The UPV-PRHLT Combination System for WMT 2010

Jesús González-Rubio and **Jesús Andrés-Ferrer** and **Germán Sanchis-Trilles**
Guillem Gascó and **Pascual Martínez-Gómez** and **Martha-Alicia Rocha**
Joan-Andreu Sánchez and **Francisco Casacuberta**
Instituto Tecnológico de Informática
Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
{jegonzalez|jandres|gsanchis}@dsic.upv.es
{ggasco|pmartinez|mrocha}@dsic.upv.es
{jandreu|fcn}@dsic.upv.es

Abstract

UPV-PRHLT participated in the System Combination task of the Fifth Workshop on Statistical Machine Translation (WMT 2010). On each translation direction, all the submitted systems were combined into a consensus translation. These consensus translations always improve translation quality of the best individual system.

1 Introduction

The UPV-PRHLT approach to MT system combination is based on a refined version of the algorithm described in (González-Rubio and Casacuberta, 2010), with additional information to cope with hypotheses of different quality.

In contrast to most of the previous approaches to combine the outputs of multiple MT systems (Bangalore et al., 2001; Jayaraman and Lavie, 2005; Matusov et al., 2006; Schroeder et al., 2009), which are variations over the ROVER voting scheme (Fiscus, 1997), we consider the problem of computing a consensus translation as the problem of modelling a set of string patterns with an adequate prototype. Under this framework, the translation hypotheses of each of the MT systems are considered as individual patterns in a set of string patterns. The *(generalised) median string*, which is the optimal prototype of a set of strings (Fu, 1982), is the chosen prototype to model the set of strings.

2 System Combination Algorithm

The median string of a set is defined as the string that minimises the sum of distances to the strings in the set. Therefore, defining a distance between strings is the primary problem to deal with.

The most common definition of distance between two strings is the Levenshtein distance, also known as edit distance (ED). This metric computes the optimal sequence of edit operations (insertions, deletions and substitutions of words) needed to transform one string into the other. The main problem with the ED is its dependence on the length of the compared strings. This fact led to the definition of a new distance whose value is independent from the length of the strings compared. This *normalised edit distance* (NED) (Vidal et al., 1995) is computed by averaging the number of edit operations by the length of the edit path. The experimentation in this work was carried out using the NED.

2.1 Median String

Given a set $E = \mathbf{e}_1, \ldots, \mathbf{e}_n, \ldots, \mathbf{e}_N$ of translation hypotheses from N MT systems, let Σ be the vocabulary in the target language and Σ^* be the free monoid over that vocabulary ($E \subseteq \Sigma^*$). The median string of the set E (noted as $\mathcal{M}(E)$) can be formally defined as:

$$\mathcal{M}(E) = \underset{\mathbf{e}' \in \Sigma^*}{\operatorname{argmin}} \sum_{n=1}^{N} \left[w_n \cdot \mathcal{D}(\mathbf{e}', \mathbf{e}_n) \right] \ , \qquad (1)$$

where $\mathcal{D}$ is the distance used to compare two strings and the value w_n, $1 \leq n \leq N$ weights the contribution of the hypothesis n to the sum of distances, and therefore, it denotes the significance of hypothesis n in the computation of the median string. The value w_n can be seen as a measure of the "quality" of hypothesis n.

Computing the median string is a NP-Hard problem (de la Higuera and Casacuberta, 2000), therefore we can only build approximations to the median string by using several heuristics. In this work, we follow two different approximations: the *set median* string (Fu, 1982) and the *approximate median* string (Martínez et al., 2000).

Proceedings of the Joint 5th Workshop on Statistical Machine Translation and MetricsMATR, pages 296–300,
Uppsala, Sweden, 15-16 July 2010. ©2010 Association for Computational Linguistics

2.2 Set Median String

The most straightforward approximation to the median string corresponds to the search of a *set median* string. Under this approximation, the search is constrained to the strings in the given input set. The set median string can be informally defined as the most "centred" string in the set. The set median string of the set E (noted as $\mathcal{M}_s(E)$) is given by:

$$\mathcal{M}_s(E) = \underset{\mathbf{e}' \in E}{\text{argmin}} \sum_{n=1}^{N} \left[w_n \cdot \mathcal{D}(\mathbf{e}', \mathbf{e}_n) \right] \ . \tag{2}$$

The set median string can be computed in polynomial time (Fu, 1982; Juan and Vidal, 1998). Unfortunately, in some cases, the set median may not be a good approximation to the median string. For example, in the extreme case of a set of two strings, either achieves the minimum accumulated distance to the set. However, the set median string is a useful initialisation in the computation of the approximate median string.

2.3 Approximate Median String

A good approximation to efficiently compute the median string is proposed in (Martínez et al., 2000). To compute the approximate median string of the set E, the algorithm starts with an initial string $\mathbf{e}$ which is improved by successive refinements in an iterative process. This iterative process is based on the application of different edit operations over each position of the string $\mathbf{e}$ looking for a reduction of the accumulated distance to the strings in the set. Algorithm 1 describes this iterative process.

The initial string can be a random string or a string computed from the set E. Martinez et al. (2000) proposed two kinds of initial strings: the set median string of E and a string computed by a greedy algorithm, both of them obtained similar results. In this work, we start with the set median string in the initialisation of the computation of the approximate median string of the set E. Over this initial string we apply the iterative procedure described in Algorithm 1 until there is no improvement. The final median string may be different from the original hypotheses.

The computational time cost of Algorithm 1 is linear with the number of hypotheses in the combination, and usually only a moderate number of iterations is needed to converge.

```
For each position $i$ in the string $\mathbf{e}$:
  1. Build alternatives:
     Substitution: Make $\mathbf{x} = \mathbf{e}$. For each word $a \in \Sigma$:
       • Make $\mathbf{x}'$ the result string of substituting the $i^{th}$ word of $\mathbf{x}$ by $a$.
       • If the accumulated distance of $\mathbf{x}'$ to $E$ is lower than the accumulated distance from $\mathbf{x}$ to $E$, then make $\mathbf{x} = \mathbf{x}'$.
     Deletion: Make $\mathbf{y}$ the result string of deleting the $i^{th}$ word of $\mathbf{e}$.
     Insertion: Make $\mathbf{z} = \mathbf{e}$. For each word $a \in \Sigma$:
       • Make $\mathbf{z}'$ the result of inserting $a$ at position $i$ of $\mathbf{e}$.
       • If the accumulated distance from $\mathbf{z}'$ to $E$ is lower than the accumulated distance from $\mathbf{z}$ to $E$, then make $\mathbf{z} = \mathbf{z}'$.
  2. Choose an alternative:
       • From the set $\{\mathbf{e}, \mathbf{x}, \mathbf{y}, \mathbf{z}\}$ take the string $\mathbf{e}'$ with less accumulated distance to $E$. Make $\mathbf{e} = \mathbf{e}'$.
```

Algorithm 1: Iterative process to refine a string $\mathbf{e}$ in order to reduce its accumulated distance to a given set E.

3 Experiments

Experiments were conducted on all the 8 translation directions cz→en, en→cz, de→en, en→de, es→en, en→es, fr→en and en→fr. Some of the entrants to the shared translation task submit lists of n-best translations, but, in our experience, if a large number of systems is available, using n-best translations does not allow to obtain better consensus translations than using single best translations, but raises computation time significantly. Consequently, we compute consensus translations only using the single best translation of each individual MT system. Table 1 shows the number of systems submitted and gives an overview of the test corpus on each translation direction. The number of running words is the average number of running words in the test corpora, from where the consensus translations were computed; the vocabulary is the merged vocabulary of these test corpora. All the experiments were carried out with the true-cased, detokenised version of the tuning and test corpora, following the WMT 2010 submission guidelines.

3.1 Evaluation Criteria

We will present translation quality results in terms of *translation edit rate* (TER) (Snover et al., 2006) and *bilingual evaluation understudy* (BLEU) (Pa-

	cz→en	en→cz	de→en	en→de	es→en	en→es	fr→en	en→fr
Submitted systems	6	11	16	12	8	10	14	13
Avg. Running words	45K	37K	47K	41K	47K	47K	47K	49K
Distinct words	24K	51K	38K	40K	23K	30K	27K	37K

Table 1: Number of systems submitted and main figures of test corpora on each translation direction. K stands for thousands of elements.

pineni et al., 2002). TER is computed as the number of edit operations (insertions, deletions and substitutions of single words and shifts of word sequences) to convert the system hypothesis into the reference translation. BLEU computes a geometric mean of the precision of n-grams multiplied by a factor to penalise short sentences.

3.2 Weighted Sum of Distances

In section 2, we define the median string of a set as the string which minimises a weighted sum of distances to the strings in the set (Eq. (1)). The weights w_n in the sum can be tuned. We compute a weight value for each MT system as a whole, i.e. all the hypotheses of a given MT system share the same weight value. We study the performance of different sets of weight looking for improvements in the quality of the consensus translations. These weight values are derived from different automatic MT evaluation measures:

- BLEU score of each system.
- 1.0 minus TER score of each system.
- Number of times the hypothesis of each system is the best TER-scoring translation.

We estimate these scores on the tuning corpora. A normalisation is performed to transform these scores into the range $[0.0, 1.0]$. After the normalisation, a weight value of 0.0 is assigned to the lowest-scoring hypothesis, i.e. the lowest-scoring hypothesis is not taking into account in the computation of the median string.

3.3 System Combination Results

Our framework to compute consensus translations allows multiple combinations varying the median string algorithm or the set of weight values used in the weighted sum of distances. To assure the soundness of our submission to the WMT 2010 system combination task, the experiments on the tuning corpora were carried out in a leaving-one-out fashion dividing the tuning data into 5 parts and averaging translation results over these 5 partitions. On each of the experiments, 4 of the partitions are devoted to obtain the weight values for the weighted sum of distances while BLEU and TER scores are calculated on the consensus translations of the remaining partition.

Table 2 shows, on each translation direction, the performance of the consensus translations on the tuning corpora. The consensus translations were computed with the set median string and the approximated median string using different sets of weight values: Uniform, all weights are set to 1.0, BLEU-based weights, TER-based weights and oracle-based weights. In addition, we display the performance of the best of the individual MT systems for comparison purposes. The number of MT systems combined for each translation direction is displayed between parentheses.

On all the translation directions under study, the consensus translations improved the results of the best individual systems. E.g. TER improved from 66.0 to 63.3 when translating from German into English. On average, the set median strings performed better than the best individual system, but its results were always below the performance of the approximate median string. The use of weight values computed from MT quality measures allows to improve the quality of the consensus translation computed. Specially, oracle-based weight values that, except for the cz→en task, always perform equal or better than the other sets of weight values. We have observed that no improvements can be achieved with uniform weight values; it is necessary to penalise low quality hypotheses.

To compute our primary submission to the WMT 2010 system combination task we choose the configurations that obtain consensus translations with highest BLEU score on the tuning corpora. The approximate median string using oracle-based scores is the chosen configuration for all translation directions, except on the cz→en translation direction for which TER-based weights performed better. As our secondary submission we

		Single	Set median				Approximated median			
		best	Uniform	Bleu	Ter	Oracle	Uniform	Bleu	Ter	Oracle
cz→en (6)	BLEU	17.6	16.5	17.8	18.2	17.6	17.1	**18.5**	**18.5**	18.0
	TER	64.5	68.7	67.6	65.2	64.5	67.0	65.9	65.4	**64.4**
en→cz (11)	BLEU	11.4	10.1	10.9	10.7	**11.0**	10.1	10.7	10.7	**11.0**
	TER	75.3	75.1	74.3	74.2	74.2	73.9	73.4	73.3	**73.0**
de→en (16)	BLEU	19.0	19.0	19.1	19.3	19.7	19.3	19.8	19.9	**20.1**
	TER	66.0	65.4	65.2	65.0	64.6	64.4	63.4	63.4	**63.3**
en→de (12)	BLEU	11.9	11.6	11.7	11.7	**12.0**	11.6	11.8	11.8	**12.0**
	TER	74.3	74.1	74.1	74.0	73.7	72.7	72.9	72.7	**72.6**
es→en (8)	BLEU	23.2	23.0	23.3	23.2	23.6	23.1	23.9	23.8	**24.2**
	TER	60.2	60.6	59.8	59.8	59.5	60.0	59.2	59.4	**59.1**
en→es (10)	BLEU	23.3	23.0	23.3	23.4	24.0	23.6	23.8	23.8	**24.2**
	TER	60.1	60.1	59.9	59.7	59.5	59.0	59.1	58.9	**58.6**
fr→en (14)	BLEU	23.3	22.9	23.2	23.2	23.4	23.4	23.8	23.8	**23.9**
	TER	61.1	61.2	60.9	60.9	60.7	60.6	60.0	60.1	**59.9**
en→fr (13)	BLEU	22.7	23.4	23.5	23.6	**23.8**	23.3	23.6	23.7	**23.8**
	TER	62.3	61.0	61.0	60.9	60.6	60.2	60.1	**60.0**	**60.0**

Table 2: Consensus translation results (case-sensitive) on the tuning corpora with the set median string and the approximate median string using different sets of weights: Uniform, BLEU-based, TER-based and oracle-based. The number of systems being combined for each translation direction is in parentheses. Best consensus translation scores are in bold.

	Best		Secondary		Primary	
	BLEU	TER	BLEU	TER	BLEU	TER
cz→en	18.2	63.9	18.3	66.7	19.0	65.1
en→cz	10.8	75.2	11.3	73.6	11.6	71.9
de→en	18.3	66.6	19.1	65.4	19.6	63.9
en→de	11.6	73.4	11.7	72.9	11.9	71.7
es→en	24.7	59.0	24.9	58.9	25.0	58.2
en→es	24.3	58.4	24.9	57.3	25.3	56.3
fr→en	23.7	59.7	23.6	59.8	23.9	59.4
en→fr	23.3	61.3	23.6	59.9	24.1	58.9

Table 3: Translation scores (case-sensitive) on the test corpora of our primary and secondary submissions to the WMT 2010 system combination task.

chose the set median string using the same set of weight values chosen for the primary submission.

We compute MT quality scores on the WMT 2010 test corpora to verify the results on the tuning data. Table 3 displays, on each translation direction, the results on the test corpora of our primary and secondary submissions and of the best individual system. These results confirm the results on the tuning data. On all translation directions, our submissions perform better than the best individual systems as measured by BLEU and TER.

4 Summary

We have studied the performance of two consensus translation algorithms that based in the computation of two different approximations to the median string. Our algorithms use a weighted sum of distances whose weight values can be tuned. We show that using weight values derived from automatic MT quality measures computed on the tuning corpora allow to improve the performance of the best individual system on all the translation directions under study.

Acknowledgements

This paper is based upon work supported by the EC (FEDER/FSE) and the Spanish MICINN under the MIPRCV "Consolider Ingenio 2010" program (CSD2007-00018), the iTransDoc (TIN2006-15694-CO2-01) and iTrans2 (TIN2009-14511) projects and the FPU scholarship AP2006-00691. This work was also supported by the Spanish MITyC under the erudito.com (TSI-020110-2009-439) project and by the Generalitat Valenciana under grant Prometeo/2009/014 and scholarships BFPI/2007/117 and ACIF/2010/226 and by the Mexican government under the PROMEP-DGEST program.

References

S. Bangalore, G. Bodel, and G. Riccardi. 2001. Computing consensus translation from multiple machine translation systems. In *IEEE Workshop on ASRU*, pages 351–354.

C. de la Higuera and F. Casacuberta. 2000. Topology of strings: Median string is np-complete. *Theoretical Computer Science*, 230:39–48.

J. Fiscus. 1997. A post-processing system to yield reduced word error rates: Recogniser output voting error reduction (rover).

K.S. Fu. 1982. *Syntactic Pattern Recognition and Applications*. Prentice Hall.

J. González-Rubio and F. Casacuberta. 2010. On the use of median string for multi-source translation. In *Proceedings of 20th International Conference on Pattern Recognition*, Istambul, Turkey, May 27-28.

S. Jayaraman and A. Lavie. 2005. Multi-engine machine translation guided by explicit word matching. In *Proc. of EAMT*, pages 143–152.

A. Juan and E. Vidal. 1998. Fast Median Search in Metric Spaces. In *Proc. of SPR*, volume 1451 of *Lecture Notes in Computer Science*, pages 905–912.

C. D. Martínez, A. Juan, and F. Casacuberta. 2000. Use of Median String for Classification. In *Proc. of ICPR*, volume 2, pages 907–910.

E. Matusov, N. Ueffing, and H-Ney. 2006. Computing consensus translation from multiple machine translation systems using enhanced hypotheses alignment. In *Proc. of EACL*, pages 33–40.

K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proc. of ACL*, pages 311–318.

J. Schroeder, T. Cohn, and P. Koehn. 2009. Word lattices for multi-source translation. In *Proc. of EACL*, pages 719–727.

M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. 2006. A study of TER with targeted human annotation. In *Proc. of AMTA*, pages 223–231.

E. Vidal, A. Marzal, and P. Aibar. 1995. Fast computation of normalized edit distances. *IEEE Transactions on PAMI*, 17(9):899–902.