Cross-lingual Sentence Compression for Subtitles

Wilker Aziz and Sheila C. M. de Sousa
University of Wolverhampton
Stafford Street, WV1 1SB
Wolverhampton, UK
W.Aziz@wlv.ac.uk
sheilacastilhoms@gmail.com

Lucia Specia
Department of Computer Science
University of Sheffield
211 Portobello, S1 4DP
Sheffield, UK
L.Specia@sheffield.ac.uk

Abstract

We present an approach for translating subtitles where standard time and space constraints are modeled as part of the generation of translations in a phrase-based statistical machine translation system (PB-SMT). We propose and experiment with two promising strategies for jointly translating and compressing subtitles from English into Portuguese. The quality of the automatic translations is measured via the human post-editing of such translations so that they become adequate, fluent and compliant with time and space constraints. Experiments show that carefully selecting the data to tune the model parameters in the PB-SMT system already improves over an unconstrained baseline and that adding specific model components to guide the translation process can further improve the final translations under certain conditions.

1 Introduction

The increasing demand for fast and cheap generation of audiovisual content is pushing research and development in the automatic translation of subtitles. Several attempts have been made in recent years to translate subtitles automatically by using different Machine Translation (MT) approaches (see Section 2). Overall, it has been shown that translation tools can be very helpful in producing adequate and fluent translations of subtitles, yielding significant time (and cost) reductions when compared to manually translating subtitles. However, subtitling has other important constraints in addition to translation quality: translations must fit the space available on the screen and time slot so that they can be read by viewers. None of the existing approaches to translating subtitles considers these constraints.

When generating or translating subtitles from audio transcripts, human subtitlers should follow several conventions. Especially due to the advent of the DVD and the increasing use of smaller and smaller screens, norms and conventions in subtitling evolve quickly (Cintas and Remael, 2007). Currently, a norm of 40 characters per line, with two lines per screen, seems to be the most widely accepted for television screen, with common variants reaching up to 50 characters per line. Regarding time, a subtitle should remain in the screen for at least 1 second and at most 6 seconds if it contains two full lines.

It is important to make a distinction between translating directly from an audio transcript and translating from a subtitle in the source language. An audio transcript is likely to breach the time/space constraints simply because of the differences between human listening and reading rates. Therefore, some compression is usually necessary when generating monolingual subtitles. Producing subtitles in a second language however may require a second level of compression: even if the source language subtitle observes the time/space constraints, depending on the language-pair, a translation can be considerably longer than the source subtitle. This is particularly the case for translation between languages with significant structural differences such as English and the Romance languages. Additionally, lower quality source subtitles may already violate the time/space constraints.

We propose an approach for joint translation and compression that can be applied to translating from both transcripts and source language subtitles. We

experiment with the translation of English subtitles from a few popular TV series, taken from the OpenSubtitle section of the Opus corpus,[1] which contains both transcripts and translations by amateur subtitlers. As we discuss in Section 3.1, this corpus is particularly appealing for compression, since even manually produced translations violate the time/space constraints: 33.5% of the translations are longer than the recommended standard, with an average of 10 ± 7 additional characters.

Since compression may incur some loss of information, it should only be performed when necessary. The proposed approach *dynamically* defines the need for compression for every source subtitle and uses this information to bias the system to produce translations with the appropriate length. In order to do so, it exploits two main strategies for joint translation and compression in Statistical MT (SMT): the tuning of the SMT model parameters using a carefully selected dataset where space/time constraints are observed and the addition of explicit model components to guide the compression of the source subtitles via the selection of translation options that globally optimize the length of the target subtitle.

Our approach brings the following main contributions to previous work: (i) it takes advantage of the paraphrases that naturally occur in SMT systems, as opposed to resorting to artificially generated and potentially noisy paraphrases, or to the deep language processing techniques required by other sentence compression approaches; (ii) it is cross-lingual and therefore aims at ensuring that the target subtitle is compressed as required, as opposed to compressing the source subtitle, which could later get de-compressed as a consequence of an automatic translation, or directly compressing the target subtitle, which would require a sentence compression method for each target language; (iii) it dynamically identifies the need for compression as a function of the time/space available for the source subtitle, avoiding unnecessary compression, which could lead to inadequate translations; (iv) it yields a more efficient method for correcting both translation and compression in a single step. Additionally, it allows a more objective way of evaluating compression and translation based on these corrections, as opposed to commonly used subjective evaluation metrics based on human judgments for adequacy and fluency.

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

2 Related work

Several attempts have been made to translate subtitles automatically using Rule-Based (RBMT), Example-Based (EBMT), Statistical (SMT) and also Translation Memory (TM) tools. The first attempt by Popowich et al. (2000) use a number of preprocessing steps in order to improve the accuracy of an RBMT system and report 70% accuracy in a manual evaluation. In (Armstrong et al., 2006) an EBMT system is built using a corpus of subtitles. A comparison using a larger heterogeneous corpus including sentences from Europarl shows that a homogeneous setting leads to better translations. Volk (2008) uses an SMT system trained on a corpus of 5 million subtitle sentences and reports that SMT outputs can still be acceptable translations as long as they lie within 5 keystrokes from a reference translation. Sousa et al. (2011) presents an objective way of measuring translation quality for subtitles in terms of post-editing time. Experiments with a number of MT/TM approaches show that post-editing draft subtitles is consistently faster than translating them, and that post-editing time can be used to compare alternative TM/MT systems.

None of these approaches considers time/space constraints to generate or assess translations. On the other hand, a number of approaches have been proposed to compress subtitles. Most work is related to the ATraNoS[2] and MUSA[3] projects. These projects focused on the monolingual compression of audio transcripts based on handcrafted deletion and substitution rules and statistics extracted from a parallel corpus of original transcripts and their compressed version (Daelemans et al., 2004; Vandeghinste and Pan, 2004). Piperidis et al. (2004) use TM and RBMT systems to translate the compressed subtitles. Glickman et al. (2006) contrast context-independent and context-dependent models to replace words in subtitles by shorter synonyms. Context models based on distributional similarity provided useful estimates, but they resulted in an accuracy of only 60%.

Previous work on general monolingual text compression can also be mentioned (Knight and Marcu, 2000; Cohn and Lapata, 2009). However, these works do not model time/space constraints explicitly and are rather aimed at compressing every input sentence. A closely related work on

[2] http://atranos.esat.kuleuven.ac.be/
[3] http://sifnos.ilsp.gr/musa/

monolingual compression is that by Ganitkevitch et al. (2011). The authors generate sentential paraphrases from phrasal paraphrases using the syntax-based SMT framework with two additional features to explicitly model compression. However, a fixed, pre-defined compression rate is used for all input sentences, as opposed to a dynamic rate that depends on the input segment and the need for compression given time/space conventions.

3 Cross-lingual sentence compression

3.1 Motivation

In what follows, we illustrate the need for compression in subtitles taking as example the English-Portuguese language pair and manually translated subtitles from 3 recent episodes of 6 popular TV series, amounting to $8,144$ pairs of subtitles. Here a *subtitle* refers to the sequence of words appearing in one screen before an end-of-sentence marker.

For this analysis, we define the notion of *ideal length* as a function of the duration of the source language subtitle. More specifically, we consider the amount of time the source language subtitle is shown on the screen to define the ideal length of its translation. We follow the conventions in (Cintas and Remael, 2007) to identify the expected number of characters given a time slot and the frame rate. For example, if the source segment remains on the screen for 1 second, given the frame rate under consideration (25 frames per second), the number of characters in the translation (as well as in the source) subtitle should not exceed 17 characters.

By looking at the manually produced target side of this corpus, we found that 33.5% of the translations do not respect this ideal length, containing an average of 10 ± 7 additional characters. This may be a consequence of the fact that the source subtitles are sometimes too lengthy, since they were mostly generated by amateur subtitlers and are often merely transcriptions from the audio. In fact, 36.28% of the source subtitles are on average 8.85 ± 6.73 characters longer than expected. Nevertheless, 45.2% of the target subtitles are longer than the source subtitles by an average of 5 ± 4.5 characters, showing the natural difference in length between the two languages.

In order to show that standard MT tools will also fail to generate time/space compliant translations, we used Google Translate, a freely available translation tool, to translate the original subtitles. We found that 42.3% of the translations do not observe the ideal length, containing an average of 11.6 ± 8.7 additional characters. Interestingly, 63% of the translations are longer than the sources, with an average of 5.5 ± 4.3 additional characters. This seems to confirm the expected tendency: longer Portuguese translations are produced from English texts. It also shows that a general purpose MT system performs worse than the average amateur subtitler, producing even longer translations.

3.2 Rationale

We propose a joint approach to sentence translation and compression. The approach is based on a modification of the standard PB-SMT framework to include time/space constraints based on the input text. While in this paper we apply this approach to the translation of subtitles, it could be used for other applications that also require dynamically compressing translations.

In a nutshell, PB-SMT learns a bilingual dictionary of phrases (the *phrase table*) and their associated translation probabilities from a parallel corpus. It is not unusual that a given phrase in the source language is assigned a number of possible phrases in the target language, to accommodate for phenomena such as the ambiguity and paraphrasing in translation. During the translation process (*decoding*), the system chooses the translation that best fits the context based on a number of model components, among which are the phrase probability to indicate how common that translation is for the source phrase. Hence, a sizeable phrase table will contain many paraphrases, some of which will be shorter than others, particularly if this phrase table is generated from a corpus where the target language may require some compression. Different from previous work where monolingual paraphrases need to be externally generated, we focus on using these naturally occurring paraphrases in the phrase table. This approach has the advantages of providing a natural filter on the quality of the paraphrases as well as allowing the control of translation quality and compression rate in a single step. Additional paraphrases generated by any means could also be added to the phrase table, for example, following the method in (Ganitkevitch et al., 2011).

Compression may incur some loss of information. To prevent unnecessary and excessive compression, we treat compression as a less deterministic process by dynamically modeling the need for

compression as a function of the time/space constraints of each specific source segment. Our approach models time/space constraints by (i) adding model components to the Moses PB-SMT system (Koehn et al., 2007) to control the need of compression, and (ii) guiding the tuning process to prefer shorter translations. Each of these strategies is described in what follows.

3.3 Dynamic length penalty

Time and space constraints can be represented as a function of the time available for the source subtitle, as described in Section 3.1. In practice, these constraints will affect the length of the target subtitle, and therefore hereafter we refer to them as a *length constraint*. To incorporate this constraint into the Moses decoder, we define a character-based length penalty to adjust translations so that they meet this constraint as the difference between an *expected length* and the *length of the current translation hypothesis*. A length constraint is thus set individually for each segment to be translated.

As typical of PB-SMT, our length penalty component h_{lp} is incrementally computed in a per-phrase basis, that is:

$$h_{lp}(\bar{f}_1^K, \bar{e}_1^K, c) = \sum_{k=1}^{K} \hat{h}_{lp}(\bar{f}_k, \bar{e_k}, c)$$

where $\bar{f}_1^K$ denotes a source sentence f broken into K contiguous phrases, $\bar{e}_1^K$ denotes the K target phrases that make up the hypothesised translation e, and c is the expected length constraint.

The character length penalty models how much the translation hypothesis deviates from the expected length constraint c, that is: $h_{lp}(\bar{f}_1^K, \bar{e}_1^K, c) \equiv c - \text{length}(\bar{e}_1^K)$, where $\text{length}(x)$ is the number of characters of the sequence x including a space between every adjacent token. Every target segment spans a portion of text that is proportional to the source phrase being covered, therefore the lenght constraint can be adjusted to the segment level as $\hat{h}_{lp}$ in:

$$\hat{h}_{lp}(\bar{f}, \bar{e}, c) = c \times \frac{\text{length}(\bar{f})}{\text{length}(f)} - \text{length}(\bar{e})$$

where $\bar{f}$ is a source phrase, $\bar{e}$ is its hypothesised translation, and $\frac{\text{length}(\bar{f})}{\text{length}(f)}$ is a scaling factor that allows computing h_{lp} in a per-phrase basis.

In order to define the *expected length* constraint, c, for a given subtitle translation, we consider the following sources of information (in characters):

```
<s id="15" lp::ideal="23" lp::input="19"
    lp::min="19">I never felt this .</s>
```

Figure 1: Example of constraints.

- *lp::ideal* is the ideal length given the duration of the subtitle and the conventions in (Cintas and Remael, 2007), as outlined in Section 3.1;
- *lp::input* is the length of the source subtitle;
- *lp::min* is the minimum of the 2 above values.

We use the decoder's XML mark-up scheme to assign the length constraints to the source subtitles as shown in Figure 1. Based on these types of information we build two variations of our approach:

LP$_2$) Two model components: We add the constraint *lp::ideal* that represents a theoretically supported value based on the source subtitle duration. That is, with *lp::ideal* the system is trained to produce translations that can be read given the time slot of the source subtitle. However, sometimes a subtitle is shown for a long time, although it contains a very short string, and therefore *lp::ideal* can lead the decoder to produce translations that are longer than necessary simply because there is space left for it. To compensate for this issue, we add a second model component: *lp::input*, which may differ significantly from the former.

LP$_1$) One model component: An alternative approach adds a single model component, *lp::min*, which puts the two above mentioned components together. If the ideal length is longer, the model targets the input length. If instead the source subtitle is longer, the model targets the ideal length, aiming at producing a translation that observes the time and space constraints even though the original text is too lengthy.

3.4 Tuning process

Adding a new component to the model requires learning its contribution and its interaction with the other components. These model parameters are adjusted in a process often referred to as *tuning*. In this process a dataset for which gold translations are known is used to incrementally tune the model parameters towards improving a measure of quality, traditionally BLEU (Papineni et al., 2002).

In order to guide the model to select translation candidates that are likely to be good while complying with the length constraint, at tuning time,

when compression is necessary the model must reward phrases that are shorter. This can be done by i) biasing the evaluation metric towards shorter translations (Ganitkevitch et al., 2011); ii) using evaluation metrics that go beyond string matching, such as METEOR (Lavie and Agarwal, 2007), which also matches synonyms and paraphrases; iii) adding multiple reference translations that vary in length; or (iv) filtering the tuning set so that it contains only pairs of segments that comply with the length constraint. These strategies do not necessarily exclude each other, and can rather complement each other. An evaluation metric that rewards compression in general does not suit our application to subtitle translation, where segments should only be compressed when necessary. As for tuning with metrics like METEOR, the lack of quality in-domain Portuguese paraphrases for the subtitle domain is an issue.[4] Since having multiple references is expensive, we opted for filtering the tuning set so that it contains only subtitle pairs that comply with the length constraint, i.e. subtitles whose target sides are equal or shorter than the source sides and equal or shorter what is expected given the duration of the sources (ideal length).

The tuning of the proposed systems is performed using these controlled datasets and the standard MERT procedure in Moses.

4 Experimental settings

4.1 Corpus

We use the most recent version of the parallel corpus of subtitles distributed as part of the Opus Project (Tiedemann, 2009). The parallel corpus is made up of freely available fan-made subtitles[5] for a large variety of TV series, movies and other audiovisual materials. The English-Brazilian Portuguese portion of the corpus amounts to 28 million subtitle pairs. We selected the top 14 million pairs to build a translation model, which we judged to be enough for a PB-SMT system. The data is already automatically pre-processed: tokenized, truecased and word-aligned.

To generate the tuning and test sets we took the most recent episodes of three TV series from the same source of fan-made subtitles, which were not included in the Opus release: Dexter (D), How I Met Your Mother (H) and Terra Nova (T). A tuning set and a test set was created for each of these series. These were pre-processed as the training data using the tools and methods provided by Opus.

After filtering the tuning sets according to the restrictions defined in Section 3.4, the resulting sets contained 1900 (D), 1130 (H) and 1320 (T) English subtitles and their single reference translations. For testing the models, a test set containing 400 source subtitles from 2 recent episodes of each series (200 per episode, in their original sequence) was compiled, amounting to 1200 subtitles. No filtering was applied to the test sets.

4.2 Models and baselines

We experiment with the two variations of the length constrained models (Section 3.3), LP_2 and LP_1. Additionally, we consider three baselines:

Baseline 1 (B_1) Google Translate, an off-the-shelf SMT system known to be often used by amateur subtitlers to generate translations.

Baseline 2 (B_2) A PB-SMT system built using Moses and the same corpus as our proposed models, but tuned on unconstrained tuning sets (2000 subtitles per series), i.e., without selecting only subtitles that are compliant with time/space constraints.

Baseline 3 (B_3) A PB-SMT system built using Moses trained on the same corpus as our proposed models, and tuned on the same tuning set (only space/time compliant subtitles), but without any length penalty.

In all cases, the tuning of the systems was performed individually for each TV series.

4.3 Evaluation

In order to objectively evaluate our approach for both translation and compression, we have human translators post-editing the machine translations and collect various information from this process. Meta-information from post-editing has been successfully used in previous work to avoid the subjective nature of explicit scoring schemes (Specia, 2011; Sousa et al., 2011).

We use a post-editing tool[6] that gathers post-editing effort indicators on a per-subtitle basis, including keystrokes, time spent by translators to post-edit the subtitle and the actual post-edited

[4] Experiments with popular methods to generate paraphrases such as (Bannard and Callison-Burch, 2005) resulted in very poor paraphrases for this domain, most likely due to the highly non-literal nature of translations.

[5] http://www.opensubtitles.org

[6] http://pers-www.wlv.ac.uk/~in1676/pet/

subtitle (Aziz et al., 2012). The tool allows the specification of the length constraints and renders the tasks differently according to how well the translation observes time/space constraints. It uses colors to facilitate the visualization of the compression needs and indicates the number of characters that need to be compressed or remain to be used in the translation.

Each test set was given to human translators along with the post-editing tool and guidelines for translation correction and compression. Eight Brazilian Portuguese native speakers and fluent speakers of English with significant experience in English-Portuguese translation post-edited the MT outputs. We base our evaluation on the computation of automatic metrics such as HTER (Snover et al., 2006) between the machine translation and its post-edited version (Section 5).

4.3.1 Post-editing guidelines and task design

Guidelines and examples of translations were given to the translators and adapted after a pilot experiment with 150 subtitles post-edited per translators. In a nutshell, translators should minimally correct translations to make them fluent and adequate (style and consistency should be disregarded) and compress them when necessary. The following instructions summarise the guidelines:

- If the translation is fluent, adequate and follows the length constraint: do not post-edit it.
- If the translation observes the length constraint but is not fluent and/or is not adequate: perform the minimum necessary corrections to make it fluent and adequate, trying to keep it within the length limit as much as possible.
- If the translation is fluent and adequate but does not observe the length constraint: compress it towards the ideal length, preserving as much as possible the meaning of the source subtitle and keeping the translation fluent.

For the final evaluation, we split each test set in batches of 50 subtitles and distributed them among the eight translators in a way that the same annotator would never see the same source subtitle more than once and would post-edit target subtitles from randomly selected systems. Subtitles in a batch were shown in their original sequence so that the translators could rely on previous and posterior contexts for both compression and correction. Annotators post-edited 200 subtitles a day.

5 Results

In this section we discuss the performance of the systems in terms of automatic metrics computed using the human post-edited translations for the 3 test sets (i.e. D, H and T). Note that translation quality and compression are jointly evaluated. We use the *multeval* toolkit (Clark et al., 2011) to score the systems and test them for statistical significance.[7] We report BLEU, TER and the hypothesis length over the reference length in percentage (LENGTH).[8]

To make the reference set we put together all post-edited translations that were length compliant. In addition, references longer than the ideal length were kept only if no compliant paraphrase was produced by any of the annotators (we observed only 5 of those cases).

For all test sets (Tables 1 to 3), systems trained using subtitles data outperform B_1 (Google) by a large margin, which shows that parallel subtitles provide phrase pairs that are naturally better/shorter than those typical of general purpose parallel data. Additionally, simply constraining the tuning set to space compliant subtitles (B_3) already yields significant improvement over B_2 (unconstrained tuning).

System	BLEU ↑	TER ↓	LENGTH
B_3	61.7	30.3	116.0
B_1	43.6^{-}	63.6^{-}	156.5^{-}
B_2	58.1^{-}	35.7^{-}	127.3^{-}
LP_2	62.2	29.5	115.5
LP_1	$64.6^{\dagger}$	$28.3^{\dagger}$	115.8

Table 1: Metric scores for the dataset D: p-values are computed with respect to B_3.

Table 1 shows that LP_1 outperforms B_3 in terms of both BLEU and TER. It suggests that the length penalty contributes to producing subtitles that require less post-editing. On the other hand, Tables 2 and 3 show no statistically significant differences between B_3 and the systems with length penalties (except for LP_2 on test set H). Moreover, while Table 2 suggests that LP_1 produces translations slightly longer than necessary (LP_1's LENGTH is larger than B_3's), Table 3 shows that LP_2 compresses the translations slightly more than

[7] Hereafter †, ‡and *denote results that are significantly better than a baseline ($p < 0.01$, 0.05 and 0.10, respectively). $^{-}$, $^{=}$and $^{\equiv}$denote results that are significantly worse than a baseline ($p < 0.01$, 0.05 and 0.10, respectively).

[8] The closer a system is to 100%, the closer its outputs are in length to what human translators produce as final subtitles.

B_3 (LP_2's LENGTH is smaller than B_3's). These somewhat conflicting results suggest that characteristics of the dataset may affect the generalization power of the length penalty (see Table 4).

System	BLEU ↑	TER ↓	LENGTH
B_3	70.8	20.0	108.5
B_1	47.0^{-}	52.8^{-}	144.3^{-}
B_2	60.6^{-}	31.3^{-}	126.9^{-}
LP_2	70.3	$21.0^{\equiv}$	109.1
LP_1	70.6	20.7	$110.0^{=}$

Table 2: Metric scores for the dataset H: p-values are computed with respect to B_3.

System	BLEU ↑	TER ↓	LENGTH
B_3	60.0	33.8	120.2
B_1	41.0^{-}	63.1^{-}	152.1^{-}
B_2	52.7^{-}	44.1^{-}	135.8^{-}
LP_2	60.4	33.4	$119.3^{\ddagger}$
LP_1	$57.9^{=}$	34.8	119.8

Table 3: Metric scores for the dataset T: p-values are computed with respect to B_3.

Table 4 shows the distribution of the input and ideal lengths in our test sets. While the average input length is almost constant across datasets, the other two constraints show that the datasets H and T require more compression than D.

Finally, although over 36% of the source subtitles in our datasets are not time/space compliant, Table 5 shows that our systems decrease this non-compliance in 10% by either filtering the tuning set (B_3) or modelling length penalties (LP_2 and LP_1). Moreover, even if the automatic compression is not enough, models LP_2 and LP_1 make manual compression easier, as the lower percentage of malformed PEs suggests.

5.1 Further improvements

The human post-editing produced 5 reference translations for a set of 1200 sentences (400 per series). We used these sentences altogether to experiment with an alternative tuning approach: a tuning set with explicit human-made, mostly length compliant, paraphrases (see Section 3.4). In Table 6 the

Set	lp::input	lp::ideal	lp::min
D	28.82 ± 15.43	36.99 ± 14.40	26.03 ± 12.86
H	28.40 ± 13.81	33.25 ± 13.77	25.97 ± 12.20
T	28.34 ± 15.22	30.14 ± 11.47	24.61 ± 11.93

Table 4: Average length constraints (in number of characters) in source subtitles.

Malformed	B_1	B_2	B_3	LP_2	LP_1
MT	44.15	34.41	25.40	24.57	25.65
PE	8.50	9.08	7.0	5.65	5.65

Table 5: Percentage of MT and human post-edited translations that are longer than the ideal length.

superscript *m* denotes a system that was retrained using this multiple-reference tuning set. We kept B_3 in the comparison to measure whether the new tuning set brings up any significant performance gain.

System	BLEU ↑	TER ↓	LENGTH
B_3^m	63.2	26.8	103.8
B_3	$62.1^{\equiv}$	27.0	106.1^{-}
LP_2^m	63.8	$26.0^{\ddagger}$	103.3^{*}
LP_1^m	64.1^{*}	$25.9^{\dagger}$	103.6

Table 6: Metric scores for a dataset of 600 unseen sentences (200 from each series) post-edited by 4 translators following the guidelines presented in Section 4.3.1: p-values are computed with respect to B_3^m.

Adding multiple references in the tuning phase yileds consistent and significant gains in performance. The new systems significantly outperform B_3 in terms of both BLEU and TER. Furthermore, B_3 is the system which is the farthest from the 100% LENGTH, that is, the improved systems produce translations that are closer in length to what human translators produce as final subtitles, with LP_2^m having the closest length. Finally, LP_1^m and LP_2^m are both significantly better than B_3^m in terms of TER.

6 Conclusions

We have presented an approach to successfully compress subtitles in a multilingual scenario by i) adequately choosing tuning data and ii) giving a PB-SMT model the capability of controlling the length of its hypotheses. Moreover, we have shown that in the presence of reliable, often shorter, paraphrases in the tuning set, more promising length-constrained models can be produced.

In future work we plan to further evaluate the model by trying to isolate edits due to translation quality from edits due to compression needs. Besides we must consider other indicators of post-editing effort such as post-editing time and keystrokes.

References

Armstrong, Stephen, Colm Caffrey, Marian Flanagan, Dorothy Kenny, Minako O'Hagan, and Andy Way. 2006. Leading by Example: Automatic Translation of Subtitles via EBMT. *Perspectives*, 3(14):163–184.

Aziz, Wilker, Sheila Castilho Monteiro de Sousa, and Lucia Specia. 2012. PET: a tool for post-editing and assessing machine translation. In *The Eighth International Conference on Language Resources and Evaluation*, LREC '12, Istanbul, Turkey, May. To appear.

Bannard, Colin and Chris Callison-Burch. 2005. Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 597–604, Stroudsburg, PA, USA. Association for Computational Linguistics.

Cintas, Jorge Díaz and Aline Remael. 2007. *Audiovisual Translation: Subtitling. Translation Practice Explained.* St Jerome Publishing.

Clark, Jonathan, Chris Dyer, Alon Lavie, and Noah Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of the Association for Computational Lingustics*.

Cohn, Trevor and Mirella Lapata. 2009. Sentence compression as tree transduction. *Journal of Artificial Intelligence Research*, 34:637–674.

Daelemans, Walter, Anja Höthker, and Erik Tjong Kim Sang. 2004. Automatic sentence simplification for subtitling in dutch and english. In *4th International Conference on Language Resources and Evaluation*, pages 1045–1048, Lisbon, Portugal.

Ganitkevitch, Juri, Chris Callison-Burch, Courtney Napoles, and Benjamin Van Durme. 2011. Learning sentential paraphrases from bilingual parallel corpora for text-to-text generation. In *Conference on Empirical Methods in Natural Language Processing*, pages 1168–1179, Edinburgh, Scotland, UK., July.

Glickman, Oren, Ido Dagan, Mikaela Keller, Samy Bengio, and Walter Daelemans. 2006. Investigating lexical substitution scoring for subtitle generation. In *10th Conference on Computational Natural Language Learning*, pages 45–52, New York City, New York.

Knight, Kevin and Daniel Marcu. 2000. Statistics-based summarization - step one: Sentence compression. In *17th National Conference of the American Association for Artificial Intelligence*, pages 703–710, Austin, USA.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics: Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic.

Lavie, A. and A. Agarwal. 2007. METEOR: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments. In *2nd Workshop on Statistical Machine Translation*, pages 228–231, Prague.

Papineni, K., S. Roukos, T. Ward, and W. Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Morristown.

Piperidis, Stelios, Iason Demiros, Prokopis Prokopidis, P. Vanroose, A. Hoethker, Walter Daelemans, E. Sklavounou, M. Konstantinou, and Y. Karavidas. 2004. Multimodal multilingual resources in the subtitling process. In *4th International Conference on Language Resources and Evaluation*, pages 205–208, Lisbon, Portugal.

Popowich, Fred, Paul Mcfetridge, Davide Turcato, and Janine Toole. 2000. Machine translation of closed captions. *Machine Translation*, 15:311–341.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *7th Conference of the Association for Machine Translation in the Americas*, pages 223–231.

Sousa, Sheila C. M., Wilker Aziz, and Lucia Specia. 2011. Assessing the post-editing effort for automatic and semi-automatic translations of DVD subtitles. In *Recent Advances in Natural Language Processing Conference*, Hissar, Bulgaria.

Specia, Lucia. 2011. Exploiting objective annotations for measuring translation post-editing effort. In *15th Conference of the European Association for Machine Translation*, pages 73–80, Leuven.

Tiedemann, Jörg. 2009. News from OPUS - A Collection of Multilingual Parallel Corpora with Tools and Interfaces. In Nicolov, N., K. Bontcheva, G. Angelova, and R. Mitkov, editors, *Recent Advances in Natural Language Processing (vol V)*, pages 237–248. John Benjamins.

Vandeghinste, Vincent and Yi Pan. 2004. Sentence compression for automated subtitling: A hybrid approach. In *ACL-04 Workshop Text Summarization Branches Out*, pages 89–95, Barcelona, Spain.

Volk, Martin. 2008. The automatic translation of film subtitles. a machine translation success story? In *Resourceful Language Technology: Festschrift in Honor of Anna*, volume 7, Uppsala, Sweden.