errors' co-occurrence on human perception are evidenced, with different degrees of sensitivity, by the automatic metrics. While some of them substantially reflect human response (*e.g.* BLEU and GTM for English-Chinese), in some cases we observe either insensitivity to specific combinations (mostly for English-Arabic), or a higher sensitivity compared to the values measured for human assessors (mostly for English-Russian, where the impact of *missing+reoordering* combinations is discounted - hence underestimated - by all the metrics). Such differences provide useful indications about the most appropriate metric to assess system improvements with respect to specific weaknesses.

## 3.5 Delivered Tools and Resources

In terms of delivered tools and resources, the major outcomes of the WP3 activities during the third year of MATECAT with respect to quality estimation are the following:

- **AQET**. A tool for adaptive QE, integrating all the components and the algorithms described in Turchi *et al.* (2014a) has been released as open source. Its C++ implementation is available at http://hlt.fbk.eu/technologies/aqet.

- **Online QuEst**. An extension of QuEst, the open source framework for QE, has been released. This version, which is described in Shah *et al.* (2014), is capable to perform online feature extraction and is available at http://www.quest.dcs.shef.ac.uk/.

- **BinQE**. By applying automatic annotation method described in Section 3.3 we produced BinQE (Turchi and Negri, 2014), a freely-available corpus for binary QE that covers different language combinations. BinQE contains the annotation with binary labels of the following existing datasets:

  - The QE dataset used for Task 1.1 at WMT 2013 (Bojar *et al.*, 2013), which consists of 2,754 English-Spanish news sentences (2,254 from the training and test sets of WMT 2012 and 500 from the test set of WMT 2013).

– The French-English dataset described in Potet *et al.* (2012), which consists of 10,881 news sentences automatically translated, along with their reference translations and post-edit.

– The English-Italian dataset described in Section 3.3.1, which consists of 1,155 ⟨*source, target, post-edit*⟩ triples from the legal domain.

# 4 Task 3.3: Enriched MT Output

The purpose of QE in a post-editing scenario is to provide assistance in deciding whether the raw machine translation output is good enough to be post-edited, or whether it is more efficient to ignore it — if MT quality is too poor, assessing and editing the suggestion can be more time-consuming than translating the segment in question from scratch. On the other hand, we do not want to be too picky about quality in deciding whether or not to suggest a translation to the user: as long as it is "good enough", even imperfect MT output can greatly speed up the overall translation process. Selecting translation hypotheses is at the very least a two-stage process. First, the system must decide whether to present a translation hypothesis to the user, or to declare failure to translate. Second, the user has to decide whether or not to use a given suggestion. In order to assist the translator in that decision, it is desirable to communicate to the user the level of confidence that the system has in its own translation.

During the project meeting in February 2014, the project partners discussed and debated ways of communicating translation quality estimates to translators within the MATECAT interface, in order to help them decide more quickly whether or not to choose a given translation for post-editing. In the end, we decided on a "traffic light" indicator. According to this idea (currently under development in view of the upcoming field test in which it will be extensively evaluated with real users):

- A **green** flag indicates that it is worth to post-edit a given MT suggestion (its quality is good enough to guarantee a minimal work for the post-editor);
- A **red** flag indicates that it is better to rewrite from scratch the proposed suggestion (due to its poor quality, post-editing would require more effort).

Defining a criterion to assign green/red flags to each suggestion is an important step to develop such traffic light mechanism. As a possible solution to this problem we opted for thresholding the HTER predictions returned by our QE component. To this end, the findings of the recent works documented in Section 3.3 provide useful indications. As demonstrated in Turchi *et al.* (2014b), reasonable thresholds separating "good" from "bad" translations fall in

the [*0.36 - 0.42*] HTER range. For this reason, the final threshold value set for the field tests will be chosen within such interval.

From the algorithmic point of view, based on the positive evaluation results measured in comparison with previous batch learning methods, the new QE component used for the upcoming field tests will rely on the online learning paradigm discussed in Section 3.2.1 (i.e., Turchi *et al.*, 2014a).

# A Publications related to WP3

List of the papers produced during the third year of MateCat and accepted for publication.

1. Arcan, Mihael, Claudio Giuliano, Marco Turchi, and Paul Buitelaar. 2014a. "Identification of Bilingual Terms from Monolingual Documents for Statistical Machine Translation." *Proceedings of the 4th International Workshop on Computational Terminology (Computerm)*, 22–31. Dublin, Ireland

2. Arcan, Mihael, Marco Turchi, Sara Tonelli, and Paul Buitelaar. 2014b. "Enhancing Statistical Machine Translation with Bilingual Terminology in a CAT Environment." *Proceedings of the 11th Biennial Conference of the Association for Machine Translation in the Americas (AMTA 2014)*. Vancouver, BC, Canada. **To appear.**

3. Turchi, Marco, Matteo Negri, and Marcello Federico. 2014b. "Data-driven Annotation of Binary MT Quality Estimation Corpora Based on Human Post-edition." *Machine translation*, Special Issue on Post-editing. **To appear.**

4. Federico, Marcello, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2014a. "Assessing the Impact of Translation Errors on Machine Translation Quality with Mixed-effects Models." *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar. **To appear.**

5. C. de Souza, José G., Marco Turchi, and Matteo Negri. 2014c. "Machine Translation Quality Estimation Across Domains." *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 409–420. Dublin, Ireland

6. C. de Souza, José G., Marco Turchi, and Matteo Negri. 2014b. "Towards a Combination of Online and Multitask Learning for MT Quality Estimation: a Preliminary Study." *Proceedings of Workshop on Interactive and Adaptive Machine Translation in 2014 (IAMT 2014)*. Vancouver, BC, Canada. **To appear.**

7. Turchi, Marco, Antonios Anastasopoulos, José G. C. de Souza, and Matteo Negri. 2014a. "Adaptive Quality Estimation for Machine Translation." *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 710–720. Baltimore, Maryland

8. C. de Souza, José G., Jesús González-Rubio, Christian Buck, Marco Turchi, and Matteo Negri. 2014a. "FBK-UPV-UEdin participation in the WMT14 Quality Estimation shared-

task." *Proceedings of the Ninth Workshop on Statistical Machine Translation*, 322–328. Baltimore, Maryland, USA

9. Turchi, Marco and Matteo Negri. 2014. "Automatic Annotation of Machine Translation Datasets with Binary Quality Judgements." *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, 1788–1792. Reykjavik, Iceland

10. Shah, Kashif, Marco Turchi, and Lucia Specia. 2014. "An efficient and user-friendly tool for machine translation quality estimation." *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. Reykjavik, Iceland

11. Buck, Christian, Kenneth Heafield, and Bas van Ooyen. 2014. "N-gram Counts and Language Models from the Common Crawl." *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, 3579–3584. Reykjavik, Iceland

# Identification of Bilingual Terms from Monolingual Documents for Statistical Machine Translation

**Mihael Arcan**[1] **Claudio Giuliano**[2] **Marco Turchi**[2] **Paul Buitelaar**[1]

[1] Unit for Natural Language Processing, Insight @ NUI Galway, Ireland
{mihael.arcan , paul.buitelaar}@insight-centre.org
[2] FBK - Fondazione Bruno Kessler, Via Sommarive 18, 38123 Trento, Italy
{giuliano, turchi}@fbk.eu

## Abstract

The automatic translation of domain-specific documents is often a hard task for generic Statistical Machine Translation (SMT) systems, which are not able to correctly translate the large number of terms encountered in the text. In this paper, we address the problems of automatic identification of bilingual terminology using Wikipedia as a lexical resource, and its integration into an SMT system. The correct translation equivalent of the disambiguated term identified in the monolingual text is obtained by taking advantage of the multilingual versions of Wikipedia. This approach is compared to the bilingual terminology provided by the Terminology as a Service (TaaS) platform. The small amount of high quality domain-specific terms is passed to the SMT system using the XML markup and the Fill-Up model methods, which produced a relative translation improvement up to 13% BLEU score points

## 1 Introduction

Translation tasks often need to deal with domain-specific terms in technical documents, which require specific lexical knowledge of the domain. Nowadays, SMT systems are suitable to translate very frequent expressions but fail in translating domain-specific terms. This mostly depends on a lack of domain-specific parallel data from which the SMT systems can learn. Translation tools such as Google Translate or open source phrase-based SMT systems, trained on generic data, are the most common solutions and they are often used to translate manuals or very specific texts, resulting in unsatisfactory translations.

This problem is particular relevant for professional translators that work with documents coming from different domains and are supported by generic SMT systems. A valuable solution to help them in handling domain-specific terms is represented by online terminology resources, e.g. IATE - Inter-Active Terminology for Europe,[1] which are continuously updated and can be easily queried. However, the manual use of these services can be very time demanding. For this reason, the identification and embedding of domain-specific terms in an SMT system is a crucial step towards increasing translator productivity and translation quality in highly specific domains.

In this paper, we propose an approach to automatically detect monolingual domain-specific terms from a source language document and identify their equivalents using Wikipedia cross-lingual links. For this purpose we extend The Wiki Machine API,[2] a tool for linking terms in text to Wikipedia pages, adding two more components able to first identify domain-specific terms, and to find their translations in a target language. The identified bilingual terms are then compared with those obtained by TaaS (Skadinš et al., 2013). The embedding of the domain-specific terms into an SMT system is performed by use of the XML markup approach, which uses the terms as preferred translation candidates at run time, and the Fill-Up model (Bisazza et al., 2011), which emphasizes phrase pairs extracted from the bilingual terms.

Our results show that the performance of our technique and TaaS are comparable in the identification of monolingual and bilingual domain-specific terms. From the machine translation point of view, our experiments highlight the benefit of integrating bilingual terms into the SMT system, and the relative improvement in BLEU score of the Fill-Up model over the baseline and the XML markup approach.



[1] http://iate.europa.eu/ [2] https://bitbucket.org/fbk/thewikimachine/



*Proceedings of the 4th International Workshop on Computational Terminology*, pages 22–31,
Dublin, Ireland, August 23 2014.

# Enhancing Statistical Machine Translation with Bilingual Terminology in a CAT Environment

**Mihael Arcan** mihael.arcan@insight-centre.com
Insight @ National University of Ireland, Galway, Galway, Ireland

**Marco Turchi** turchi@fbk.eu
**Sara Tonelli** satonelli@fbk.eu
FBK - Fondazione Bruno Kessler, Via Sommarive 18, 38123 Trento, Italy

**Paul Buitelaar** paul.buitelaar@insight-centre.com
Insight @ National University of Ireland, Galway, Galway, Ireland

**Abstract**

In this paper, we address the problem of extracting and integrating bilingual terminology into a Statistical Machine Translation (SMT) system for a Computer Aided Translation (CAT) tool scenario. We develop a framework that, taking as input a small amount of parallel in-domain data, gathers domain-specific bilingual terms and injects them in an SMT system to enhance the translation productivity. Therefore, we investigate several strategies to extract and align bilingual terminology, and to embed it into the SMT. We compare two embedding methods that can be easily used at run-time without altering the normal activity of an SMT system: XML markup and the cache-based model. We tested our framework on two different domains showing improvements up to 15% BLEU score points.

## 1 Introduction

Recent studies (Federico et al., 2012; Läubli et al., 2013; Green et al., 2013) have shown significant productivity gains when human translators post-edit machine translation output rather than translating documents from scratch. This evidence has raised interest in the integration of machine translation systems within CAT software. In this context, an important open issue is how to support translators with domain-specific information when dealing with highly specific texts, i.e. manuals coming from different domains (information technology (IT), legal, agriculture, etc.). Translation tools such as Google Translate,[1] Bing Translator[2] or open source SMT systems such as Moses (Koehn et al., 2007) trained on generic data are the most common solutions, but they often result in unsatisfactory translations. A valuable alternative to support professional translators is represented by online terminology resources, e.g. IATE,[3] which are continuously updated and can be easily queried. However, the manual use of these services can be very time demanding when working with a CAT tool. For these reasons, the automatic identification and integration of bilingual domain-specific terms into an SMT system is a crucial step towards increasing translation quality of high-specific texts in a CAT environment. This also reduces translators' initial overload when dealing with different domains, because terminological lists are managed directly by the SMT system and no additional human intervention for retrieving domain-specific terminology is required.

[1] http://translate.google.com/ [2] http://www.bing.com/translator
[3] Inter-Active Terminology for Europe, http://iate.europa.eu/

**Noname manuscript No.**
(will be inserted by the editor)

# Data-driven Annotation of Binary MT Quality Estimation Corpora Based on Human Post-editions

**Marco Turchi · Matteo Negri · Marcello Federico**

the date of receipt and acceptance should be inserted later

**Abstract** Advanced computer-assisted translation (CAT) tools include automatic quality estimation (QE) mechanisms to support post-editors in identifying and selecting useful suggestions. Based on supervised learning techniques, QE relies on high-quality data annotations obtained from expensive manual procedures. However, as the notion of MT quality is inherently subjective, such procedures might result in unreliable or uninformative annotations. To overcome these issues, we propose an automatic method to obtain binary annotated data that explicitly discriminate between useful (suitable for post-editing) and useless suggestions. Our approach is fully data-driven and bypasses the need for explicit human labelling. Experiments with different language pairs and domains demonstrate that it yields better models than those based on the adaptation into binary datasets of the available QE corpora. Furthermore, our analysis suggests that the learned thresholds separating useful from useless translations are significantly lower than as suggested in the existing guidelines for human annotators. Finally, a verification experiment with several translators operating with a CAT tool confirms our empirical findings.

**Keywords** Statistical MT · Quality estimation · Productivity · Use of post-editing data

M. Turchi
Fondazione Bruno Kessler, Povo - Trento, Italy
E-mail: turchi@fbk.eu

M. Negri
Fondazione Bruno Kessler, Povo - Trento, Italy
E-mail: negri@fbk.eu

M. Federico
Fondazione Bruno Kessler, Povo - Trento, Italy
E-mail: federico@fbk.eu

# Assessing the Impact of Translation Errors on Machine Translation Quality with Mixed-effects Models

**Marcello Federico, Luisa Bentivogli, Matteo Negri, Marco Turchi**
FBK - Fondazione Bruno Kessler
Via Sommarive 18, 38123 Trento, Italy
{federico,bentivogli,negri,turchi}@fbk.eu

## Abstract

Learning from errors is a crucial aspect of improving expertise. Based on this notion, we discuss a robust statistical framework for analysing the impact of different error types on machine translation (MT) output quality. Our approach is based on linear mixed-effects models, which allow the analysis of error-annotated MT output taking into account the variability inherent to the specific experimental setting from which the empirical observations are drawn. Our experiments are carried out on different language pairs involving Chinese, Arabic and Russian as target languages. Interesting findings are reported, concerning the impact of different error types both at the level of human perception of quality and with respect to performance results measured with automatic metrics.

## 1 Introduction

The dominant statistical approach to machine translation (MT) is based on learning from large amounts of parallel data and tuning the resulting models on reference-based metrics that can be computed automatically, such as BLEU (Papineni et al., 2001), METEOR (Banerjee and Lavie, 2005), TER (Snover et al., 2006), GTM (Turian et al., 2003). Despite the steady progress in the last two decades, especially for few well resourced translation directions having English as target language, this way to approach the problem is quickly reaching a performance plateau. One reason is that parallel data are a source of reliable information but, alone, limit systems knowledge to observed positive examples (*i.e.* how a sentence should be translated) without explicitly modelling any notion of error (*i.e.* how a sentence should *not* be translated). Another reason is that, as a development and evaluation criterion, automatic metrics provide a holistic view of systems' behaviour without identifying the specific issues of a translation. Indeed, the global scores returned by MT evaluation metrics depend on comparisons between translation hypotheses and reference translations, where the causes and the nature of the differences between them are not identified.

To cope with these issues and define system improvement priorities, the focus of MT evaluation research is gradually shifting towards profiling systems' behaviour with respect to various typologies of errors (Vilar et al., 2006; Popović and Ney, 2011; Farrús et al., 2012, *inter alia*). This shift has enriched the traditional MT evaluation framework with a new element, that is the actual errors done by a system. Until now, most of the research has focused on the relationship (*i.e.* the correlation) between two elements of the framework: humans and automatic evaluation metrics. As a new element of the framework, which becomes a sort of "evaluation triangle", the analysis of error annotations opens interesting research problems related to the relationships between: *i)* error types and human perception of MT quality and *ii)* error types and the sensitivity of automatic metrics.

Besides motivating further investigation on metrics featuring high correlation with human judgements (a well-established MT research sub-field, which is out of the scope of this paper), connecting the vertices of this triangle raises new challenging questions such as:

**(1) Which types of MT errors have the highest impact on human perception of translation quality?** Surprisingly, little prior work focused on this side of the triangle. Error annotations have been considered to highlight strengths and weaknesses of MT engines or to investigate the influence of different error types on post-editors' work. However, the direct connection between er-

# Machine Translation Quality Estimation Across Domains

**José G. C. de Souza**
University of Trento
Fondazione Bruno Kessler
Trento, Italy
desouza@fbk.eu

**Marco Turchi**
Fondazione Bruno Kessler
Trento, Italy
turchi@fbk.eu

**Matteo Negri**
Fondazione Bruno Kessler
Trento, Italy
negri@fbk.eu

## Abstract

Machine Translation (MT) Quality Estimation (QE) aims to automatically measure the quality of MT system output without reference translations. In spite of the progress achieved in recent years, current MT QE systems are not capable of dealing with data coming from different train/test distributions or domains, and scenarios in which training data is scarce. We investigate different multitask learning methods that can cope with such limitations and show that they overcome current state-of-the-art methods in real-world conditions where training and test data come from different domains.

## 1 Introduction

Machine Translation (MT) Quality Estimation (QE) aims to automatically predict the quality of MT output without using reference translations (Blatz et al., 2003; Specia et al., 2009). QE systems usually employ supervised machine learning models that use different information extracted from (source, target) sentence pairs as features along with quality scores as labels. The notion of quality that these models measure can be indicated by different scores. Some examples are the average number of edits required to post-edit the MT output, i.e., human translation edit rate[1] (HTER (Snover et al., 2006)), and the time (in seconds) required to post-edit a translation produced by an MT system (Specia, 2011).

Research on QE has received a strong boost in recent years due to the increase in the usage of MT systems in real-world applications. Automatic and reference-free MT quality prediction demonstrated to be useful for different applications, such as: deciding whether the translation output can be published without post-editing (Soricut and Echihabi, 2010), filtering out low-quality translation suggestions that should be rewritten from scratch (Specia et al., 2009), selecting the best translation output from a pool of MT systems (Specia et al., 2010), and informing readers of the translation whether it is reliable or not (Turchi et al., 2012). Another example is the computer-assisted translation (CAT) scenario, in which it might be necessary to predict the quality of translation suggestions generated by different MT systems to support the activity of post editors working with different genres of text.

The dominant QE framework presents some characteristics that can limit models' applicability in such real-world scenarios. First, the scores used as training labels (HTER, time) are costly to obtain because they are derived from manual post-editions of MT output. Such requirement makes it difficult to develop models for domains in which there is a limited amount of labeled data. Second, the learning methods currently used (for instance in the framework of QE shared evaluation campaigns)[2] assume that training and test data are sampled from the same distribution. Though reasonable as a first evaluation setting to promote research in the field, this controlled scenario is not realistic as different data in real-world applications might be post-edited by different translators, the translations might be generated by different MT systems and the documents being translated might belong to different domains or genres. To



[1] Edit distance is calculated as the number of edits (word insertions, deletions, substitutions, and shifts) divided by the number of words in the reference. Lower HTER values indicate better translations.

[2] In the last two editions of the yearly Workshop on Machine Translation, several QE shared tasks have been proposed (Callison-Burch et al., 2012; Bojar et al., 2013).



*Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 409–420, Dublin, Ireland, August 23-29 2014.

# Towards a Combination of Online and Multitask Learning for MT Quality Estimation: a Preliminary Study

**José G. C. de Souza**[(1,2)] desouza@fbk.eu
**Marco Turchi**[(1)] turchi@fbk.eu
**Matteo Negri**[(1)] negri@fbk.eu
[(1)] FBK - Fondazione Bruno Kessler, Trento, Italy
[(2)] University of Trento, Italy



**Abstract**

Quality estimation (QE) for machine translation has emerged as a promising way to provide real-world applications with methods to estimate at run-time the reliability of automatic translations. Real-world applications, however, pose challenges that go beyond those of current QE evaluation settings. For instance, the heterogeneity and the scarce availability of training data might contribute to significantly raise the bar. To address these issues we compare two alternative machine learning paradigms, namely *online* and *multi-task* learning, measuring their capability to overcome the limitations of current batch methods. The results of our experiments, which are carried out in the same experimental setting, demonstrate the effectiveness of the two methods and suggest their complementarity. This indicates, as a promising research avenue, the possibility to combine their strengths into an online multi-task approach to the problem.

## 1 Introduction

Quality Estimation (QE) for Machine Translation (MT) is the task of estimating the quality of a translated sentence at run-time and without access to reference translations (Specia et al., 2009).

As a quality indicator, in a typical QE setting, automatic systems have to predict either the time or the number of editing operations (e.g. in terms of HTER[1]) required to a human to transform the machine-translated sentence into an adequate and fluent translation. In recent years, QE gained increasing interest in the MT community as a possible way to: decide whether a given translation is good enough for publishing as is, inform readers of the target language only whether or not they can rely on a translation, filter out sentences that are not good enough for post-editing by professional translators, or select the best translation among options from multiple MT or translation memory systems.

So far, despite its many possible applications, QE research has been mainly conducted in controlled lab testing scenarios that disregard some of the possible challenges posed by real working conditions. Indeed, the large body of research resulting from three editions of the shared QE task organized within the yearly Workshop on Machine Translation (WMT (Callison-Burch et al., 2012; Bojar et al., 2013, 2014)) has relied on simplistic assumptions

[1]The HTER (Snover et al., 2006) measures the minimum edit distance between the MT output and its manually post-edited version. Edit distance is calculated as the number of edits (word insertions, deletions, substitutions, and shifts) divided by the number of words in the reference. Lower HTER values indicate better translations.

# Adaptive Quality Estimation for Machine Translation

**Marco Turchi**[1] **Antonios Anastasopoulos**[3]
**José G. C. de Souza**[1,2] **Matteo Negri**[1]
[1] FBK - Fondazione Bruno Kessler, Via Sommarive 18, 38123 Trento, Italy
[2] University of Trento, Italy
[3] National Technical University of Athens, Greece
{turchi,desouza,negri}@fbk.eu
anastasopoulos.ant@gmail.com

## Abstract

The automatic estimation of machine translation (MT) output quality is a hard task in which the selection of the appropriate algorithm and the most predictive features over reasonably sized training sets plays a crucial role. When moving from controlled lab evaluations to real-life scenarios the task becomes even harder. For current MT quality estimation (QE) systems, additional complexity comes from the difficulty to model user and domain changes. Indeed, the instability of the systems with respect to data coming from different distributions calls for adaptive solutions that react to new operating conditions. To tackle this issue we propose an online framework for adaptive QE that targets reactivity and robustness to user and domain changes. Contrastive experiments in different testing conditions involving user and domain changes demonstrate the effectiveness of our approach.

## 1 Introduction

After two decades of steady progress, research in statistical machine translation (SMT) started to cross its path with translation industry with tangible mutual benefit. On one side, SMT research brings to the industry improved output quality and a number of appealing solutions useful to increase translators' productivity. On the other side, the market needs suggest concrete problems to solve, providing real-life scenarios to develop and evaluate new ideas with rapid turnaround. The evolution of computer-assisted translation (CAT) environments is an evidence of this trend, shown by the increasing interest towards the integration of suggestions obtained from MT engines with those derived from translation memories (TMs).

The possibility to speed up the translation process and reduce its costs by post-editing good-quality MT output raises interesting research challenges. Among others, these include deciding *what* to present as a suggestion, and *how* to do it in the most effective way.

In recent years, these issues motivated research on automatic QE, which addresses the problem of estimating the quality of a translated sentence given the source and without access to reference translations (Blatz et al., 2003; Specia et al., 2009; Mehdad et al., 2012). Despite the substantial progress done so far in the field and in successful evaluation campaigns (Callison-Burch et al., 2012; Bojar et al., 2013), focusing on concrete market needs makes possible to further define the scope of research on QE. For instance, moving from controlled lab testing scenarios to real working environments poses additional constraints in terms of adaptability of the QE models to the variable conditions of a translation job. Such variability is due to two main reasons:

1. **The notion of MT output quality is highly subjective** (Koponen, 2012; Turchi et al., 2013; Turchi and Negri, 2014). Since the quality standards of individual users may vary considerably (*e.g.* according to their knowledge of the source and target languages), the estimates of a static QE model trained with data collected from a group of post-editors might not fit with the actual judgements of a new user;

2. **Each translation job has its own specificities** (domain, complexity of the source text, average target quality). Since data from a new job may differ from those used to train the QE model, its estimates on the new instances might result to be biased or uninformative.

The ability of a system to self-adapt to the be-



*Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 710–720,
Baltimore, Maryland, USA, June 23-25 2014. 

# FBK-UPV-UEdin participation in the WMT14 Quality Estimation shared-task

**José G. C. de Souza***
University of Trento
Fondazione Bruno Kessler
Trento, Italy
desouza@fbk.eu

**Jesús González-Rubio***
PRHLT Group
U. Politècnica de València
Valencia, Spain
jegonzalez@prhlt.upv.es

**Christian Buck***
University of Edinburgh
School of Informatics
Edinburgh, Scotland, UK
cbuck@lantis.de

**Marco Turchi, Matteo Negri**
Fondazione Bruno Kessler
turchi,negri@fbk.eu

## Abstract

This paper describes the joint submission of Fondazione Bruno Kessler, Universitat Politècnica de València and University of Edinburgh to the Quality Estimation tasks of the Workshop on Statistical Machine Translation 2014. We present our submissions for Task 1.2, 1.3 and 2. Our systems ranked first for Task 1.2 and for the Binary and Level1 settings in Task 2.

## 1 Introduction

Quality Estimation (QE) for Machine Translation (MT) is the task of evaluating the quality of the output of an MT system without reference translations. Within the WMT 2014 QE Shared Task four evaluation tasks were proposed, covering both word and sentence level QE. In this work we describe the Fondazione Bruno Kessler (FBK), Universitat Politècnica de València (UPV) and University of Edinburgh (UEdin) approach and system setup for the shared task.

We developed models for two sentence-level tasks and for the word-level task: Task 1.2, scoring for post-editing effort, and Task 1.3, predicting post-editing time and Task 2, binary and multi-class classification for word-level QE. As opposed to previous editions of the shared task, this year the participants were not supplied with the MT system that was used to produce the translation. Furthermore no system-internal features were provided. Thus, while the trained models are tuned to detect the errors of a specific system the features have to be generated independently (black-box).

## 2 Sentence Level QE

We submitted runs to two sentence-level tasks: Task 1.2 and Task 1.3. The first task aims at predicting the Human mediated Translation Edit Rate (HTER) (Snover et al., 2006) between a suggestion generated by a machine translation system and its manually post-edited version. The data set contains 1,104 English-Spanish sentence pairs post-edited by one translator (896 for training and 208 for test). The second task requires to predict the time, in miliseconds, that was required to post edit a translation given by a machine translation system. Participants are provided with 858 English-Spanish sentence pairs, source and suggestion, along with their respective post-edited sentence and post-editing time in seconds (650 data points for training and 208 for test). We participated in the scoring mode of both tasks.

*Contributed equally to this work.

### 2.1 Features

For our sentence-level submissions we compute features using different resources that do not use the MT system internals. We use the same set of features for both Task 1.2 and 1.3.

**QuEst Black-box features (quest79).** We extract 79 black-box features that capture the complexity, fluency and adequacy aspects of the QE problem. These features are extracted using the implementation provided by the QuEst framework (Specia et al., 2013). Among them are the 17 baseline features provided by the task organizers.

The **complexity** features are computed on the source sentence and indicate the complexity of translating the segment. Examples of these features are the language model (LM) probabilities of the source sentence computed in a corpus of the source language, different surface counts like the number of punctuation marks and the number of tokens in the source sentence, among others.

The **fluency** features are computed over the translation generated by the MT system and indicate how fluent the translation is in the target

# Automatic Annotation of Machine Translation Datasets with Binary Quality Judgements

**Marco Turchi, Matteo Negri**

FBK, Fondazione Bruno Kessler
38123 Povo, Trento, Italy
{turchi,negri}@fbk.eu

**Abstract**

The automatic estimation of machine translation (MT) output quality is an active research area due to its many potential applications (*e.g.* aiding human translation and post-editing, re-ranking MT hypotheses, MT system combination). Current approaches to the task rely on supervised learning methods for which high-quality labelled data is fundamental. In this framework, quality estimation (QE) has been mainly addressed as a regression problem where models trained on *(source, target)* sentence pairs annotated with continuous scores (in the [0-1] interval) are used to assign quality scores (in the same interval) to unseen data. Such definition of the problem assumes that continuous scores are informative and easily interpretable by different users. These assumptions, however, conflict with the subjectivity inherent to human translation and evaluation. On one side, the subjectivity of human judgements adds noise and biases to annotations based on scaled values. This problem reduces the usability of the resulting datasets, especially in application scenarios where a sharp distinction between "good" and "bad" translations is needed. On the other side, continuous scores are not always sufficient to decide whether a translation is actually acceptable or not. To overcome these issues, we present an automatic method for the annotation of *(source, target)* pairs with *binary* judgements that reflect an empirical, and easily interpretable notion of quality. The method is applied to annotate with binary judgements three QE datasets for different language combinations. The three datasets are combined in a single resource, called BinQE, which can be freely downloaded from *http://hlt.fbk.eu/technologies/binqe*.
**Keywords:** Machine Translation, Quality Estimation, Corpus Annotation.

## 1. Introduction

In the machine translation (MT) field, quality estimation (QE) is the task of determining the quality of an automatic translation given its source sentence (Specia et al., 2009; Soricut and Echihabi, 2010; Bach et al., 2011; Specia, 2011; Mehdad et al., 2012). Differently from standard MT evaluation methods that rely on metrics such as BLEU (Papineni et al., 2002), QE aims to return predictions for unseen translated sentences without relying on reference translations. This makes QE particularly suitable from an application-oriented perspective, since in many situations the quality of automatic translations has to be measured at run-time and without the support of external, manually created benchmarks.

Among the many potential applications, quality estimates are extremely useful in computer-assisted translation (CAT) where, for each segment of a source document, human translators are presented with suggestions obtained from a translation memory (TM) or an MT engine. In both cases, to enhance translators' productivity, *useful* suggestions (whose correction requires less effort than re-translation from scratch) should be promoted, and the *useless* ones should be demoted or automatically filtered out.

While TM suggestions are typically accompanied by *fuzzy match* scores (indicating the amount of overlap between the source sentence and previously translated segments stored in the translation memory), MT outputs can be presented with different quality indicators that account for their reliability. Such indicators typically consist in *effort score* labels (indicating the expected amount of post-editing needed by a human to correct a translation into a publishable material), or *time* estimates (indicating the expected time in seconds needed for the correction). Besides the fact that these quality indicators are not comparable with fuzzy match scores,[1] the idea that translation quality can be represented with scaled values raises other issues related to their use and interpretation in the CAT framework.

The first problem is that quality judgements are subjective (Koponen, 2012; Turchi et al., 2014), as also evidenced by the low inter-annotator agreement on the available datasets (Callison-Burch et al., 2012). Since different annotators often produce different quality scores for the same *(source, target)* pair, the resulting datasets are usually affected by various levels of noise and bias in labels' distribution. This issue complicates the task of learning reliable QE models and can drastically reduce their usability.

Another problem, also related to the subjectivity of human judgements, is that continuous quality scores are not easily interpretable. For instance, a *0.49 effort score* does not say much about the actual quality of a translation, nor about how different users will perceive it.

An intuitive solution to make QE judgements suitable for practical use is to approach the problem as a binary task. Our hypothesis is that, especially for some applications such as CAT technology, QE models trained on binary datasets would make possible to obtain quality judgements that are more informative and easily interpretable than continuous scores. To this aim, the existing datasets can be transformed into binary datasets by setting a threshold that discriminates between "good" and "bad" translations. Following such thresholding method, instances with an effort score (or time in seconds) above the threshold would be marked as *bad* (*i.e.* useless) while those below the threshold would be marked as *good* (*i.e.* useful). This strategy,

[1] This problem is out of the scope of our investigation, which exclusively focuses on making QE judgements more suitable for practical use in binary tasks.

# An efficient and user-friendly tool for machine translation quality estimation

**Kashif Shah**[§], **Marco Turchi**[†], **Lucia Specia**[§]

§Department of Computer Science, University of Sheffield, UK
{kashif.shah,l.specia}@sheffield.ac.uk
†Fondazione Bruno Kessler, University of Trento, Italy
turchi@fbk.eu

**Abstract**

We present a new version of QUEST – an open source framework for machine translation quality estimation – which brings a number of improvements: (i) it provides a Web interface and functionalities such that non-expert users, e.g. translators or lay-users of machine translations, can get quality predictions (or internal features of the framework) for translations without having to install the toolkit, obtain resources or build prediction models; (ii) it significantly improves over the previous runtime performance by keeping resources (such as language models) in memory; (iii) it provides an option for users to submit the source text only and automatically obtain translations from Bing Translator; (iv) it provides a ranking of multiple translations submitted by users for each source text according to their estimated quality. We exemplify the use of this new version through some experiments with the framework.

**Keywords:** Machine Translation, Translation Evaluation, Translation Quality Estimation

## 1. Introduction

Metrics to predict the quality of texts translated automatically by Machine Translation (MT) systems have become a necessity in many scenarios. These metrics, referred to as quality estimation (QE), or also *confidence estimation*, are aimed at MT systems in use. They consist in prediction models generally built using supervised machine learning algorithms from examples of source texts and their machine translations (i.e., no access to reference translations) described through a number of features and labelled for quality. The notion of "quality" in QE metrics is defined according to the application and represented by labels – post-editing effort, gisting reliability, etc. – and features – for example, a binary grammar checker feature will be important for fluency prediction, but less useful for gisting reliability prediction.

A number of positive results have been reported in recent work in the field. Examples include improving post-editing efficiency by filtering out low quality segments which would require more effort or time to correct than translating from scratch (Specia et al., 2009; Specia, 2011), selecting high quality segments to be published as they are, without post-editing (Soricut and Echihabi, 2010), selecting a translation from either an MT system or a translation memory for post-editing (He et al., 2010), selecting the best translation from multiple MT systems (Specia et al., 2010; Avramidis, 2013), and highlighting sub-segments that need revision (Bach et al., 2011). For recent overviews of various algorithms and features we refer the reader to the WMT12-13 editions of the shared task on QE (Callison-Burch et al., 2012; Bojar et al., 2013).

QUEST (Specia et al., 2013) is an open-source framework for QE which provides a wide range of feature extractors from source and translation texts, as well as external resources and tools. These lead to an average of 150 features (depending on the language pair) and go from simple, language-independent features, to advanced, linguistically motivated features. They include features that rely on information from the MT system that generated the translations, and features that are oblivious to the way translations were produced. In addition, QUEST integrates a well-known machine learning toolkit, `scikit-learn`,[1] and other algorithms that are known to perform well on this task, facilitating experiments with existing features and techniques for feature selection and model building. QUEST also provides documentation for users to add their own features and learning algorithms. However, QUEST is not directly usable by end-users, such as professional translators. The tasks of installing and configuring the toolkit, obtaining the necessary resources, and building new models from data require technical knowledge of natural language processing, machine translation and machine learning.

In this paper we describe a number of improvements over the current version of QUEST which are meant to make it more accessible to non-expert users, as well as more efficient (i.e., faster). In particular, we provide

[1] `http://scikit-learn.org/`

# $N$-gram Counts and Language Models from the Common Crawl

**Christian Buck**[†], **Kenneth Heafield**[‡], **Bas van Ooyen**[*]
[†]University of Edinburgh, Edinburgh, Scotland
[‡]Stanford University, Stanford, CA, USA
[*] Owlin BV, Utrecht, Netherlands
christian.buck@ed.ac.uk, heafield@cs.stanford.edu, bas@owlin.com

**Abstract**

We contribute 5-gram counts and language models trained on the Common Crawl corpus, a collection over 9 billion web pages. This release improves upon the Google $n$-gram counts in two key ways: the inclusion of low-count entries and deduplication to reduce boilerplate. By preserving singletons, we were able to use Kneser-Ney smoothing to build large language models. This paper describes how the corpus was processed with emphasis on the problems that arise in working with data at this scale. Our unpruned Kneser-Ney English 5-gram language model, built on 975 billion deduplicated tokens, contains over 500 billion unique $n$-grams. We show gains of 0.5–1.4 BLEU by using large language models to translate into various languages.

**Keywords:** web corpora, language models, multilingual

## 1. Introduction

The sheer amount of data in multiple languages makes web-scale corpora attractive for many natural language processing tasks. Of particular importance is language modeling, where web-scale language models have been shown to improve machine translation and automatic speech recognition performance (Brants et al., 2007; Chelba and Schalkwyk, 2013; Guthrie and Hepple, 2010). In this work, we contribute $n$-gram counts and language models trained on the Common Crawl corpus.[1]

Google has released $n$-gram counts (Brants and Franz, 2006) trained on one trillion tokens of text. However, they pruned any $n$-grams that appeard less than 40 times. Moreover, all words that appeared less than 200 times were replaced with the unknown word. Both forms of pruning make the counts unsuitable for estimating a language model with the popular and successful Kneser-Ney smoothing algorithm, which requires unpruned counts even if the final model is to be pruned.

The second issue with the publicly available Google $n$-gram counts (Brants and Franz, 2006) is that the training data was not deduplicated, so boilerplate such as copyright notices has unreasonably high counts (Lin et al., 2010). Google has shared a deduplicated version (Bergsma et al., 2010) in limited contexts (Lin et al., 2010), but it was never publicly released (Lin, 2013). Our training data was deduplicated before counting $n$-grams.

Microsoft provides a web service (Wang et al., 2010) that can be queried for language model probabilities. The service is currently limited to the English language whereas we provide models for many languages. Moreover, an initial experiment on reranking English machine translation output led to so many queries that the service went down several times, despite client-side caching. Using the Microsoft service during machine translation decoding would entail far more queries and require lower latency.

[1] http://statmt.org/ngrams

## 2. Data Preparation

The Common Crawl[2] is a publicly available crawl of the web. We use the 2012, early 2013, and "winter" 2013 crawls, consisting of 3.8 billion, 2 billion, and 2.3 billion pages, respectively. Because both 2013 crawls are similar in terms of seed addresses and distribution of top-level domains in this work we only distinguish 2012 and 2013 crawls.

The data is made available both as raw HTML and as text only files. The latter collection consists of all HTML and RSS files from which all tags were stripped. The HTML comes in the original encoding, while the text has been converted to UTF-8, albeit with the occasional invalid character.

Using the HTML files has the advantage of being able to exploit the document structure to select paragraphs and to tell boilerplate from actual content. However, parsing such large amounts of HTML is non-trivial and requires many normalization steps.

In this work we focus on processing the text only files which we downloaded and processed locally on a small cluster. The advantages of structured text do not outweigh the extra computing power needed to process them.

### 2.1. Language Detection

The first step in our pipeline is splitting the data by language. We explored the option of automatically detecting the main language for every page but found that mixed-language content is quite common. By using the Compact Language Detector 2 (CLD2)[3] we are able to partition every document into monolingual spans. CLD2 is able to detect 175 languages and fast enough to process the entire corpus within a week.

Table 1 shows the relative contribution of the most common languages in the separated data. At this stage of processing we have no meaningful notion of token or line counts and therefore report the size of the extracted files. As

[2] http://commoncrawl.org/
[3] https://code.google.com/p/cld2/

# References

Aker, Ahmet, Monica Paramita, and Robert Gaizauskas. 2013. "Extracting bilingual terminologies from comparable corpora." *Proceedings of ACL*. Sofia, Bulgaria.

Arcan, Mihael, Claudio Giuliano, Marco Turchi, and Paul Buitelaar. 2014a. "Identification of Bilingual Terms from Monolingual Documents for Statistical Machine Translation." *Proceedings of the 4th International Workshop on Computational Terminology (Computerm)*, 22–31. Dublin, Ireland.

Arcan, Mihael, Marco Turchi, Sara Tonelli, and Paul Buitelaar. 2014b. "Enhancing Statistical Machine Translation with Bilingual Terminology in a CAT Environment." *Proceedings of the 11th Biennial Conference of the Association for Machine Translation in the Americas (AMTA 2014)*. Vancouver, BC, Canada. **To appear.**

Baayen, R Harald, Douglas J Davidson, and Douglas M Bates. 2008. "Mixed-effects modeling with crossed random effects for subjects and items." *Journal of memory and language*, 59(4):390–412.

Bentivogli, Luisa, Pamela Forner, Bernardo Magnini, and Emanuele Pianta. 2004. "Revising the wordnet domains hierarchy: semantics, coverage and balancing." *Proceedings of the Workshop on Multilingual Linguistic Ressources*, 101–108.

Bertoldi, Nicola, Mauro Cettolo, and Marcello Federico. 2013. "Cache-based Online Adaptation for Machine Translation Enhanced Computer Assisted Translation." *Proceedings of Machine Translation Summit XIV*. Nice, France.

Bisazza, Arianna, Nick Ruiz, and Marcello Federico. 2011. "Fill-up versus interpolation methods for phrase-based smt adaptation." *Proceedings of IWSLT*.

Blatz, John, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2003. *Confidence Estimation for Machine Translation*. Summer workshop final report, JHU/CLSP.

Blatz, John, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. "Confidence estimation for machine translation." *Proceedings of the international conference on Computational Linguistics*, 315–321.

Bojar, Ondrej, Christian Buck, Chris Callison-Burch, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2013. "Findings

of the 2013 Workshop on Statistical Machine Translation." *Eighth Workshop on Statistical Machine Translation*, WMT-2013, 1–44. Sofia, Bulgaria.

Bojar, Ondrej, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Aleš Tamchyna. 2014. "Findings of the 2014 workshop on statistical machine translation." *Proceedings of the Ninth Workshop on Statistical Machine Translation*, 12–58. Baltimore, Maryland, USA.

Bouamor, Dhouha, Nasredine Semmar, and Pierre Zweigenbaum. 2012. "Identifying bilingual multi-word expressions for statistical machine translation." *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. Istanbul, Turkey.

Brown, Peter F., Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. "The mathematics of statistical machine translation: parameter estimation." *Computational Linguistics*, 19:263–311.

Buck, Christian, Kenneth Heafield, and Bas van Ooyen. 2014. "N-gram Counts and Language Models from the Common Crawl." *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, 3579–3584. Reykjavik, Iceland.

C. de Souza, José G., Christian Buck, Marco Turchi, and Matteo Negri. 2013a. "FBK-UEdin Participation to the WMT13 Quality Estimation Shared-Task." *Proceedings of the 8th Workshop on Statistical Machine Translation (WMT'13)*. Sofia, Bulgaria.

C. de Souza, José G., Miquel Esplà-Gomis, Marco Turchi, and Matteo Negri. 2013b. "Exploiting Qualitative Information from Automatic Word Alignment for Cross-lingual NLP Tasks." *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 771–776. Sofia, Bulgaria.

C. de Souza, José G., Jesús González-Rubio, Christian Buck, Marco Turchi, and Matteo Negri. 2014a. "FBK-UPV-UEdin participation in the WMT14 Quality Estimation shared-task." *Proceedings of the Ninth Workshop on Statistical Machine Translation*, 322–328. Baltimore, Maryland, USA.

C. de Souza, José G., Marco Turchi, and Matteo Negri. 2014b. "Towards a Combination of Online and Multitask Learning for MT Quality Estimation: a Preliminary Study." *Proceedings of Workshop on Interactive and Adaptive Machine Translation in 2014 (IAMT 2014)*. Vancouver, BC, Canada. **To appear.**

C. de Souza, José G., Marco Turchi, and Matteo Negri. 2014c. "Machine Translation Quality Estimation Across Domains." *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 409–420. Dublin, Ireland.

Callison-Burch, Chris, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2012. "Findings of the 2012 Workshop on Statistical Machine Translation." *Proceedings of the Seventh Workshop on Statistical Machine Translation (WMT'12)*, 10–51. Montréal, Canada.

Cettolo, Mauro, Christian Girardi, and Marcello Federico. 2012. "Wit$^3$: Web inventory of transcribed and translated talks." *Proceedings of the 16$^{th}$ Conference of the European Association for Machine Translation (EAMT)*, 261–268. Trento, Italy.

Chen, Jianhui, Ji Liu, and Jieping Ye. 2012. "Learning incoherent sparse and low-rank patterns from multiple tasks." *ACM Transactions on Knowledge Discovery from Data*, 5(4):22.

Chen, Jianhui, Jiayu Zhou, and Jieping Ye. 2011. "Integrating low-rank and group-sparse structures for robust multi-task learning." *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '11*, 42. New York, New York, USA.

Crammer, Koby, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. 2006. "Online Passive-Aggressive Algorithms." *J. Mach. Learn. Res.*, 7:551–585. Software available at https://code.google.com/p/sofia-ml/.

Denkowski, Michael, Chris Dyer, and Alon Lavie. 2014. "Learning from post-editing: Online model adaptation for statistical machine translation." *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, 395–404. Gothenburg, Sweden.

Federico, Marcello, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2014a. "Assessing the Impact of Translation Errors on Machine Translation Quality with Mixed-effects Models." *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar. **To appear.**

Federico, Marcello, Nicola Bertoldi, Mauro Cettolo, Matteo Negri, Marco Turchi, Marco Trombetti, Alessandro Cattelan, Antonio Farina, Domenico Lupinetti, Andrea Martines, Alberto Massidda, Holger Schwenk, Loïc Barrault, Frederic Blain, Philipp Koehn, Christian Buck, and Ulrich Germann. 2014b. "THE MATECAT TOOL." *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: System Demonstrations*, 129–132. Dublin, Ireland.

Federico, Marcello, Alessandro Cattelan, and Marco Trombetti. 2012. "Measuring User Productivity in Machine Translation Enhanced Computer Assisted Translation." *Proceedings of the Tenth Conference of the Association for Machine Translation in the Americas (AMTA)*.

Friedman, Milton. 1937. "The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance." *Journal of the American Statistical Association*, 32(200):675–701.

Friedman, Milton. 1940. "A Comparison of Alternative Tests of Significance for the Problem of m Rankings." *The Annals of Mathematical Statistics*, 11(1):86–92.

Geurts, Pierre, Damien Ernst, and Louis Wehenkel. 2006. "Extremely randomized trees." *Machine Learning*, 63(1):3–42.

Graves, Alex. 2008. "Rnnlib: A recurrent neural network library for sequence learning problems." http://sourceforge.net/projects/rnnl/.

Holm, Sture. 1979. "A Simple Sequentially Rejective Multiple Test Procedure." *Scandinavian Journal of Statistics*, 6(2):pp. 65–70.

Jalali, Ali, PD Ravikumar, S Sanghavi, and C Ruan. 2010. "A Dirty Model for Multi-task Learning." *Advances in Neural Information Processing Systems (NIPS) 23*.

Kirchhoff, K, D Capurro, and A Turner. 2013. "A conjoint analysis framework for evaluating user preferences in machine translation." *Machine Translation*, 1–17.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. "Moses: open source toolkit for statistical machine translation." *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, 177–180. Stroudsburg, PA, USA.

Koponen, Maarit. 2012. "Comparing Human Perceptions of Post-editing Effort with Post-editing Operations." *Proceedings of the Seventh Workshop on Statistical Machine Translation*, 181–190.

Lin, Chin-Yew and Franz Josef Och. 2004. "Orange: a method for evaluating automatic evaluation metrics for machine translation." *Proceedings of Coling 2004*, 501–507. Geneva, Switzerland.

Luong, Ngoc-Quang, Laurent Besacier, and Benjamin Lecouteux. 2014. "Word confidence estimation and its integration in sentence quality estimation for machine translation." *Knowledge and Systems Engineering*, vol. 244, 85–98. Springer.

Mehdad, Yashar, Matteo Negri, and Marcello Federico. 2012. "Match without a Referee: Evaluating MT Adequacy without Reference Translations." *Proceedings of the Machine Translation Workshop (WMT2012)*.

Mihalcea, Rada. 2007. "Using Wikipedia for Automatic Word Sense Disambiguation." *Proceedings of NAACL-HLT*, 196–203.

Pan, Sinno Jialin and Qiang Yang. 2010. "A Survey on Transfer Learning." *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. "Bleu: a method for automatic evaluation of machine translation." *Annual Meeting of the Association for Computational Linguistics*, 311–318. Philadelphia, PA, USA.

Parrella, Francesco. 2007. "Online support vector regression." *Master's Thesis, Department of Information Science, University of Genoa, Italy*. Sofware available at: http://www2.imperial.ac.uk/~gmontana/onlinesvr.htm.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. "Scikit-learn: Machine Learning in Python." *Journal of Machine Learning Research*, 12:2825–2830.

Plitt, Mirko and François Masselot. 2010. "A productivity test of statistical machine translation post-editing in a typical localisation context." *Prague Bulletin of Mathematical Linguistics*, 93:7–16.

Popović, Maja and Hermann Ney. 2011. "Towards automatic error analysis of machine translation output." *Comput. Linguist.*, 37(4):657–688.

Potet, Marion, Emmanuelle Esperança-Rodier, Laurent Besacier, and Hervé Blanchon. 2012. "Collection of a Large Database of French-English SMT Output Corrections." *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. Istanbul, Turkey.

Rubino, Raphael, José G. C. de Souza, Jennifer Foster, and Lucia Specia. 2013. "Topic Models for Translation Quality Estimation for Gisting Purposes." *Machine Translation Summit (MT Summit) XIV*, 295–302.

Sanchis, Alberto, Alfons Juan, and Enrique Vidal. 2007. "Estimation of confidence measures for machine translation." *Proceedings of the Machine Translation Summit XI*, 407–412.

Schmid, Helmut. 1995. "Improvements in Part-of-Speech Tagging with an Application to German." *Proceedings of the ACL SIGDAT-Workshop*, 47–50. Dublin, Ireland.

Shah, Kashif, Marco Turchi, and Lucia Specia. 2014. "An efficient and user-friendly tool for machine translation quality estimation." *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. Reykjavik, Iceland.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. "A Study of Translation Edit Rate with Targeted Human Annotation." *Proceedings of the Association for Machine Translation in the Americas*, 223–231.

Specia, Lucia, Nicola Cancedda, Marc Dymetman, Marco Turchi, and Nello Cristianini. 2009. "Estimating the sentence-level quality of machine translation systems." *Proceedings of the 13th Annual Conference of the European Association for Machine Translation (EAMT'09)*, 28–35. Barcelona, Spain.

Specia, Lucia, Dhwaj Raj, and Marco Turchi. 2010. "Machine Translation Evaluation versus Quality Estimation." *Machine translation*, 24(1):39–50.

Steinberger, Ralf, Bruno Pouliquen, Anna Widiger, Camelia Ignat, Tomaz Erjavec, Dan Tufis, and Dániel Varga. 2006. "The JRC-Acquis: a Multilingual Aligned Parallel Corpus with 20+ Languages." *CoRR*, abs/cs/0609058.

Tiedemann, Jörg. 2009. "News from OPUS - A collection of multilingual parallel corpora with tools and interfaces." N. Nicolov, K. Bontcheva, G. Angelova, and R. Mitkov (eds.), *Recent Advances in Natural Language Processing*, vol. V, 237–248. Borovets, Bulgaria: John Benjamins, Amsterdam/Philadelphia.

Turchi, Marco, Antonios Anastasopoulos, José G. C. de Souza, and Matteo Negri. 2014a. "Adaptive Quality Estimation for Machine Translation." *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 710–720. Baltimore, Maryland.

Turchi, Marco and Matteo Negri. 2014. "Automatic Annotation of Machine Translation Datasets with Binary Quality Judgements." *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, 1788–1792. Reykjavik, Iceland.

Turchi, Marco, Matteo Negri, and Marcello Federico. 2013. "Coping with the Subjectivity of Human Judgements in MT Quality Estimation." *Proceedings of the 8th Workshop on Statistical Machine Translation (WMT'13)*. Sofia, Bulgaria.

Turchi, Marco, Matteo Negri, and Marcello Federico. 2014b. "Data-driven Annotation of Binary MT Quality Estimation Corpora Based on Human Post-edition." *Machine translation*, Special Issue on Post-editing. **To appear.**

Turian, Joseph P., Luke Shen, and I. Dan Melamed. 2003. "Evaluation of machine translation and its evaluation." *In Proceedings of MT Summit IX*, 386–393. New Orleans, LA, USA.

Ueffing, Nicola and Hermann Ney. 2007. "Word-level confidence estimation for machine translation." *Computational Linguistics*, 33:9–40.

Wisniewski, Guillaume, Anil Kumar Singh, Natalia Segal, and François Yvon. 2013. "Design and Analysis of a Large Corpus of Post-Edited Translations: Quality Estimation, Failure Analysis and the Variability of Post-Edition." *Machine Translation Summit XIV*, 117–124.