Crawl and crowd to bring machine translation to under-resourced languages

Toral Ruiz, Antonio

Published in:
Language Resources and Evaluation

DOI:
10.1007/s10579-016-9363-6

Publication date:
2017

Lang Resources & Evaluation (2017) 51:1019–1051
DOI 10.1007/s10579-016-9363-6

ORIGINAL PAPER

Crawl and crowd to bring machine translation to under-resourced languages

Antonio Toral[1] · Miquel Esplá-Gomis[2] · Filip Klubička[3] · Nikola Ljubešić[3] · Vassilis Papavassiliou[4] · Prokopis Prokopidis[4] · Raphael Rubino[5] · Andy Way[1]

Published online: 25 June 2016

Abstract We present a widely applicable methodology to bring machine translation (MT) to under-resourced languages in a cost-effective and rapid manner. Our proposal relies on web crawling to automatically acquire parallel data to train statistical MT systems if any such data can be found for the language pair and domain of interest. If that is not the case, we resort to (1) crowdsourcing to translate small amounts of text (hundreds of sentences), which are then used to tune statistical MT models, and (2) web crawling of vast amounts of monolingual data (millions of

✉ Antonio Toral
atoral@computing.dcu.ie

Miquel Esplá-Gomis
mespla@dlsi.ua.es

Filip Klubička
fklubick@ffzg.hr

Nikola Ljubešić
nljubesi@ffzg.hr

Vassilis Papavassiliou
vpapa@ilsp.gr

Prokopis Prokopidis
prokopis@ilsp.gr

Raphael Rubino
rrubino@prompsit.com

Andy Way
away@computing.dcu.ie

[1] ADAPT Centre, School of Computing, Dublin City University, Dublin, Ireland

[2] Dep. Llenguatges i Sistemes Informàtics, Universitat d'Alacant, Alacant, Spain

[3] Faculty of Humanities and Social Sciences, University of Zagreb, Zagreb, Croatia

[4] Institute for Language and Speech Processing, Athens, Greece

sentences), which are then used to build language models for MT. We apply these to two respective use-cases for Croatian, an under-resourced language that has gained relevance since it recently attained official status in the European Union. The first use-case regards tourism, given the importance of this sector to Croatia's economy, while the second has to do with tweets, due to the growing importance of social media. For tourism, we crawl parallel data from 20 web domains using two state-of-the-art crawlers and explore how to combine the crawled data with bigger amounts of general-domain data. Our domain-adapted system is evaluated on a set of three additional tourism web domains and it outperforms the baseline in terms of automatic metrics and/or vocabulary coverage. In the social media use-case, we deal with tweets from the 2014 edition of the soccer World Cup. We build domain-adapted systems by (1) translating small amounts of tweets to be used for tuning by means of crowdsourcing and (2) crawling vast amounts of monolingual tweets. These systems outperform the baseline (Microsoft Bing) by 7.94 BLEU points (5.11 TER) for Croatian-to-English and by 2.17 points (1.94 TER) for English-to-Croatian on a test set translated by means of crowdsourcing. A complementary manual analysis sheds further light on these results.

Keywords Statistical machine translation · Web crawling · Crowdsourcing

1 Introduction

Machine translation (MT) can be considered a mature technology nowadays, with many recent success stories involving different languages and domains, both in research, e.g. Graham et al. (2014), and in industry, e.g. Zhechev (2012). All these cases have something in common: for all the language pairs and domains involved there are vast amounts of parallel data available. However, this is not the case for most language pairs and domains. For example, if we look at the level of MT support for European languages, out of 30 languages, only 3 languages are considered to count with moderate to good support (English, Spanish and French), while the level of support for the remaining 27 languages ranges from fragmentary to weak/none (Rehm and Uszkoreit 2013).

Due to the fact that the vast majority of language pairs and domains are under-resourced regarding MT, we believe it is essential to come up with approaches to tackle this resource bottleneck so that these languages and domains can be equipped with MT support in a way that is both cost-effective and rapid. In this paper we propose the use of two techniques to this end: web crawling and crowdsourcing. Namely, we propose to use web crawling to automatically acquire parallel data to train statistical MT (SMT) systems, if any such data can be found for the language pair and domain of interest. If that is not the case, we resort to (1) crowdsourcing to translate small amounts of text (hundreds of sentences), which can then be used to

[5] Prompsit Language Engineering, S.L., Elx, Spain

tune SMT models, and to (2) monolingual crawling of vast amounts of monolingual data (millions of sentences), which are then used to build language models for SMT.

To assess the feasibility of our proposed approach, we present two use-cases involving Croatian, an under-resourced[1] language that has gained relevance in the European context, since it recently attained official status in the European Union, on July 1st 2013. The first use-case regards the tourism domain, given the strategic importance of this sector to Croatia's economy. The second use-case has to do with tweets, due to the growing importance of social media.

For tourism, we will take advantage of the fact that one can find parallel data for this domain on the Internet. We will crawl parallel data from a set of web domains containing content about tourism. We will then use the acquired information to train MT systems adapted to tourism. Finally, we will assess how these systems perform on a novel set of tourism web domains.

For the social media use-case, we will build MT systems to translate tweets from the 2014 edition of the soccer World Cup, the biggest event yet on Twitter.[2] This use-case is specially challenging because there are no sources of parallel data available. We will build domain-adapted systems by (1) translating small amounts of tweets to be used for tuning by means of crowdsourcing and (2) crawling vast amounts of monolingual tweets.

This paper is part of the Abu-MaTran project,[3] whose main line of research has to do with rapid development of MT with a focus on industrial uptake, and has taken the family of South Slavic languages as its case study. This paper, aiming at providing a broad and detailed picture on rapid and cost-effective approaches to bring MT to under-resourced languages, builds upon previous work done on this topic within the project, mainly regarding:

- Acquisition of monolingual corpora. We have previously acquired vast quantities of monolingual data from the web, both for standard language (Ljubešić and Klubička 2014) and for user-generated content (Ljubešić et al. 2014).
- Acquisition of parallel corpora. We have already collected parallel corpora from the web both generic[4] and domain-specific for tourism (Esplà-Gomis et al. 2014).
- MT for a specific domain: tourism (Toral et al. 2014). In this paper we extend on this use-case by (1) covering both translation directions (this previous work looked only at Croatian-to-English) and (2) providing a real-world evaluation, as we test our systems separately on different web domains.

[1] According to a study (Rehm and Uszkoreit 2013) that analyses the state of language technology support for 30 European languages in four areas (machine translation, speech, text analytics and language resources), Croatian is given the lowest mark out of five (weak/none) for three of the areas and the second lowest mark (fragmentary) for the remaining area (language resources).

[2] http://www.theguardian.com/technology/2014/jul/15/twitter-world-cup-tweets-germany-brazil.

[3] http://abumatran.eu.

[4] http://nlp.ffzg.hr/resources/corpora/hrenwac/.

The rest of the paper is organised as follows. Section 2 gives an overview of the state-of-the-art of the two techniques that we use in our work: web crawling and crowdsourcing. This is followed by our use-cases on tourism and World Cup tweets in Sects. 3 and 4, respectively. Finally, we draw up conclusions and outline avenues of future work in Sect. 5.

2 Background

2.1 Web crawling for language resources

The Internet has become the largest source of information, specially regarding written text. The vast amount of texts publicly available in many languages has lead to a view of the web as a huge corpus (Kilgarriff and Grefenstette 2003) that can be used by linguists studying language use and change, and at the same time be exploited in applied research fields like MT, cross-lingual information retrieval, multilingual information extraction, etc. For this reason a considerable research effort has been devoted to exploiting this information for natural language processing during the last decades.

Obtaining monolingual data from the Web is relatively easy. The process usually starts by sending queries to a search engine in order to obtain seed URLs (Baroni et al. 2009) or by traversing a top level domain (e.g. *.es) and performing language identification to keep the web pages in the language of interest, e.g. Catalan (Boleda et al. 2006). Beside using search engines and/or crawling for collecting data, some services like Twitter nowadays have APIs for accessing the published data, or they simply make their database dumps available for download, which is the case of Wikipedia. The CommonCrawl project[5] should be mentioned here as it allows researchers to traverse a frequently updated crawl of the whole web in search of specific data, and therefore bypass the data collection process.

Once that monolingual corpora have been obtained for a set of languages, a common approach is to classify them in specific domains (i.e. construct domain-specific subsets of the initial collections) and then extract parallel sentences from the identified comparable corpora. Talvensaari et al. (2008) used a focused crawling system to produce comparable corpora in the genomics domain in English, Spanish and German languages. Munteanu and Marcu (2006) attempted to extract parallel sub-sentential fragments from comparable bilingual corpora using a signal-processing approach for producing training data sets for MT systems. A report on different methodologies used to collect small-scale corpora in nine language pairs and various comparability levels was provided by Skadina et al. (2010) and collected corpora were investigated for defining criteria and metrics of comparability.

Obtaining parallel corpora from multilingual websites is a complex problem, involving several sub-tasks, such as detecting parallel documents in a web site, language identification, sentence alignment, etc. From these tasks, detecting parallel

[5] http://commoncrawl.org.

documents is specially challenging. On the one hand, it can be difficult to find parallel documents in a collection of multilingual web pages which usually belong to the same domain. On the other hand, there are some specific features of websites (URLs, HTML structure, HTML metadata, etc.) which can be exploited. The main strategies available in the bibliography for detecting parallel documents in multilingual websites are:

- similarities in the URLs corresponding to web pages from a web site (Nie et al. 1999);
- parallelisms in the structure of HTML files (Resnik and Smith 2003); and
- content-similarity techniques (mostly based on bag-of-words overlapping metrics) (Ma and Liberman 1999).

Additional heuristics can be found in the bibliography, such as file size comparison, language markers in the HTML structure, mutual hyper-links between web pages, or images co-occurrence (Papavassiliou et al. 2013). It is usual to combine several of these methods in order to improve the performance.

All these strategies are aimed at being generic and language independent, that is, methods that can be applied, in theory, to any website and any pair of languages. However, ad-hoc approaches are also proposed for web domains where one could gather valuable data which would not be possible to be acquired by relying on generic approaches. These approaches allow the developer to tailor the corpora acquisition process to the specific characteristics of the web domains that are to be targeted. Two examples of ad-hoc approaches follow. The *SETimes* corpus (Tyers and Alperen 2010), which contains parallel corpora for nine languages gathered from newstories found at http://setimes.com. The *OpenSubtitles* corpus (Tiedemann 2009) contains parallel corpora consisting of subtitles. In this case the acquisition process is tailored to the nature of the text, e.g. by using the timing information as part of the alignment algorithm.

2.2 Crowdsourcing in natural language processing

Crowdsourcing has become a popular technique in the field of natural language processing in the last years as it allows to complete several tasks in a cheap and fast manner, while the quality of the results is reasonable. This technique is commonly used for tasks such as annotation of training data and evaluation of systems' output. One of the pioneering works in this topic used crowdsourcing to annotate data for five tasks (affect recognition, word similarity, recognizing textual entailment, event temporal ordering, and word sense disambiguation) (Snow et al. 2008). A workshop devoted to the use of crowdsourcing to the creation of language and speech data was held in 2010.[6]

In the area of MT, crowdsourcing has been used mainly to evaluate MT output (Callison-Burch 2009) and to produce translations. We give a more detailed account on previous work on producing translations since that is the task for which we use

[6] https://sites.google.com/site/amtworkshop2010/home.

crowdsourcing in this work. Irvine and Klementiev (2010) created translation lexicons between English and 42 rare languages. Zaidan and Callison-Burch (2011) solicited redundant translations from crowdsourcing workers and used supervised machine learning to select the best translation. Resnik et al. (2013) created translations relying only on monolingual speakers. Ambati and Vogel (2010) produced translations for sentences and phrases, both with and without context. Zbib et al. (2013) compared MT systems trained and tuned on the same parallel data with translations obtained using crowdsourcing versus professional translations. Munro (2010) presented an applied use of crowdsourcing to translate text messages as part of an emergency response system. Wasala et al. (2013) built a web-based platform for the collaborative localisation of user-generated content based on crowdsourcing techniques.

Finally, regarding South Slavic languages, crowdsourcing has been used to correct errors in the Slovene Wordnet (Fišer et al. 2014) and in building a morphosyntactically annotated corpus for Croatian (Klubička and Ljubešić 2014).

3 Tourism use-case

This section presents our use-case on the tourism domain and is structured as follows. First, Sect. 3.1 gives a detailed account of the crawling procedures followed to obtain the different types of corpora used in this use-case. This is followed in Sect. 3.2 by the evaluation, which covers the data sets used (Sect. 3.2.1), the MT systems built (Sect. 3.2.2) and finally the results (Sect. 3.2.3).

3.1 Crawling

The performance of SMT systems is highly dependent on the quantity and quality of available training data and, crucially, on the relevance of the training data to the translation task. Since there seems to be a consensus that there is never enough in-domain training data that is directly relevant to the translation task at hand (Axelrod et al. 2011), a common approach is to develop a general-scope SMT system using a large parallel corpus and adapt it to a specific purpose using relatively smaller in-domain parallel resources acquired from the web (Laranjeira et al. 2014). In the following subsections we provide more details on our approach for the acquisition of monolingual (Sect. 3.1.1), as well as general (Sect. 3.1.2) and domain-specific (Sect. 3.1.3) parallel resources for the tourism use-case.

3.1.1 Monolingual

The Web is a very cheap source of monolingual linguistic material, but choosing the right content for one's needs is still an open challenge. Many languages have dedicated top-level domains (TLDs) and building large corpora for those languages often consists of collecting linguistically relevant material from the respective TLD.

The methodology of building web corpora in general has matured in the last decade and the process of building a web corpus generally consists of the following steps:

1. crawling—recursively visiting and storing the content of web documents, starting from a seed list of URLs
2. deduplication—discarding web documents that are identical to already retrieved documents
3. content extraction consisting of
    - encoding guessing—although there is text encoding information encoded both in the HTTP header and the document header (in case of HTML documents), it is sometimes false and so regularly one additional encoding check-up is performed
    - language identification—classifying each document by the language(s) used in it
    - boilerplate removal—removing headers, menus, footers and all remaining document boilerplate, leaving just the essential text of the document
4. near-deduplication—many documents on the web are just minor edits of other documents, so typically all documents (or paragraphs) having a 5-gram intersection of more than 50 % to an already retrieved document (or paragraph) are discarded
5. linguistic processing—segmenting and annotating retrieved text on various linguistic levels

Over the last years out-of-the-box solutions for building large web corpora started to emerge, the Spiderling crawler[7] (Suchomel and Pomikálek 2012) being one of the examples which additionally improved on the set-out methodology by merging crawling with content extraction and deduplication with the goal of steering the crawl towards web domains that yield more relevant and unique language material.

The well-developed methodology and available tools nowadays make building billion-token corpora of languages with a few million speakers from highly developed countries a rather straightforward task.

3.1.2 General-domain parallel corpora

Besides being useful for collecting monolingual data, the web also hides multilingual gems in form of parallel data.

The simplest approach to collecting parallel data from the web is to manually identify large multilingual websites, crawl them and write content extractors that provide additional metadata and alignments between documents. A good example of such an approach to collecting parallel data, covering Croatian and English, is the SETimes corpus[8] which consists of news articles from the http://setimes.com

[7] http://nlp.fi.muni.cz/trac/spiderling.

[8] http://nlp.ffzg.hr/resources/corpora/setimes/.

domain published in nine southeastern European languages and English. While the amount of parallel content retrieved with such methods is significant, the diversity of the content is often quite limited.

Another quite simple, but frequently noisy source of parallel data are subtitles. While the OpenSubtitles corpus, which is part of the OPUS collection,[9] is infamous for its noise, clean gems of that sort of content can be found as well, one example containing Croatian and English language being the TED talks subtitles corpus.[10]

In addition to these well known sources of parallel material, there are many others, scattered across the web, smaller and less known, but with more diverse content. The task of identifying those unknown "hotspots" of parallel material actually requires, if no high-coverage web search API is available, for most of the web (or specific TLD) to be crawled. Therefore this task aligns well with the task of collecting large collections of monolingual data. An example of a Croatian–English parallel corpus produced as a byproduct of building the Croatian web corpus (Ljubešić and Erjavec 2011; Ljubešić and Klubička 2014) is the hrenWaC corpus.[11] This corpus was built by (1) calculating content and structure similarity between each English document from a specific domain and each Croatian document from the same domain and (2) manually validating the pairs of documents with highest similarity. Future versions of this corpus will be built by automating the second part of the corpus construction process with methods described in the remainder of this section.

3.1.3 Domain-specific parallel corpora

One of the aspects covered in this work is the creation of domain-adapted MT systems. To build such systems, we propose an approach based on the collection of parallel data from the Internet. To this end, we combined two automatic parallel-data crawlers, ILSP-FC (Papavassiliou et al. 2013) and bitextor, in order to crawl a set of 23 websites from the domain of tourism to build our English–Croatian tourism corpus (see Table 1). The table is divided in two parts: training and testing. This division corresponds to how the corpus resulting of crawling was used to train and evaluate different MT systems, as described in Sect. 3.2. A previous study (Esplà-Gomis et al. 2014) already showed the usefulness of combining these two crawlers in order to maximise the amount of data harvested. Both tools are able to produce a sentence-aligned parallel corpus given a list of URLs corresponding to multilingual websites, in this case, in English and Croatian. However, Esplà-Gomis et al. (2014) already reported the difficulty of straightforwardly combining the output of both crawlers: the differences in the processing carried out by each crawler on the text extracted from web pages results in a highly redundant corpus, hence impeding the measurement of the contribution of each crawler regarding the quality of the resulting MT system. To deal with this problem, a post-processing workflow has been designed to ensure the comparability of the data obtained by both crawlers.

[9] http://opus.lingfil.uu.se.

[10] http://nlp.ffzg.hr/resources/corpora/ted-talks/.

[11] http://nlp.ffzg.hr/resources/corpora/hrenwac/.

This section first describes the tools used for crawling, and then the processing performed to obtain the final corpus.

ILSP Focused Crawler. The ILSP Focused Crawler (ILSP-FC)[12] is a modular system that includes components and methods for all the tasks required to acquire domain-specific corpora from the Web. The system is available as an open-source Java project and due to its modular architecture, each of its components can be easily substituted by alternatives with the same functionalities. Depending on user-defined configuration, the crawler employs processing workflows for the creation of either monolingual corpora or bilingual collections (i.e. pairs of parallel documents acquired from multilingual web sites). The main modules integrated in ILSP-FC are:

1. page fetcher: adopts a multithreaded crawling implementation in order to ensure concurrent visiting of multiple web pages/hosts.
2. normaliser: parses the structure of each fetched web page and extracts its metadata; detects page encoding and encodes text to UTF-8 if required.
3. cleaner: extracts structural information (i.e. title, heading, etc.) and identifies boileplate paragraphs.
4. language identifier: uses the *Cybozu*[13] library to detect the main language of a document, as well as paragraphs in a language different from the main one.
5. link extractor: examines the anchor text of the extracted links and ranks them by the probability that a link from a page points to a candidate translation of this page, with the purpose of forcing the crawler to visit candidate translations first.
6. de-duplicator: checks each document against all others and identifies (near) duplicates based on lists of quantised word frequencies extracted from each document and on the number of common paragraphs.
7. pair detector: examines each document against all others and identifies pairs of documents that could be considered parallel. Its main methods are based on URL similarity, co-occurrences of images with the same filename in two documents, and the documents' structural similarity.

Bitextor. Bitextor[14] is a free/open-source tool for harvesting bitexts from multilingual websites. The version 4.0 of the tool is a re-implementation of that described by Esplà-Gomis and Forcada (2010). This is an improved version which outperforms the previous ones in text pre-processing and post-processing. The new architecture of the tool is highly modular, consisting of a collection of scripts (mainly in python and bash) which are organised in a Unix pipeline. This design is aimed at optimising the parallelisation of resources for crawling large websites. One of the main differences between this version of bitextor and the previous one is that the techniques based on URL similarity have been replaced by new methods based on bag-of-words overlapping. URL similarity can be a useful source of information in some cases, specially when URLs are the same for the translated versions of the

[12] http://nlp.ilsp.gr/redmine/projects/ilsp-fc.

[13] http://code.google.com/p/language-detection/.

[14] http://sourceforge.net/projects/bitextor/.

Table 1 List of websites crawled to build the tourism-domain English–Croatian parallel corpus, including their URL and a short description of the contents

	URL	Description
Train	http://www.adria-bol.hr	Tourist agency based in the city of Bol
	http://www.animafest.hr	Portal of the World Festival of Animated Film in Zagreb
	http://bol.hr	Tourism portal of the city of Bol
	http://www.burin-korcula.hr	Tourist agency based in Korčula island
	http://www.camping.hr	Website of the Croatian Camping Union
	http://www.dalmatia.hr	Official tourism portal of Dalmatia Country
	http://dubrovnik-festival.hr	Portal of the Dubrovnik Summer Festival
	http://www.events.hr	Croatian online travel agency
	http://www.galileo.hr	Croatian online travel agency
	http://hhi.hr	Hydrographic Institute of Croatia
	http://www.kvarner.hr	Official tourism portal of Kvarner County
	http://plavalaguna.hr	Hotel chain *Laguna Poreč*
	http://www.liburnia.hr	Hotel chain *Liburnia Riviera*
	http://m.pulainfo.hr	Tourism portal of the city of Pula
	http://www.portauthority.hr	Croatian Association of Port Autorities
	http://www.putomania.com.hr	Portal about travelling around the world
	http://www.tzg-rab.hr	Tourism portal about Rab island
	http://tzgrovinj.hr	Official tourism portal of Rovinj-Rovigno
	http://www.uniline.hr	Croatian online tourist agency
	http://urbanfestival.blok.hr	Festival of urban culture
Test	http://www.istra.hr	Official tourism portal of Istria
	http://www.val-losinj.hr	Tourist agency based in Losinj island
	http://tz-malilosinj.hr	Tourism portal of the town of Mali Lošinj

The corpus is divided in two parts (training and testing) corresponding to how the corpus is used for the experiments described in Sect. 3.2

same web page in a web site with the only variation of a directory name, or a variable value indicating the language. However, this is not a generalised practice and is, therefore, not guaranteed to work for all websites. However, bag-of-words overlapping metrics have proved to be a useful method (Achananuparp et al. 2008) which can be applied to any document (even to raw text) and which complements the structural information of the HTML documents with linguistic information.

Given a multilingual website and the pair of targeted languages (L_1, L_2) for which a parallel corpus is to be created, bitextor performs the following steps:

1. the website is downloaded by means of the tool *HTTrack*,[15] keeping only HTML documents (*D*);

[15] http://www.httrack.com/.

2. the collection of downloaded documents is preprocessed by: (1) fixing errors in the HTML structure of the documents with *Apache Tika*,[16] (2) removing boilerplates with boilerpipe[17] (Kohlschütter et al. 2010), and (3) discarding duplicates;
3. the language of each file is detected with LangID,[18] and only those identified as being writen in L_1 or L_2 are kept;
4. for each document D_i in L_1 and each document D_j in L_2, a word-overlapping-based score $S(D_i, D_j)$ is obtained with the help of a bilingual lexicon provided by the user;[19]
5. for every document, an n-best candidates list is obtained by choosing the n documents with the highest score;
6. the n-best candidates list of each document is re-ranked by applying a new score $S'(D_i, D_j)$ combining word-overlapping-based scores with a score based on the Levenshtein edit distance between the HTML structure of each pair of documents;
7. the n-best candidates list of each document D_i is symmetrised: first, only candidates D_j which have D_i in their corresponding n-best list are kept; then, the candidates list of each D_i is re-ranked by computing the average score between $S'(D_i, D_j)$ and $S'(D_j, D_i)$;
8. hunalign[20] (Varga et al. 2005) is used to obtain an indicative score regarding the quality of the sentence-alignment between both documents.

Resulting corpora. Both ILSP-FC and bitextor were used to build a tourism domain-specific corpora. Two different settings were used for each tool:

- *fc-all*: Includes all the pairs detected by the tool (i.e. default configuration);
- *fc-reliable*: Includes a subset of the *all* configuration where only those pairs identified through image co-occurrences and high structural similarity (Papavassiliou et al. 2013) are kept;
- *Bitextor-10best*: 10-best candidate lists are used to get the pairs of documents, which allows to align a document with several alternative documents in the other language; and
- *Bitextor-1best*: 1-best candidate lists are used to get the pairs of documents; this setting is more strict than *10-best*, since it only aligns documents which are mutual best candidates.

The settings *bitextor-1best* and *fc-reliable* are accuracy-oriented. Both settings focus on the quality of the parallel data harvested at the cost of loosing some potentially useful parallel data. Conversely, *bitextor-10best* and *fc-all* are recall-oriented

[16] http://tika.apache.org/.

[17] http://code.google.com/p/boilerpipe/.

[18] https://github.com/saffsd/langid.py.

[19] $S(D_j, D_i)$ is also obtained, since score $S(\cdot)$ is not symmetric.

[20] http://mokk.bme.hu/resources/hunalign/.

Table 2 Information about the parallel data obtained by crawling

	Segment	Total tokens		Unique tokens	
	Pairs	en	hr	en	hr
Bitextor-10best	64,489	460,466	367,949	202,247	168,906
Bitextor-1best	48,234	338,875	268,993	168,798	150,414
fc-all	89,801	543,837	427,946	277,801	266,367
fc-reliable	74,728	459,272	359,225	245,905	233,905

The table includes the number of segment pairs, the total number of tokens and the number of unique tokens on the lowercased corpus, both for English and Croatian

settings, which presumably lead to noisier parallel data. Table 2 includes a more detailed analysis about the amount of data in each data set.

In order to homogenise the output of each crawler, the content of the detected document pairs is reprocessed by applying the steps of the following workflow on paragraphs[21] detected by each crawler.

1. *segmentation*: paragraphs are segmented by using the splitter implemented in the NLTK package[22] for Python.
2. *sentence alignment*: hunalign is then used to align the sentences from each document pair.[23]
3. *cleaning*: some filters are applied to remove useless translation units (TUs), such as those with identical content in both translation unit variants (TUVs), or those containing only numbers or punctuation in any of the TUVs.
4. *merging*: finally the corpora obtained by each tool are merged in a translation memory (TM), keeping, for each TU, the list of crawlers/settings that detected it, the source files from which it was extracted, and the score assigned by hunalign to the aligned sentence pair.

At the end of this process, a TM was obtained, consisting of 142,147 TUs. The TM was formatted following the TMX specification.[24] The special tag `<prop>` was used to store the complementary information mentioned in the *merging* step of the normalisation workflow.

Table 3 shows the Jaccard index (Chakrabarti 2003, Chapter 3) between the collections of translation units in the final TM obtained with each setting. Additionally, the last column of this table reports the Jaccard index between the collection of TUs obtained with each setting and the total number of TUs in the TM, that is, the proportion of the TM obtained with each setting. Table 4 shows the same

[21] For our task, paragraphs are blocks of text which may contain more than one sentence.

[22] http://www.nltk.org/.

[23] The English–Croatian bilingual lexicon available at http://sourceforge.net/projects/bitextor/files/bitextor/bitextor-4.0/dictionaries/ was used for sentence alignment with hunalign. In addition, this tool was run with the option bisent to ensure one-to-one sentence alignments.

[24] http://www.gala-global.org/oscarStandards/tmx/tmx14b.html.

Table 3 Jaccard index measuring the relation between the different corpora obtained by each setting in the final translation memory

	Bitextor-1best (%)	fc-all (%)	fc-reliable (%)	Merged (%)
Jaccard index between aligned corpora				
Bitextor-10best	72.55	10.90	11.76	46.08
Bitextor-1best	–	11.64	12.77	34.47
fc-all	–	–	83.22	64.17
fc-reliable	–	–	–	53.40

The last column measures the Jaccard index of each setting with the *merged* corpus obtained when producing the union of all the settings

Table 4 Number of different segment pairs in the intersection between the different corpora obtained by each setting in the final translation memory

	Bitextor-1best	fc-all	fc-reliable	Merged
Different segment pairs in the interesection of corpora				
Bitextor-10best	47,396	15,169	14,650	64,489
Bitextor-1best	–	14,392	13,926	48,234
fc-all	–	–	74,728	89,801
fc-reliable	–	–	–	74,728

The last column measures the amount of segment pairs in the *merged* corpus obtained by each of the settings

results in absolute terms, in order to have a more clear idea of the results shown in Table 3. These results show a small intersection between the two crawlers. As it can be seen, the *10best* configuration for bitextor obtained about 46 % of the TUs, while FC with setting *all* produced about 64 % of them, a reasonably balanced coverage for each setting. In general, only 15,190 TUs were obtained by both crawlers, which accounts for about 10.9 % of the total TM. This confirms that both crawlers are complementary and it is therefore useful to combine them to obtain larger parallel corpora.

3.2 Evaluation

This section contains the evaluation on MT for tourism. We will cover the data sets used, the MT systems built and finally the results obtained.

3.2.1 Data sets

The data sets used to train and evaluate the domain-especific MT systems were obtained from the parallel corpus described in Sect. 3.1.3. A collection of training sets were obtained for the different tools/settings used for crawling, which were later used to evaluate the contribution of each of them to the final MT system. In

addition to these training sets, a development set and a test set were built. To ensure the independence of the test set from the training and development sets, the original corpus was split into two parts: one containing sentence pairs from the first 20 websites in Table 1 (heneceforth Corpus_{20}), for training and development, and one containing 3 websites (Corpus_3) containing the last 3 websites in the same table, for testing. The different data sets built for our experiments were filtered to contain only sentence pairs useful for MT training and evaluation. Therefore, sentence pairs with at least one sentence shorter than 4 words or longer than 50 words were removed.

In the following we detail the procedures we follow to build the development, training and test sets.

Development set. The development and the training sets were derived from Corpus_{20}. Acknowleging that the quality of the development set has an important impact on MT performance, the development set was built taking the following elements into consideration:

- *crawlers overepresentation*: having different amounts of parallel sentences from bitextor and ILSP-FC in the development set would result in one of them being over-represented. To avoid this, only sentence pairs detected by both crawlers were included in the development set.
- *parallelness of the data*: in order to ensure that the data is reasonably parallel we kept sentence pairs (1) detected by both crawlers (these already likely to be parallel since they are retrieved by two systems that use different methods) and (2) with alignment score (as provided by hunalign) higher than 1.0.
- *data redundancy*: this set should be as heterogeneous as possible to enhance the representativeness of the data used for tuning MT systems. To that end, we remove duplicate[25] and near-duplicate sentences, both in English and Croatian. Regarding near-deduplication, we added a sentence pair to the set only if the fuzzy match scores[26] (FMS) between the sentence and those already in the development set are lower than 30 % for both languages. From the set of parallel sentences that fulfill these restrictions, we randomly chose 1000.
- *overfitting*: our preliminary experiments showed that building both the training sets and the development set from the same corpus can lead to overfitting the MT system, since the sentences in the development set can be too similar to those in the training sets. To avoid this, we also applied the FMS filtering described above to our 1000 sentence pairs to avoid those matching more than 30 % with any of those in the training sets. The resulting development set contained 532 sentence pairs.

Training sets. Four training sets were built from Corpus_{20}, one for each tool/setting. These training sets contain all those sentence pairs in Corpus_{20} which do not contain sentences already used for building the development set. The training sets contain:

[25] The comparison between the sentences was performed on lowercased text from which non-alphabetic characters (spaces, punctuation, and numbers) were removed.

[26] Fuzzy match scores measure the similarity between two strings by using the Levenshtein distance (Sikes 2007) to detect the elements (words in our case) matching between them.

- fc-all: 19,010 sentence pairs
- fc-reliable: 16,793 sentence pairs
- bitextor-10best: 27,274 sentence pairs
- bitextor-1best: 20,547 sentence pairs

In addition, we used a set of general-domain parallel corpora (hrenWaC,[27] SETimes[28] and TED Talks[29]), accounting all in all to 385,874 sentence pairs, to train the baseline systems.

Test sets. A different test set was built from each website in Corpus_3, consisting of 500 sentence pairs each. The objective was to obtain a high-quality collection of parallel sentences, representative of the data one could find in a real translation task. We followed the method used to obtain the development set, but changing some parameters. First, we removed exact duplicates, but we did not apply the FMS filtering, since it is realistic to find similar sentences in a translation for a website. We also used the hunalign score as a reference of parallelness of the data, although we relaxed this constraint and used a threshold of 0.5. Finally, we tried to use only sentence pairs detected by both crawlers, although it was not possible for all websites given the amount of data crawled. Only for *tz-malilosinj* we could obtain 500 sentence pairs detected by both crawlers. For the other two websites, we completed the collection of sentence pairs detected by both crawlers with the same amount of sentence pairs detected only by bitextor and only by ILPS-FC to reach 500 sentence pairs.

Monolingual data. SMT systems rely on a monolingual target language model (LM). Since we will build MT systems for both directions (Croatian to English and English to Croatian), we need monolingual corpora to train LMs for both target languages. For the Croatian-to-English direction, we used the data provided for the WMT14 translation task (Bojar et al. 2014),[30] as described in our system submission to that shared task (Rubino et al. 2014). For the English-to-Croatian direction, we used the target side of the general-domain parallel corpora (hrenWaC, SETimes and TED Talks) and hrWaC 2.0 (Ljubešić and Klubička 2014), a monolingual Croatian corpus crawled from the .hr top-level domain following the procedure described in Sect. 3.1.1.

3.2.2 SMT systems

Prior to training, tuning and evaluating the SMT systems, the corpora presented in Sect. 3.2.1 are pre-processed following these steps: punctuation normalising, tokenising, truecasing and escaping problematic characters. The scripts are available with the Moses toolkit (Koehn et al. 2007). The final evaluation of the SMT systems using the test set is done on the original tokenization and casing of the text, based on a de-tokenization and original-casing (*de-truecasing*) post-processing step.

[27] http://nlp.ffzg.hr/resources/corpora/hrenwac/.

[28] http://nlp.ffzg.hr/resources/corpora/setimes/.

[29] http://zeljko.agic.me/resources/.

[30] http://www.statmt.org/wmt14/translation-task.html.

The truecasing step involves the use of a truecase model, one for each language. We train them using all the available training and monolingual data in order to obtain reliable statistics on which words are usually written with uppercased variants. Truecasing is then applied to all the parallel and monolingual data, including the development and test sets.

Individual English LMs and the Croatian LM (concatenation of all Croatian data) are trained using KenLM (Heafield 2011) while the final English LM is built with the SRILM toolkit (Stolcke et al. 2011) thanks to its out-of-the-box LM interpolation mechanism. All LMs are unpruned modified Kneser-Ney smoothed 5-grams.

Phrase-based SMT systems are then built using Moses version 2.1.1.[31] and MGiza++ (Gao and Vogel 2008) for word alignment, both with their default parameters. Tuning of the SMT systems is carried out on the development set with the Margin Infused Relaxed Algorithm (MIRA) (Hasler et al. 2011).

In order to evaluate the performance of SMT systems built on crawled data, we define a baseline system trained on the general-domain parallel corpora described in Sect. 3.1.2. We tune this baseline system with two development sets: a non-tourism one, to be considered as a *pure* general-domain baseline system, and the crawled one described in Sect. 3.2.1. The non-tourism development set consists of 1000 sentences from the WMT13 test set[32] manually translated to Croatian.

Finally, after training individual SMT systems using the general and domain-specific parallel data, we interpolate the phrase and reordering tables of the best performing crawling setups on the development set (*reliable* and *1best*), prior to interpolating them with the baseline tables. These interpolations are performed by minimising the perplexity on the domain-specific development set (Sennrich 2012).

3.2.3 Results

The machine-translated text is evaluated in the target language against its translation reference based on two popular automatic metrics: BLEU (Papineni et al. 2002)[33] and TER (Snover et al. 2006).[34] BLEU is the *de facto* standard metric in the MT field. We use also TER as it is an error-rate metric whose score is based on the number of operations (insertions, deletions and edits) that are required to bring the MT output to match the reference, and thus provides an indication of the effort required to post-edit the MT output. To compare the results of our different SMT systems, statistical significance tests are carried out on BLEU with paired bootstrap resampling (Koehn 2004),[35] using 1,000 iterations.

According to these automatic metrics, as well as the out-of-vocabulary (OOV) rates between the test sets and the different training sets, we evaluate the SMT systems and present the results in Tables 5, 6 and 7 for the *istra*, *tz-malilosinj* and

[31] https://github.com/moses-smt/mosesdecoder/tree/RELEASE-2.1.1.

[32] http://www.statmt.org/wmt13/test.tgz.

[33] ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v13a.pl.

[34] http://www.umiacs.umd.edu/~snover/terp/.

[35] http://www.ark.cs.cmu.edu/MT/paired_bootstrap_v13a.tar.gz.

Table 5 Results obtained when translating the test set extracted from the *istra* website with the SMT systems built upon the non-tourism (baseline), the domain-specific crawled data individually, as well as the interpolation of the best crawled-based systems with and without the baseline

System		BLEU	TER	src OOV (%)
Croatian → English				
Baseline	Dev news	0.2884	0.6486	5.7
Baseline	Dev crawl	0.3032*	0.6418	
fc	All	0.2383	0.7092	12.2
fc	Reliable	0.2428	0.7032	12.6
Bitextor	1best	0.2118	0.7317	14.5
Bitextor	10best	0.2044	0.7423	13.9
Interpolation	Best crawl	0.2463	0.7022	10.8
Interpolation	Baseline + crawl	**0.3105***	**0.6382**	**4.9**
English →Croatian				
Baseline	Dev news	0.3010	0.6708	3.8
Baseline	Dev crawl	0.3111*	0.6625	
fc	All	0.2208	0.7633	6.8
fc	Reliable	0.2200	0.7592	6.9
Bitextor	1best	0.1931	0.7861	7.6
Bitextor	10best	0.1875	0.7898	7.2
Interpol	Best crawl	0.2339	0.7406	5.9
Interpol	Baseline + crawl	**0.3219***	**0.6484**	**3.4**

Results with * indicate significant improvement over the baseline with $p \geq 0.01$. The best result according to each metric (highest for BLEU and lowest for TER and OOV) is shown in bold

val-losinj test sets respectively. In these tables, the results obtained with the baseline system tuned using two development sets are presented along with the results obtained with the data crawled for each crawling configuration individually.

Table 5 shows the results obtained when decoding the *istra* website. For the Croatian to English direction, using the crawled development set leads to a 1.48pts absolute BLEU improvement (0.68 TER) over the baseline with the news development set. The crawlers-based systems reach between 0.2044 and 0.2428 BLEU ([0.7022, 0.7423] TER), while interpolating the data crawled by *fc-reliable* and *bitextor-1best* leads to a 0.35pt absolute BLEU improvement (0.1 TER). The interpolation of the baseline and the crawled-based systems leads to a 2.21pts absolute improvement (1.04 TER) over the non-adapted baseline. In the opposite direction, English-to-Croatian, we observe the same trends: using a crawled development set leads to significant improvement over the use of the news set and the interpolation of the baseline and the crawl-based system improves significantly over the baseline (2.09 BLEU and 2.24 TER).

Table 6 Results obtained when translating the test set extracted from the *tz-malilosinj* website with the SMT systems built upon the non-tourism (baseline), the domain-specific crawled data individually, as well as the interpolation of the best crawled-based systems with and without the baseline

System		BLEU	TER	src OOV (%)
Croatian →English				
Baseline	Dev news	**0.2571**	**0.6641**	5.0
Baseline	Dev crawl	0.2523	0.6724	
fc	All	0.1861	0.7481	14.1
fc	Reliable	0.1893	0.7415	14.6
Bitextor	1best	0.1696	0.7611	17.6
Bitextor	10best	0.1693	0.7626	16.9
Interpol	Best crawl	0.1976	0.7355	13.1
Interpol	Baseline + crawl	**0.2571**	0.6735	**4.5**
English →Croatian				
Baseline	Dev news	0.2603	0.6832	2.7
Baseline	Dev crawl	0.2624	0.6752	
fc	All	0.1634	0.7845	6.8
fc	Reliable	0.1671	0.7850	7.2
Bitextor	1best	0.1469	0.8058	8.6
Bitextor	10best	0.1467	0.8082	8.2
Interpol	Best crawl	0.1819	0.7672	6.3
Interpol	Baseline + crawl	**0.2623**	**0.6725**	**2.6**

The best result according to each metric (highest for BLEU and lowest for TER and OOV) is shown in bold

In Table 6, the results obtained when decoding the *tz-malilosinj* website are presented. For the Croatian to English direction, using a crawled development set does not improve over using a news set. A 0.48pt absolute BLEU degradation (0.83 TER) is observed when the developement set is switched from a news to a crawled one, but it is not significant. The BLEU scores obtained by the crawl-based systems are between 0.1693 and 0.1893 BLEU ([0.7415, 0.7626] TER), slightly lower (higher) than the ones obtained on the *istra* test set. Interpolating the crawl-based systems indicates the complementarity of the datasets, leading to a 0.83pt absolute BLEU improvement (0.6 TER) over the best individual crawler. Interpolating the baseline and the crawl-based systems does not improve (nor degrade) over the baseline in terms of BLEU but it does degrade slightly in terms of TER (0.94pt absolute). For the English to Croatian direction, a similar trend to what is observed with the *istra* test set is presented, except for the final interpolated model which does not improve significantly over the baseline with the *tz-malilosinj* test set.

Table 7 presents the results obtained on the *val-losinj* website as a test set. For both directions, using a web crawled development set leads to significant improvements over the use of a news development set. Compared to the other test sets used in our experiments, the BLEU scores obtained with the crawl-based are between 0.2579 and 0.2956, which are the highest BLEU obtained by a crawl-based system among the three test sets. Interpolating the crawl-based systems for the Croatian to English direction leads to a slight (non-significant) improvement over the baseline, which is notable compared to the other test sets and directions.

Table 7 Results obtained when translating the test set extracted from the *val-losinj* website with the SMT systems built upon the non-tourism (baseline), the domain-specific crawled data individually, as well as the interpolation of the best crawled-based systems with and without the baseline

System		BLEU	TER	src OOV (%)
Croatian →English				
Baseline	Dev news	0.2982	0.6948	4.2
Baseline	Dev crawl	0.3045*	0.6666	
fc	All	0.2956	0.6765	7.4
fc	Reliable	0.2841	0.6867	7.9
Bitextor	1best	0.2753	0.6954	8.8
Bitextor	10best	0.2780	0.6890	8.3
Interpol	Best crawl	0.3057	0.6648	6.4
Interpol	Baseline + crawl	**0.3496***	**0.6253**	**2.6**
English →Croatian				
Baseline	Dev news	0.2933	0.6686	3.1
Baseline	Dev crawl	0.3054*	0.6869	
fc	All	0.2804	0.7067	4.7
fc	Reliable	0.2616	0.7199	4.9
Bitextor	1best	0.2579	0.7266	4.9
Bitextor	10best	0.2590	0.7187	4.6
Interpol	Best crawl	0.2946	0.6918	3.9
Interpol	Baseline + crawl	**0.3504***	**0.6366**	**2.3**

Results with * indicate significant improvement over the baseline with $p \geq 0.01$. The best result according to each metric (highest for BLEU and lowest for TER and OOV) is shown in bold

Finally, interpolating the baseline and the crawl-based systems leads to significant improvements over the baselines for both directions.

The three test sets allow us to make general observations and conclusions on the obtained results, both on OOV rates and automatic metrics. In terms of OOV words, the baseline dataset has a better coverage of the English and Croatian vocabulary compared to crawled data, due to the size of the dataset. Except for the *val-losinj* test set, the *fc-all* crawler leads to the highest coverage compared to the other crawlers and configurations. The accuracy-oriented configuration of each crawler shows a complementarity of their data in terms of vocabulary coverage. Finally, the best coverage is obtained by combining the baseline and the crawled data.

In terms of automatic metrics, both of them show similar trends in most of the cases. When comparing OOV rates and automatic metrics, we see that lower source-side OOV rates are not necessarily correlated with improvements on automatic metrics.[36] This phenomenon is observed on the *fc-all – fc-reliable*, and *bitextor-10best – bitextor-1best* comparisons on the *val-losinj* test set. Amongst the crawlers and configurations tested, the accuracy-oriented configurations lead to equal or better scores than recall-oriented ones, except for the *val-losinj* test set. Finally, except for the *tz-malilosinj* test set, using a crawled development set with the

[36] While one might intuitively think that lower OOVs should correlate with better scores in terms of automatic MT evaluation metrics, this is not always the case as there are many other factors at play. MT evaluation metrics take into account word order, shifts, n-gram matching, etc. On top of these, a sizable portion of OOVs tend to be named entities, which in many cases are fine to be left untranslated, and if so whether the MT system covers them or not will not have any impact on the score produced by the MT metric.

baseline system improves over using a news development set, and interpolating the baseline and the crawl-based systems outperforms the baseline.

4 World cup tweets use-case

In contrast to our first use-case, where parallel data for the language pair and domain of interest was available, our second use-case regards the scenario where this is not the case. This case of study emerges from Brazilator,[37] a project on rapid adaptation of MT for social media carried out jointly by Dublin City University and Microsoft Research over 5 weeks during the Summer of 2014 (the duration of the World Cup and the previous week to that event), where the MT group at Dublin City University built domain-adapted systems for soccer tweets for thirteen languages and twenty-four language pairs.

We used the Microsoft Translator Hub[38] as the platform to run all the experiments for this task. The Hub is a web-based platform where users can easily build SMT systems tailored to their needs by uploading parallel and monolingual training data sets. Systems built on the Hub can be trained using either both users' data and Microsoft's or solely users' data. In a nutshell, we (1) used the general-domain systems for English-to-Croatian and Croatian-to-English provided by the Hub as our baselines and (2) built domain-adapted systems by uploading our in-domain datasets (parallel data for tuning and monolingual data for language modelling) to the Hub and subsequently training systems with these datasets, which are interpolated with the respective baseline systems provided by the Hub.

It is worth mentioning that The Hub provides normalisation of English tweets in a preprocessing step using an unsupervised approach based on random walks (Hassan and Menezes 2013). That said, Croatian tweets were not normalised prior to their translation.

As might be apparent by this point, our approach for adapting the MT systems to the domain and genre of World Cup tweets focuses solely on data. The reason for this is that we cannot adapt the MT systems from an algorithmic point of view as we do not have access to their inner functionalities in the Hub.

In the remainder of this section we cover in detail the domain adaptation process and the evaluation of the resulting MT systems.

4.1 Domain adaptation

Our data-driven approach for World Cup tweets concerns two types of datasets: training and tuning. The next two sections delve in the details for each of these.

[37] http://cngl.ie/brazilator/#/about.

[38] https://hub.microsofttranslator.com/.

4.1.1 Training data

Here we describe the parallel and monolingual datasets we have used for training the domain-adapted MT systems for World Cup tweets as well as their acquisition.

Parallel data. Although we did not have access to in-domain (soccer) parallel data for training, the Hub requires at least 10,000 sentence pairs to train a new system. Hence we decided to provide out-of-domain parallel data for training. Namely, the parallel corpora used for this include the corpora used for the general-domain baseline used for tourism (cf. Sect. 3.2.1) as well as our previous version of the web-crawled corpus of the tourism domain (Esplà-Gomis et al. 2014). All in all these corpora account for 440,264 sentence pairs.

Monolingual data. The biggest downside of the existing methodology for crawling monolingual data (cf. Sect. 3.1.1) is the way the boilerplate removal process works. It namely applies strong assumptions about the well-formedness of the text and ignores all content not obeying the orthographical rules of the language. This is why web corpora often contain a non-correspondingly low amount of user-generated content, i.e. content produced by non-experts.

When researchers are in need of user-generated content, they mostly switch to crawling specific web sites with content extractors being tailored to the specific information source, thereby ensuring that all relevant material, along with the metadata, will be collected. Optimal information sources are those that do not have to be crawled, but open APIs for retrieving their content. One good example of such an information source is Twitter.

While corpora of tweets for frequently used languages are built mostly by using just the Twitter Stream API and filtering out tweets not satisfying the language criterion, this approach is an overkill for smaller languages that make just tiny fractions of the whole Twitter production.

One possible approach to "fishing out" the Twitter data of smaller languages, given that the Twitter language identification algorithm nowadays is still quite inaccurate, is the one implemented in the TweetCaT tool (Ljubešić et al. 2014).[39] The tool takes as input high-frequency seed terms specific for the language of interest that are used to identify users tweeting in the desired language via the Twitter Search API. Once the users of interest are identified, their timelines are retrieved on which language identification is run. Tweets of all users passing the language criterion are retrieved in an iterative fashion since Twitter gives access only to the last 200 tweets. It was shown (Ljubešić et al. 2014) that by using this approach, for languages having just two million speakers, a reasonably sized corpus of 600 thousand tweets containing around 7 million words can be built in ten days, while running the tool for one year yields 8 million tweets that make up a corpus of around 100 million words.

For English, we use in-domain tweets (440,923) from the World Cup provided by Microsoft. These were tweets submitted by users to be translated and extracted from translation logs.

[39] https://github.com/nljubesi/tweetcat.

As there were no Croatian tweets in the translation logs, we crawled tweets (4,373,988) for this language using TweetCaT and a language identification tool focused on Twitter data and very similar South Slavic languages (Ljubešić and Kranjčić 2015). Thus, the monolingual data for Croatian, while belonging to the same genre as our target data (tweets), is not in-domain.

4.1.2 Tuning and test sets

Tuning and test sets are produced by means of crowdsourcing. Given the amount of work required for human translation and the short time length of the project, we decided to build tuning and test sets of the minimum suitable size. Pecina et al. (2012) showed that tuning sets of more than 400-600 sentence pairs do not improve translation quality (according to automatic evaluation metrics). Our tuning and test sets have 500 sentence pairs each, as that is the minimum size allowed by the Hub.

From the English tweets stored in the translation logs, we randomly selected two sets of 500 tweets each, to be used as tuning and test sets, respectively. These tuning and test datasets were then translated manually from English into Croatian by means of crowdsourcing on the CrowdFlower platform.[40] This crowdsourcing platform allows to configure the jobs using a number of options. We used a number of them with the aim of obtaining translations of a reasonable quality. The options that we used follow:

- Geography. One can select a set of countries from which workers are allowed to work on the job. We limited the country of workers to Croatia.
- Performance level. Contributors of the platform fall into three levels, according to their performance. Our jobs were limited to level 1 contributors, defined by Crowdflower as "high performance contributors who account for 60 % of monthly judgments and maintain a high level of accuracy".
- Language capability. One can restrict the contributors to work on the job by their language skills. One can select workers to be part of the so-called 'editorial crowd', defined as "highly competent in English spelling, syntax and grammar". Our jobs required workers from the editorial crowd. There are four additional restrictions than can be applied so that only workers "skilled in basic syntax and comprehension" of French, German, Portuguese and Spanish, respectively can be selected. As Croatian is not included we ignored these language-specific restrictions.
- Speed trap. If set, contributors are automatically removed from the job if they take less than a specified amount of time to complete a task. Our jobs contained tasks of 10 translations each and the time trap was set to 100 seconds. Hence if a worker were to take less than an average of 10 seconds to translate per sentence, they would automatically be removed from the job.

Figure 1 shows a snapshot of the translation task in CrowdFlower. Contributors are shown the instructions to complete the task followed by the task items. While each

[40] http://crowdflower.com/.

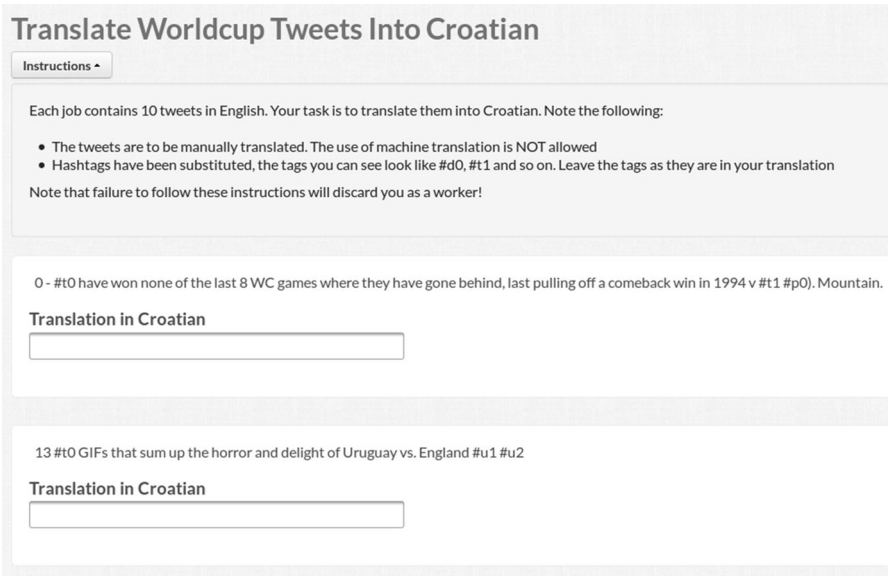

Fig. 1 Snapshot of the translation task in CrowdFlower

task comprises 10 translations, only the first two are shown in the snapshot due to space constraints.

The task (translation of 1000 tweets from English into Croatian) was completed by 12 contributors, with the amounts of translations produced by each contributor varying greatly (from 10 translations to 350), as shown in Fig. 2.

Contributors had the option to evaluate the task by providing scores (1 to 5, 1 being the worst and 5 the best) regarding different aspects: overall judgment, clarity of the instructions, ease of the job and pay level. Just over half of the contributors (7) provided such scores. The scores given by the contributors for overall judgment (4.57), clarity of instructions (4.87) and pay level (4.14) are in all cases well above 4 points. Conversely, the score given for ease of the job is the lowest (2.99) by quite a large margin. Our intuition is that most (if not all) of the contributors were probably non-professional translators and translating tweets is a hard task compared to translating other types of text, due e.g. to noise, lack of contextual information, etc.

In order to ensure that the crowdsourced translations were of the required level of quality for our purposes, a native speaker checked a random subset of 50 such translations. Out of these, 11 contained one or more significant errors, while the remaining 78 % of translations were acceptable. The most frequent translation errors were typographical in nature, accounting for 35 % of the errors. The second most frequent errors were incomplete translations (parts of tweets not being translated) which made up 27 % of all errors. Beside these errors, a frequent imperfection in the translation process was the loss of original letter casing which occurred in 12 of the 50 inspected translations.

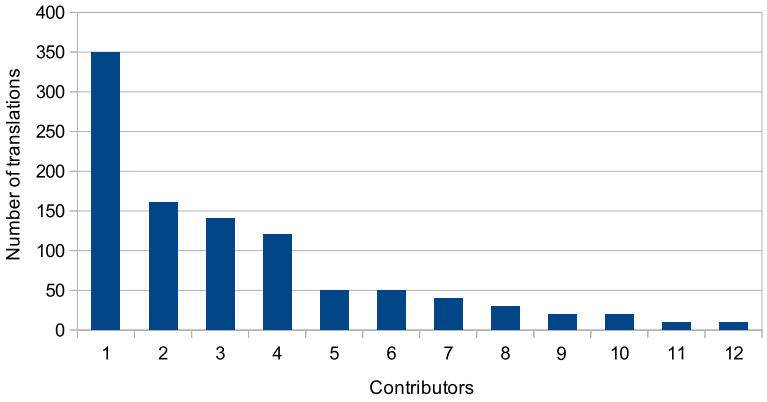

Fig. 2 Distribution of translations by contributor in the crowdsourcing task

In other words, 22 % of the translations do contain errors but this does not mean that they are completely wrong. Rather, the errors have a limited span measured in amount of consecutive words (specially the most frequent errors, typos), and therefore, while the references affected are not perfect, the effect of these errors for evaluating with automatic metrics are limited to short n-grams. Moreover, these limited-span errors should affect the evaluation of multiple MT systems in the same manner, thus not leading to any artificial bias (e.g. rewarding the translations of a given MT system while penalising the translations of another MT system). As for wrong casing, the metrics we use in the evaluation (BLEU and TER) being case-insensitive by default, this does not affect the evaluation.

The instructions (cf. Fig. 1) clearly indicated that the use of MT was not permitted, but we still checked that the contributors did not submit MT translations. While we did this manually as the subset of translations to be checked was small, there are recent approaches to do this checking automatically, e.g. (Rarrick et al. 2011), though of course they do not yield perfect accuracy.

All in all, while these translations as they are would not be adequate for dissemination per se, they are fit for our purposes, i.e. to be used as development and test sets.

4.2 Evaluation

4.2.1 Automatic evaluation

The MT systems for World Cup tweets are evaluated with the same automatic metrics that were used to evaluate the systems for the tourism domain (cf. Sect. 3.2.3).

Table 8 shows the results for the systems built for the direction Croatian to English. There are four systems; the baseline (general-domain system provided by the Hub) and three domain-adapted systems, which incorporate the different in-domain types of data: +inDev (tuning data, cf. Sect. 4.1.2) and +mono

(monolingual data, cf. Sect. 4.1.1) as well as both types combined (+inDev+inMono). The three domain-adapted systems are trained with the parallel dataset described in Sect. 4.1.1 and interpolated with the baseline.

All the three adapted systems outperform the baseline according to both BLEU and TER, with the improvements for BLEU being statistically significant ($p \geq 0.01$). The improvements in terms of BLEU (and TER) when adding different types of in-domain data are as follows: 1.91pt absolute (1.66 TER) and 5.53 relative (2.8) for tuning data, 7.14 absolute (8.20) and 20.69 relative (13.87) for monolingual data and 7.94 (5.11) and 23.01 relative (8.64) for both data types combined.

Table 9 shows the results for the systems built for the direction English to Croatian. The baseline is, again, the general-domain system provided by the Hub. For this translation direction, due to the fact that we have bigger amounts of monolingual data for adaptation (around 4 million compared to less than half a million for the opposite direction), we build two domain-adapted systems adding different portions of the monolingual data: 1 million (+inMono1M) and the whole dataset (+inMono4M). As in the opposite direction, the domain-adapted systems are trained with the parallel dataset described in Sect. 4.1.1 and interpolated with the baseline.

The use of an in-domain tuning set (+inDev) results in slightly higher BLEU (0.34 absolute, not significantly better) and TER (0.24). The addition of in-domain monolingual data leads to drops both in BLEU (−0.53) and TER (2.30). Only by

Table 8 Automatic evaluation scores for World Cup tweets (Croatian→English)

System	BLEU	TER
Baseline	0.3451	0.5914
+inDev	0.3642*	0.5748
+inMono	0.4165*	**0.5094**
+inDev+inMono	**0.4245***	0.5403

Results with * indicate significant improvement over the baseline with $p \geq 0.01$. The best result according to each metric (highest for BLEU and lowest for TER) is shown in bold

Table 9 Automatic evaluation scores for World Cup tweets (English→Croatian)

System	BLEU	TER
Baseline	0.2501	0.7098
+inDev	0.2535	0.7122
+inMono1M	0.2448	0.7328
+inDev+inMono1M	0.2682*	0.6970
+inDev+inMono4M	**0.2718***	**0.6904**

Results with * indicate significant improvement over the baseline with $p \geq 0.01$. The best result according to each metric (highest for BLEU and lowest for TER) is shown in bold

adding both types of in-domain data can we reach significant improvements, with the scores growing with the size of monolingual data, from 1.81 BLEU points absolute and 1.28 TER with 1 million, to 2.17 BLEU and 1.94 TER with 4.4 million.

4.2.2 Manual analysis

A Croatian native speaker inspected the translations produced by the baseline and the adapted MT system for a subset of the test set for both translation directions. The translations were inspected blindly (i.e. not knowing which system was which), and the native speaker was asked to select those sentences for which the translation by one system was considerably better than the translation by the other. Tables 10 and 11 provide samples of these translations for Croatian-to-English and English-to-Croatian, respectively.

In the first sample translation for Croatian-to-English (cf. Table 10), the adapted system solves two issues regarding word sense and out-of-vocabulary compared to the baseline: (1) "Osvajač" is wrongly translated as "invader" by the baseline while it is correctly translated as "winner" by the adapted system, and (2) "SP" is left untranslated by the baseline while it is correctly translated as "world cup" by the adapted system. In the second sentence, again the adapted system solves a word-sense issue compared to the baseline: "wc" is wrongly translated as "toilet" by the baseline while the adapted system produces the right translation, even if the locative case marker ("-u") is left as an attached postposition ("-in"). The third sentence regards a case where the adapted system is slightly worse than the baseline: while "arrive" preserves the core meaning of the source "doći", only "come" can be considered an accurate translation in this context.

In the first sample translation for English-to-Croatian (cf. Table 11), a domain-specific term ("World Cup") is accurately translated by the adapted system ("svjetsko prvenstvo") while the baseline leaves it untranslated. The second

Table 10 Sample translations for World Cup tweets (Croatian→English)

Source	**Osvajač SP** 1978 Mario kempes iz argentine
Reference	1978 **World Cup winner** Mario Kempes of Argentina
Baseline	**Invader SP** 1978 Mario kempes of Argentina
Adapted	1978 **WORLD CUP winner** Mario kempes of Argentina
Source	Nemogu dočekati da ih vidim ponižene u **wc-u**
Reference	Can't wait to see em humiliated in **wc**
Baseline	I can't wait to see them degraded in the **toilet** in
Adapted	I can't wait to see them humiliated in **wc-in**...
Source	Koliko daleko može Engleska **doći** na Svjetskom prvenstvu?
Reference	How far will England **progress** at the World Cup?
Adapted	How far can England **arrive** at the World Cup?
Baseline	How far can England **come** in the World Cup?

Table 11 Sample translation for World Cup tweets (English→Croatian)

Source	Adriana Lima Brings Futbol to a Sports Bar #p0 FIFA **World Cup**
Reference	Adriana Lima donosi futbol u sportski bar #p0 FIFA **svjetsko prvenstvo**
Baseline	Adriana Lima donosi Futbol sport bar #p0 FIFA **World Cup**
Adapted	Adriana Lima donosi Futbol u sportski bar #p0 **Svjetsko prvenstvo**
Source	Essential World Cup accessory for the 'Socceroos' **fan** in your life
Reference	Osnovni dodatak na Svjetskom prventsvu za svakog **pobornika** "Socceroos-a" u životu
Baseline	Neophodan dodatak svjetski kup za 'Socceroos' **ventilatora** u vašem životu
Adapted	Neophodan svjetski kup pribor za 'Socceroosi' **fan** u životu
Source	Fascinating insight into the **World Cup** from Squawka's Q&A with Andrew Cole
Reference	Fascinatan pogled na **Svjetsko Prvenstvo** iz Squawkionog Pitanja i odgovora sa Andrew Coleom
Adapted	Fascinantan uvid u **Svjetskom kupu** od Squawka's Q&A sa Andrew Cole
Baseline	Fascinantan uvid u **Svjetsko prvenstvo** od Squawka je Q&A sa Andrew Cole

sentence also regards the translation of a domain-specific term; "fan" is wrongly translated as "ventilatora" (i.e. considering a wrong word sense of the source word meaning "a device for creating a current of air by movement") by the baseline while the adapted system leaves it untranslated. While the latter results in an acceptable translation, it is not ideal as it would be considered substandard, whereas "pobornika" (the term used in the reference) or even "obožavatelja" would be preferred. The third sentence is an example of the adapted system performing worse than the baseline, but the issue is quite straightforward - both systems chose an accurate term in the translation ("kup" and "prvenstvo" are interchangable in this context), but only the phrase in the baseline is in the right case (accusative). Although this creates an ungrammatical construction, the underlying meaning is no less transparent because of it.

To summarise the findings of this analysis, we have provided qualitative evidence that the adapted system is better at translating terms that belong to the domain of our use-case. In particular, the adapted system solves issues posed by word sense and out-of-vocabulary compared to the baseline. While there are cases in which the translation produced by the adapted system is worse than the baseline, the degradation has little impact on understanding the meaning of the source sentence. Conversely, in those cases where the translation of the adapted system is better, the improvement delivered by this system is important to understand the source text.

5 Conclusions and future work

This paper presented a methodology to bring machine translation to under-resourced languages in a way that is both cost-effective and rapid. Our proposal relies on web crawling to automatically acquire parallel data to train SMT systems if any such

data can be found for the language pair and domain of interest. If that is not the case, we propose to use (1) crowdsourcing to translate small amounts of text (hundreds of sentences), which are then used to tune statistical SMT models, and (2) web crawling of vast amounts of monolingual data (millions of sentences), which are then used to build language models for SMT.

We demonstrated the usefulness of this methodology by applying it to two use-cases for Croatian, an under-resourced language that has gained relevance in the European context, since it recently attained official status in the European Union.

The first use-case regarded the tourism domain, given the strategic importance of this sector to Croatia's economy. For that, we exploited web crawling to acquire parallel data for this domain from the Internet. More specifically, we crawled parallel data from 20 web domains using two state-of-the-art crawlers and we explored how to combine the resulting in-domain crawled data with bigger amounts of publicly-available general-domain data. We then evaluated the resulting systems on a set of three additional tourism web domains. The adapted systems outperformed the baseline on two domains (the improvement ranging from 2.21 to 5.14 absolute points for translations into English and from 2.09 to 5.71 into Croatian, in terms of BLEU) and did not improve nor degrade on the third domain. We concluded that this variation has to do with tourism being a rather wide domain and the three web sites used as test sets covering different aspects of tourism. In any case, the adapted systems achieved higher coverage for all the three domains.

The second use-case had to do with tweets, due to the growing importance of social media. Specifically, we built MT systems to translate tweets from the 2014 edition of the soccer World Cup. This use-case was specially challenging because there are no sources of immediately appropriate parallel data available. We built domain-adapted systems by (1) translating small amounts of tweets to be used for tuning by means of crowdsourcing and (2) crawling vast amounts of monolingual tweets (in the range of millions). Our domain-adapted systems outperformed the baseline provided by Microsoft Bing by 7.94 points in terms of BLEU for Croatian-to-English and by 2.17 points for English-to-Croatian on a test set translated by means of crowdsourcing. As this test set was translated by non-professional translators, a native speaker checked a subset to make sure the translations were of the required level of quality. In addition, we carried out a complementary manual analysis that has provided qualitative evidence that the adapted systems result in notable improvements in translation quality, specifically with respect to the translation of domain-specific terms.

We have claimed that our methodology allows MT to be brought to under-resourced languages in a rapid and cost-effective manner. Now we briefly discuss why this is the case for the two parts of our methodology: crawling and crowdsourcing. Parallel and monolingual crawling are inherently cost-effective as they are fully automatic. As part of the Abu-MaTran project we have focused on making the crawlers used in this paper ready for commercial exploitation (Papavassiliou et al. 2014). As for crowdsourcing, we have recently compared it in terms of cost-effectiveness and rapidness with professional translation obtained from a language service provider. Crowdsourcing stands out as being an order of magnitude cheaper and considerably faster to obtain translations with: the

translation of 1,000 English sentences into Croatian took less than a working day through crowdsourcing while the turnaround offered by the language service provider amounted to 10 working days. That said, it is obvious that this figure does not provide the complete picture regarding the cost-effectiveness of crowdsourced versus professional translations as one has to also take into account the impact that the quality of the translated dataset will have down the pipeline, in our case for tuning MT systems. Such a study is part of our future work.

A further contribution of this paper regards the language resources that have been produced, namely (1) a parallel corpus for the tourism domain, released as a TMX translation memory,[41] and (2) parallel (English–Croatian) and monolingual (Croatian) tweets, available upon request.

While the use-cases presented in this paper are limited to Croatian for the sake of space and clarity, it is worth noting that our methodology is widely-applicable. In fact, the first approach (crawling of parallel data), has been applied to a number of other language pairs, such as English–Finnish (Rubino et al. 2015). As for the second approach, crowdsourcing to obtain tuning sets was performed for 4 other European languages (French, German, Greek and Portuguese) as part of Brazilator, while acquisition of monolingual tweets has been successfully used recently for 5 Iberian languages (Basque, Catalan, Galician, Portuguese and Spanish) in our participation at the TweetMT shared task (Toral et al. 2015).

In this work we built MT systems between an under-resourced language (Croatian) and, arguably, the best-resourced language (English). We would like to assess the feasibility of the approach we have put forward to build MT systems between an under-resourced language and other less well-resourced languages. In this vein, following on with Croatian and the tourism use-case, we plan to build MT systems between Croatian and German, Slovene and Italian. Our motivation has to do with the relevance of these languages and the fact that these language pairs are not well supported by third-party MT systems. More precisely, (1) these three languages account for over 50 % of incoming tourists in Croatia and (2) it seems that on-line MT systems covering these language pairs do not perform translation directly but use English as a pivot language.

Acknowledgments This research is supported by the European Union Seventh Framework Programme FP7/2007–2013 under grant agreement PIAP-GA-2012-324414 (Abu-MaTran) and by the ADAPT Centre for Digital Content Technology, funded under the SFI Research Centres Programme (Grant 13/RC/2106) and co-funded under the European Regional Development Fund.

References

Achananuparp, P., Hu, X., & Shen, X. (2008). The evaluation of sentence similarity measures. In I. Y. Song, J. Eder & T. Nguyen (Eds.), *Data warehousing and knowledge discovery* (Vol. 5182, pp. 305–316). Lecture Notes in Computer Science. Berlin, Heidelberg: Springer. doi:10.1007/978-3-540-85836-2_29.

Ambati, V., & Vogel, S. (2010). Can crowds build parallel corpora for machine translation systems? In *Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with*

[41] http://hdl.handle.net/11356/1049.

Amazon's mechanical turk (pp. 62–65). Los Angeles: Association for Computational Linguistics. http://www.aclweb.org/anthology/W10-0710.

Axelrod, A., He, X., & Gao, J. (2011). Domain adaptation via pseudo in-domain data selection. In *Proceedings of the 2011 conference on empirical methods in natural language processing* (pp. 355–362). Edinburgh, Scotland, UK: Association for Computational Linguistics. http://www.aclweb.org/anthology/D11-1033.

Baroni, M., Bernardini, S., Ferraresi, A., & Zanchetta, E. (2009). The wacky wide web: A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, *43*(3), 209–226. doi:10.1007/s10579-009-9081-4.

Bojar, O., Buck, C., Federmann, C., Haddow, B., Koehn, P., Leveling, J., et al. (2014). Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the ninth workshop on statistical machine translation* (pp. 12–58). Association for Computational Linguistics, Baltimore, Maryland, USA. http://www.aclweb.org/anthology/W/W14/W14-3302.

Boleda, G., Bott, S., Meza, R., Castillo, C., Badia, T., & López, V. (2006). In *Proceedings of the 2nd international workshop on Web as Corpus, chap CUCWeb: A Catalan corpus built from the Web*. http://aclweb.org/anthology/W06-1704.

Callison-Burch, C. (2009). Fast, cheap, and creative: Evaluating translation quality using amazon's mechanical turk. In *Proceedings of the 2009 conference on empirical methods in natural language processing* (pp. 286–295). EMNLP 2009, 6–7 August 2009, Singapore, A meeting of SIGDAT, a Special Interest Group of the ACL, ACL. http://www.aclweb.org/anthology/D09-1030.

Chakrabarti, S. (2003). *Mining the Web: Discovering knowledge from hypertext data*. Massachusetts:Morgan Kaufmann.

Esplà-Gomis, M., Klubička, F., Ljubešić, N., Ortiz-Rojas, S., Papavassiliou, V., & Prokopidis, P. (2014). Comparing two acquisition systems for automatically building an english-croatian parallel corpus from multilingual websites. In N. C. C. Chair, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk & S. Piperidis (Eds.), *Proceedings of the ninth international conference on language resources and evaluation (LREC'14)*. European Language Resources Association (ELRA), Reykjavik, Iceland.

Esplà-Gomis, M., & Forcada, M. L. (2010). Combining content-based and URL-based heuristics to harvest aligned bitexts from multilingual sites with bitextor. *The Prague Bulletin of Mathematical Linguistics*, *93*, 77–86.

Fišer, D., Tavčar, A., & Erjavec, T. (2014). slowcrowd: A crowdsourcing tool for lexicographic tasks. In: N. C. C. Chair, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk & S. Piperidis (Eds.),*Proceedings of the ninth international conference on language resources and evaluation (LREC'14)*. European Language Resources Association (ELRA), Reykjavik, Iceland.

Gao, Q., & Vogel, S. (2008). Parallel implementations of word alignment tool. In *Software engineering, testing, and quality assurance for natural language processing, association for computational linguistics* (pp. 49–57).

Graham, Y., Baldwin, T., Moffat, A., & Zobel, J. (2014). Is machine translation getting better over time? In *Proceedings of the 14th conference of the European Chapter of the Association for Computational Linguistics* (pp. 443–451), Gothenburg, Sweden. http://www.aclweb.org/anthology/E14-1047.

Hasler, E., Haddow, B., & Koehn, P. (2011). Margin infused relaxed algorithm for moses. *The Prague Bulletin of Mathematical Linguistics*, *96*, 69–78.

Hassan, H., & Menezes, A. (2013). Social text normalization using contextual graph random walks. In *Proceedings of the 51st annual meeting of the association for computational linguistics* (Vol.1: Long Papers, pp. 1577–1586), Association for Computational Linguistics, Sofia, Bulgaria. http://www.aclweb.org/anthology/P13-1155.

Heafield, K. (2011). Kenlm: Faster and smaller language model queries. In *Proceedings of the sixth workshop on statistical machine translation* (pp. 187–197). Association for Computational Linguistics.

Irvine, A., & Klementiev, A. (2010). Using mechanical turk to annotate lexicons for less commonly used languages. In *Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with Amazon's mechanical turk* (pp. 108–113). Association for Computational Linguistics, Stroudsburg, PA, USA. http://www.aclweb.org/anthology/W10-0717.pdf.

Kilgarriff, A., & Grefenstette, G. (2003). Introduction to the special issue on the web as corpus. *Computational Linguistics*, *29*(3), 333–347.

Klubička, F., & Ljubešić, N. (2014). Using crowdsourcing in building a morphosyntactically annotated and lemmatized silver standard corpus of croatian. In T. Erjavec & J. Ž. Gros (Eds.), *Language*

technologies: Proceedings of the 17th International Multiconference Information Society IS2014. Slovenia: Ljubljana.

Koehn, P. (2004). Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 conference on empirical methods in natural language processing* (pp. 388–395). EMNLP 2004, A meeting of SIGDAT, a Special Interest Group of the ACL, held in conjunction with ACL 2004, 25–26 July 2004, Barcelona, Spain, ACL. http://www.aclweb.org/anthology/W04-3250.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions* (pp. 177–180). http://dl.acm.org/citation.cfm?id=1557769.1557821.

Kohlschütter, C., Fankhauser, P., & Nejdl, W. (2010). Boilerplate detection using shallow text features. *Proceedings of the third ACM international conference on Web search and data mining* (pp. 441–450). New York, NY, USA.

Laranjeira, B., Moreira, V., Villavicencio, A., Ramisch, C., & José Finatto, M. (2014). Comparing the quality of focused crawlers and of the translation resources obtained from them. In *Proceedings of the ninth international conference on language resources and evaluation (LREC-2014)*. European Language Resources Association (ELRA).

Ljubešić, N., & Erjavec, T. (2011). hrWaC and slWac: Compiling Web Corpora for Croatian and Slovene. *Text, speech and dialogue—14th international conference, TSD 2011* (pp. 395–402). Pilsen: Czech Republic, Springer, Lecture Notes in Computer Science.

Ljubešić, N., & Klubička, F. (2014). bs, hr, srWaC—Web corpora of Bosnian, Croatian and Serbian. *Proceedings of the 9th Web as Corpus Workshop (WaC-9)* (pp. 29–35). Gothenburg, Sweden: Association for Computational Linguistics.

Ljubešić, N., & Kranjčić, D. (2015). Discriminating between closely related languages on twitter. *Informatica*, *39*(1), 1–8.

Ljubešić, N., Fišer, D., & Erjavec, T. (2014). TweetCaT: A tool for building twitter corpora of smaller languages. In: N. C. C. Chair, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk & S. Piperidis (Eds.), *Proceedings of the ninth international conference on language resources and evaluation (LREC'14)*. European Language Resources Association (ELRA), Reykjavik, Iceland.

Ma, X., & Liberman, M. (1999). Bits: A method for bilingual text search over the web. *Machine Translation Summit VII* (pp. 538–542), Singapore.

Munro, R. (2010). Crowdsourced translation for emergency response in haiti: the global collaboration of local knowledge. In *AMTA workshop on collaborative crowdsourcing for translation*, Denver, Colorado.

Munteanu, S. D., & Marcu, D. (2006). Extracting parallel sub-sentential fragments from non-parallel corpora. In *Proceedings of the 21st international conference on computational linguistics and 44th annual meeting of the association for computational linguistics* (pp. 81–88). Association for Computational Linguistics.

Nie, J. Y., Simard, M., Isabelle, P., & Durand, R. (1999). Cross-language information retrieval based on parallel texts and automatic mining of parallel texts from the Web. In *Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval* (pp. 74–81), ACM, Berkeley, California, USA, SIGIR'99.

Papavassiliou, V., Prokopidis, P., & Thurmair, G. (2013). A modular open-source focused crawler for mining monolingual and bilingual corpora from the web. In *Proceedings of the sixth workshop on building and using comparable corpora* (pp. 43–51), Association for Computational Linguistics, Sofia, Bulgaria. http://www.aclweb.org/anthology/W13-2506.

Papavassiliou, V., Prokopidis, P., Esplà-Gomis, M., & Ortiz-Rojas, S. (2014). D3.2. corpora acquisition software. Public deliverable, The Abu-MaTran Project (PIAP- GA-2012-324414).

Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics* (pp. 311–318). doi:10.3115/1073083.1073135.

Pecina, P., Toral, A., & van Genabith, J. (2012). Simple and effective parameter tuning for domain adaptation of statistical machine translation. *Proceedings of the 24th international conference on computational linguistics (Coling 2012), Coling 2012 Organizing Committee* (pp. 2209–2224). India: Mumbai.

Rarrick, S., Quirk, C., & Lewis, W. (2011). Mt detection in web-scraped parallel corpora. In *Proceedings of MT Summit XIII*, Asia-Pacific Association for Machine Translation. http://research.microsoft.com/pubs/153367/MT-Summit-Detection_Lewis_0819.pdf.

Rehm, G., & Uszkoreit, H. (2013). *META-NET Strategic Research Agenda for Multilingual Europe 2020 Incorporated*. Springer.

Resnik, P., & Smith, N. A. (2003). The Web as a parallel corpus. *Computational Linguistics*, *29*(3), 349–380.

Resnik, P., Buzek, O., Kronrod, Y., Hu, C., Quinn, A. J., & Bederson, B. B. (2013). Using targeted paraphrasing and monolingual crowdsourcing to improve translation. *ACM Trans Intell Syst Technol*, *4*(3), 38:1–38:21. doi:10.1145/2483669.2483671.

Rubino, R., Toral, A., Sánchez-Cartagena, V. M., Ferrández-Tordera, J., Ortiz Rojas, S., Ramírez-Sánchez, G., et al. (2014). Abu-matran at wmt 2014 translation task: Two-step data selection and rbmt-style synthetic rules. In *Proceedings of the ninth workshop on statistical machine translation* (pp. 171–177).

Rubino, R., Pirinen, T., Esplà-Gomis, M., Ljubešić, N., Ortiz Rojas, S., Papavassiliou, V., et al. (2015). Abu-matran at wmt 2015 translation task: Morphological segmentation and web crawling. In *Proceedings of the tenth workshop on statistical machine translation*, Association for Computational Linguistics, Lisbon, Portugal (pp. 184–191) http://aclweb.org/anthology/W15-3022.

Sennrich, R. (2012) Perplexity minimization for translation model domain adaptation in statistical machine translation. In *Proceedings of the 13th conference of the European chapter of the association for computational linguistics* (pp. 539–549). http://dl.acm.org/citation.cfm?id=2380816.2380881.

Sikes, R. (2007). Fuzzy matching in theory and practice. *MultiLingual*, *18*(6), 39–43.

Skadina, I., Vasiljevs, A., Skadins, R., Gaizauskas, R., & Tufis, D. (2010). Analysis and evaluation of comparable corpora for under resourced areas of machine translation. In *Proceedings of the 3rd workshop on building and using comparable corpora. Applications of parallel and comparable corpora in natural language engineering and the humanities* (pp. 6–14).

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., & Weischedel, R. (2006). A study of translation error rate with targeted human annotation. In *Proceedings of the association for machine translation in the Americas*.

Snow, R., O'Connor, B., Jurafsky, D., & Ng, A.Y. (2008). Cheap and fast - but is it good? evaluating non-expert annotations for natural language tasks. In *2008 conference on empirical methods in natural language processing, EMNLP 2008, Proceedings of the conference*, 25–27 October 2008, Honolulu, Hawaii, USA, A meeting of SIGDAT, a Special Interest Group of the ACL, ACL (pp. 254–263). http://www.aclweb.org/anthology/D08-1027.

Stolcke, A., Zheng, J., Wang, W., & Abrash, V. (2011). Srilm at sixteen: Update and outlook. In *Proceedings of IEEE automatic speech recognition and understanding workshop* (p. 5).

Suchomel, V., & Pomikálek, J. (2012). Efficient web crawling for large text corpora. In S. S. Adam Kilgarriff (Ed.), *Proceedings of the seventh Web as Corpus Workshop (WAC7)*, Lyon (pp. 39–43).

Talvensaari, T., Pirkola, A., Järvelin, K., Juhola, M., & Laurikkala, J. (2008). Focused web crawling in the acquisition of comparable corpora. *Inf Retr*, *11*(5), 427–445. doi:10.1007/s10791-008-9058-8.

Tiedemann, J. (2009). News from opus-a collection of multilingual parallel corpora with tools and interfaces. *Recent Advances in Natural Language Processing*, *5*, 237–248.

Toral, A., Rubino, R., Esplà-Gomis, M., Pirinen, T., Way, A., & Ramirez-Sanchez, G. (2014). Extrinsic evaluation of web-crawlers in machine translation: A case study on Croatian–English for the tourism domain. In *Proceedings of the 17th Conference of the European Association for Machine Translation (EAMT)* (pp. 221–224).

Toral, A., Wu, X., Pirinen, T., Qiu, Z., Bicici, E., & Du, J. (2015). Dublin city university at the tweetmt 2015 shared task. TweetMT@ SEPLN. In *Proceedings of the La Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN)*.

Tyers, F. M., & Alperen, M. S. (2010). South-east european times: A parallel corpus of balkan languages. In *Proceedings of the LREC workshop on exploitation of multilingual resources and tools for Central and (South-) Eastern European Languages* (pp. 49–53).

Varga, D., Németh, L., Halácsy, P., Kornai, A., Trón, V., & Nagy, V. (2005). Parallel corpora for medium density languages. *Recent advances in natural language processing* (pp. 590–596). Bulgaria: Borovets.

Wasala, A., Schäler, R., Buckley, J., Weerasinghe, R., & Exton, C. (2013). Building multilingual language resources in web localisation: A crowdsourcing approach. In I. Gurevych & J. Kim (Eds.),

The people's Web meets NLP, theory and applications of natural language processing (pp. 69–99). Berlin, Heidelberg: Springer. doi:10.1007/978-3-642-35085-6_3.

Zaidan, O. F., & Callison-Burch, C. (2011). Crowdsourcing translation: Professional quality from non-professionals. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies* (Vol. 1, pp. 1220–1229). Association for Computational Linguistics, Stroudsburg, PA, USA, HLT '11. http://dl.acm.org/citation.cfm?id=2002472.2002626.

Zbib, R., Markiewicz, G., Matsoukas, S., Schwartz, R. M., & Makhoul, J. (2013). Systematic comparison of professional and crowdsourced reference translations for machine translation. In *HLT-NAACL* (pp. 612–616).

Zhechev, V. (2012). Machine translation infrastructure and post-editing performance at autodesk. *AMTA 2012 workshop on post-editing technology and practice (WPTP 2012)* (pp. 87–96), San Diego, USA.