An Empirical Study on Computing Consensus Translations from Multiple Machine Translation Systems

Wolfgang Macherey
Google Inc.
1600 Amphitheatre Parkway
Mountain View, CA 94043, USA
wmach@google.com

Franz Josef Och
Google Inc.
1600 Amphitheatre Parkway
Mountain View, CA 94043, USA
och@google.com

Abstract

This paper presents an empirical study on how different selections of input translation systems affect translation quality in system combination. We give empirical evidence that the systems to be combined should be of similar quality and need to be almost uncorrelated in order to be beneficial for system combination. Experimental results are presented for composite translations computed from large numbers of different research systems as well as a set of translation systems derived from one of the best-ranked machine translation engines in the 2006 NIST machine translation evaluation.

1 Introduction

Computing consensus translations from the outputs of multiple machine translation engines has become a powerful means to improve translation quality in many machine translation tasks. Analogous to the ROVER approach in automatic speech recognition (Fiscus, 1997), a composite translation is computed by voting on the translation outputs of multiple machine translation systems. Depending on how the translations are combined and how the voting scheme is implemented, the composite translation may differ from any of the original hypotheses. While elementary approaches simply select for each sentence one of the original translations, more sophisticated methods allow for combining translations on a word or a phrase level.

Although system combination could be shown to result in substantial improvements in terms of translation quality (Matusov et al., 2006; Sim et al., 2007), not every possible ensemble of translation outputs has the potential to outperform the primary translation system. In fact, an adverse combination of translation systems may even deteriorate translation quality. This holds to a greater extent, when the ensemble of translation outputs contains a significant number of translations produced by low performing but highly correlated systems.

In this paper we present an empirical study on how different ensembles of translation outputs affect performance in system combination. In particular, we will address the following questions:

- *To what extent can translation quality benefit from combining systems developed by multiple research labs?*
  Despite an increasing number of translation engines, most state-of-the-art systems in statistical machine translation are nowadays based on implementations of the same techniques. For instance, word alignment models are often trained using the GIZA++ toolkit (Och and Ney, 2003); error minimizing training criteria such as the *Minimum Error Rate Training* (Och, 2003) are employed in order to learn feature function weights for log-linear models; and translation candidates are produced using phrase-based decoders (Koehn et al., 2003) in combination with n-gram language models (Brants et al., 2007).

  All these methods are established as *de facto* standards and form an integral part of most statistical machine translation systems. This, however, raises the question as to what extent translation quality can be expected to improve when similarly designed systems are combined.

- *How can a set of diverse translation systems be built from a single translation engine?*
  Without having access to different translation

Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 986–995, Prague, June 2007.

engines, it is desirable to build a large number of diverse translation systems from a *single* translation engine that are useful in system combination. The mere use of N-best lists and word lattices is often not effective, because N-best candidates may be highly correlated, thus resulting in small diversity compared to the first best hypothesis. Therefore, we need a canonical way to build a large pool of diverse translation systems from a *single* translation engine.

- *How can an ensemble of translation outputs be selected from a large pool of translation systems?*
  Once a large pool of translation systems is available, we need an effective means to select a small ensemble of translation outputs for which the combined system outperforms the best individual system.

These questions will be investigated on the basis of three approaches to system combination: (i) an MBR-like candidate selection method based on *BLEU correlation matrices*, (ii) confusion networks built from word sausages, and (iii) a novel two-pass search algorithm that aims at finding consensus translations by reordering bags of words constituting the consensus hypothesis.

Experiments were performed on two Chinese-English text translation corpora under the conditions of the large data track as defined for the 2006 NIST machine translation evaluation (MT06). Results are reported for consensus translations built from system outputs provided by MT06 participants as well as systems derived from one of the best-ranked translation engines.

The remainder of this paper is organized as follows: in Section 2, we describe three combination methods for computing consensus translations. In Sections 3.1 and 3.2, we present experimental results on combining system outputs provided by MT06 participants. Section 3.3 shows how correlation among translation systems affects performance in system combination. In Section 3.4, we discuss how a single translation engine can be modified in order to produce a large number of diverse translation systems. First experimental results using a greedy search algorithm to select a small ensemble of translation outputs from a large pool of canonically built translation systems are reported. A summary presented in Section 4 concludes the paper.

2 Methods for System Combination

System combination in machine translation aims to build a composite translation from system outputs of multiple machine translation engines. Depending on how the systems are combined and which voting scheme is implemented, the consensus translation may differ from any of the original candidate translations. In this section, we discuss three approaches to system combination.

2.1 System Combination via Candidate Selection

The easiest and most straightforward approach to system combination simply returns one of the original candidate translations. Typically, this selection is made based on translation scores, confidence estimations, language and other models (Nomoto, 2004; Paul et al., 2005). For many machine translation systems, however, the scores are often not normalized or may even not be available, which makes it difficult to apply this technique. We therefore propose an alternative method based on "correlation matrices" computed from the BLEU performance measure (Papineni et al., 2001).

Let $\mathbf{e}_1, ..., \mathbf{e}_M$ denote the outputs of M translation systems, each given as a sequence of words in the target language. An element of the BLEU correlation matrix $\mathbf{B} = (b_{ij})$ is defined as the sentence-based BLEU score between a candidate translation $\mathbf{e}_i$ and a pseudo-reference translation $\mathbf{e}_j$ $(i, j = 1, ..., M)$:

$$b_{ij} = \mathrm{BP}(\mathbf{e}_i, \mathbf{e}_j) \cdot \exp\left(\frac{1}{4}\sum_{n=1}^{4} \log \rho_n(\mathbf{e}_i, \mathbf{e}_j)\right). \tag{1}$$

Here, BP denotes the brevity penalty factor with ρ_n designating the n-gram precisions.

Because the BLEU score is computed on a sentence rather than a corpus-level, n-gram precisions are capped by the maximum over $\frac{1}{2\cdot|\mathbf{e}_i|}$ and ρ_n in order to avoid singularities, where $|\mathbf{e}_i|$ is the length of the candidate translation [1].

Due to the following properties, $\mathbf{B}$ can be interpreted as a correlation matrix, although the term does not hold in a strict mathematical sense: (i) $b_{ij} \in [0, 1]$; (ii) $b_{ij} = 1.0 \iff \mathbf{e}_i = \mathbf{e}_j$; (iii) $b_{ij} = 0.0 \iff \mathbf{e}_i \cap \mathbf{e}_j = \varnothing$, i.e., b_{ij} is zero if and only if none of the words which constitute $\mathbf{e}_i$ can be found

[1] Note that for non-zero n-gram precisions, ρ_n is always larger than $\frac{1}{2\cdot|\mathbf{e}|}$.

in $\mathbf{e}_j$ and vice versa. The BLEU correlation matrix is in general, however, not symmetric, although in practice, $||b_{ij} - b_{ji}||$ is typically negligible.

Each translation system m is assigned to a *system prior weight* $\omega_m \in [0, 1]$, which reflects the performance of system m relatively to all other translation systems. If no prior knowledge is available, ω_m is set to $1/M$.

Now, let $\boldsymbol{\omega} = (\omega_1, ..., \omega_M)^\top$ denote a vector of system prior weights and let $\mathbf{b}_1, ..., \mathbf{b}_M$ denote the row vectors of the matrix $\mathbf{B}$. Then the translation system with the highest consensus is given by:

$$\begin{aligned} \mathbf{e}^* &= \mathbf{e}_{m^*} \quad \text{with} \\ m^* &= \underset{\mathbf{e}_m}{\operatorname{argmax}} \left\{ \boldsymbol{\omega}^\top \cdot \mathbf{b}_m \right\} \end{aligned} \tag{2}$$

The candidate selection rule in Eq. (2) has two useful properties:

- The selection does not depend on scored translation outputs; the mere target word sequence is sufficient. Hence, this technique is also applicable to rule-based translation systems [2].
- Using the components of the row-vector $\mathbf{b}_m$ as feature function values for the candidate translation $\mathbf{e}_m$ ($m = 1, ..., M$), the system prior weights $\boldsymbol{\omega}$ can easily be trained using the Minimum Error Rate Training described in (Och, 2003).

Note that the candidate selection rule in Eq. (2) is equivalent to re-ranking candidate translations according to the *Minimum Bayes Risk* (MBR) decision rule (Kumar and Byrne, 2004), provided that the system prior weights are used as estimations of the posterior probabilities $p(\mathbf{e}|\mathbf{f})$ for a source sentence $\mathbf{f}$. Due to the proximity of this method to the MBR selection rule, we call this combination scheme *MBR-like system combination*.

2.2 ROVER-Like Combination Schemes

ROVER-like combination schemes aim at computing a composite translation by voting on confusion networks that are built from translation outputs of multiple machine translation engines via an iterative application of alignments (Fiscus, 1997). To accomplish this, one of the original candidate translations, e.g. $\mathbf{e}_m$, is chosen as the primary translation hypothesis, while all other candidates $\mathbf{e}_n$ ($n \neq m$) are aligned with the word sequence of the primary translation. To limit the costs when aligning a permutation of the primary translation, the alignment metric should allow for small shifts of contiguous word sequences in addition to the standard edit operations *deletions*, *insertions*, and *substitutions*. These requirements are met by the *Translation Edit Rate* (TER) (Snover et al., 2006):

$$\text{TER}(\mathbf{e}_i, \mathbf{e}_j) := \frac{\text{Del} + \text{Ins} + \text{Sub} + \text{Shift}}{|\mathbf{e}_j|} \tag{3}$$

The outcome of the iterated alignments is a word transition network which is also known as *word sausage* because of the linear sequence of correspondence sets that constitute the network. Since both the order and the elements of a correspondence set depend on the choice of the primary translation, each candidate translation is chosen in turn as the primary system. This results in a total of M word sausages that are combined into a single super network. The word sequence along the cost-minimizing path defines the composite translation.

To further optimize the word sausages, we replace each system prior weight ω_m with the l_p-norm over the normalized scalar product between the weight vector $\boldsymbol{\omega}$ and the row vector $\mathbf{b}_m$:

$$\omega'_m := \frac{(\boldsymbol{\omega}^\top \cdot \mathbf{b}_m)^\ell}{\sum_{\tilde{m}} (\boldsymbol{\omega}^\top \cdot \mathbf{b}_{\tilde{m}})^\ell}, \qquad \ell \in [0, +\infty) \tag{4}$$

As ℓ approaches $+\infty$, $\omega'_m = 1$ if and only if system m has the highest consensus among all input systems; otherwise, $\omega'_m = 0$. Thus, the word sausages are able to emulate the candidate selection rule described in Section 2.1. Setting $\ell = 0$ yields uniform system prior weights, and setting $\mathbf{B}$ to the unity matrix provides the original prior weights vector. Word sausages which take advantage of the refined system prior weights are denoted by *word sausages+*.

2.3 A Two-Pass Search Algorithm

The basic idea of the two-pass search algorithm is to compute a consensus translation by reordering words that are considered to be constituents of the final consensus translation.

Initially, the two-pass search is given a repository of candidate translations which serve as pseudo references together with a vector of system prior weights. In the first pass, the algorithm uses a greedy strategy to determine a *bag of words* which minimizes the *position-independent word error rate* (PER). These words are considered to be

[2] This property is not exclusive to this combination scheme but also holds for the methods discussed in Sections 2.2 and 2.3.

constituents of the final consensus translation. The greedy strategy implicitly ranks the constituents, i.e., words selected at the beginning of the first phase reduce the PER the most and are considered to be more important than constituents selected in the end. The first pass finishes when putting further constituents into the bag of words does not improve the PER.

The list of constituents is then passed to a second search algorithm, which starts with the empty string and then expands all active hypotheses by systematically inserting the next unused word from the list of constituents at different positions in the current hypothesis. For instance, a partial consensus hypothesis of length l expands into $l + 1$ new hypotheses of length $l + 1$. The resulting hypotheses are scored with respect to the TER measure based on the repository of weighted pseudo references. Low-scoring hypotheses are pruned to keep the space of active hypotheses small. The algorithm will finish if either no constituents are left or if expanding the set of active hypotheses does not further decrease the TER score. Optionally, the best consensus hypothesis found by the two-pass search is combined with all input translation systems via the MBR-like combination scheme described in Section 2.1. This refinement is called *two-pass+*.

2.4 Related Work

Research on multi-engine machine translation goes back to the early nineties. In (Robert and Nirenburg, 1994), a semi-automatic approach is described that combines outputs from three translation systems to build a consensus translation. (Nomoto, 2004) and (Paul et al., 2005) used translation scores, language and other models to select one of the original translations as consensus translation. (Bangalore et al., 2001) used a multiple string alignment algorithm in order to compute a single confusion network, on which a consensus hypothesis was computed through majority voting. Because the alignment procedure was based on the Levenshtein distance, it was unable to align translations with significantly different word orders. (Jayaraman and Lavie, 2005) tried to overcome this problem by using confidence scores and language models in order to rank a collection of synthetic combinations of words extracted from the original translation hypotheses. Experimental results were only reported for the METEOR metric (Banerjee and Lavie, 2005). In (Matusov et al., 2006), pairwise word alignments of the original translation hypotheses were estimated for an enhanced statistical alignment model in order to explicitly capture word re-ordering. Although the proposed method was not compared with other approaches to system combination, it resulted in substantial gains and provided new insights into system combination.

Table 1: *Corpus statistics for two Chinese-English text translation sets: ZHEN-05 is a random selection of test data used in NIST evaluations prior to 2006; ZHEN-06 comprises the NIST portion of the Chinese-English evaluation data used in the 2006 NIST machine translation evaluation.*

corpus		**Chinese**	**English**
ZHEN-05	sentences	2390	
	chars / words	110647	67737
ZHEN-06	sentences	1664	
	chars / words	64292	41845

3 Experimental Results

Experiments were conducted on two corpora for Chinese-English text translations, the first of which is compiled from a random selected subset of evaluation data used in the NIST MT evaluations up to the year 2005. The second data set consists of the NIST portion of the Chinese-English data used in the MT06 evaluation and comprises 1664 Chinese sentences collected from broadcast news articles (565 sentences), newswire texts (616 sentences), and news group texts (483 sentences). Both corpora provide 4 reference translations per source sentence. Table 1 summarizes some corpus statistics.

For all experiments, system performance was measured in terms of the IBM-BLEU score (Papineni et al., 2001). Compared to the NIST implementation of the BLEU score, IBM-BLEU follows the original definition of the brevity penalty (BP) factor: while in the NIST implementation the BP is always based on the length of the shortest reference translation, the BP in the IBM-BLEU score is based on the length of the reference translation which is closest to the candidate translation length. Typically, IBM-BLEU scores tend to be smaller than NIST-BLEU scores. In the following, BLEU always refers to the IBM-BLEU score.

Except for the results reported in Section 3.2, we used uniform system prior weights throughout all experiments. This turned out to be more stable when combining different sets of translation systems and helped to improve generalization.

Table 2: *BLEU scores and brevity penalty (BP) factors determined on the ZHEN-06 test set for primary systems together with consensus systems for the MBR-like candidate selection method obtained by combining each three adjacent systems with uniform system prior weights. Primary systems are sorted in descending order with respect to their BLEU score. The 95% confidence intervals are computed using the bootstrap re-sampling normal approximation method (Noreen, 1989).*

combination	**primary system**			**consensus**				**oracle**	
	BLEU	**CI 95%**	**BP**	**BLEU**	**Δ**	**BP**	**pair-CI 95%**	**BLEU**	**BP**
01, 02, 03	32.10	(±0.88)	0.93	32.97	(+0.87)	0.92	[+0.29, +1.46]	38.54	0.94
01, 15, 16*	32.10	(±0.88)	0.93	23.55	(-8.54)	0.92	[-9.29, -7.80]	33.55	0.95
02, 03, 04	31.71	(±0.90)	0.96	31.55	(-0.16)	0.92	[-0.65, +0.29]	37.23	0.95
03, 04, 05	29.59	(±0.88)	0.87	29.55	(-0.04)	0.88	[-0.53, +0.41]	35.55	0.92
03, 04, 06*	29.59	(±0.88)	0.87	29.83	(+0.24)	0.90	[-0.29, +0.71]	35.69	0.93
04, 05, 06	27.70	(±0.87)	0.94	28.52	(+0.82)	0.91	[+0.15, +1.49]	34.67	0.94
05, 06, 07	27.05	(±0.81)	0.88	28.21	(+1.16)	0.92	[+0.63, +1.66]	33.89	0.94
05, 06, 08*	27.05	(±0.81)	0.88	28.47	(+1.42)	0.91	[+0.95, +1.95]	34.18	0.93
06, 07, 08	27.02	(±0.76)	0.92	28.12	(+1.10)	0.94	[+0.59, +1.59]	33.87	0.95
07, 08, 09	26.75	(±0.79)	0.97	27.79	(+1.04)	0.94	[+0.52, +1.51]	33.54	0.95
08, 09, 10	26.41	(±0.81)	0.92	26.78	(+0.37)	0.94	[-0.07, +0.86]	32.47	0.96
09, 10, 11	25.05	(±0.84)	0.90	24.96	(-0.09)	0.94	[-0.59, +0.46]	30.92	0.97
10, 11, 12	23.48	(±0.68)	1.00	24.24	(+0.76)	0.94	[+0.27, +1.30]	30.08	0.96
11, 12, 13	23.26	(±0.74)	0.95	24.05	(+0.79)	0.92	[+0.40, +1.23]	29.56	0.93
12, 13, 14	22.38	(±0.78)	0.87	22.68	(+0.30)	0.89	[-0.28, +0.95]	28.58	0.91
13, 14, 15	22.13	(±0.72)	0.89	21.29	(-0.84)	0.90	[-1.33, -0.33]	26.61	0.92
14, 15, 16	17.42	(±0.66)	0.93	18.45	(+1.03)	0.92	[+0.45, +1.56]	23.30	0.95
15	17.20	(±0.64)	0.91	—	—	—	—	—	—
16	15.21	(±0.63)	0.96	—	—	—	—	—	—

3.1 Combining Multiple Research Systems

In a first experiment, we investigated the effect of combining translation outputs provided from different research labs. Each translation system corresponds to a primary system submitted to the NIST MT06 evaluation [3]. Table 2 shows the BLEU scores together with their corresponding BP factors for the primary systems of 16 research labs (site names were anonymized). Primary systems are sorted in descending order with respect to their BLEU score. Table 2 also shows the consensus translation results for the MBR-like candidate selection method. Except where marked with an asterisk, all consensus systems are built from the outputs of three adjacent systems. While only few combined systems show a degradation, the majority of all consensus translations achieve substantial gains between 0.2% and 1.4% absolute in terms of BLEU score on top of the best individual (primary) system. The column CI provides 95% confidence intervals for BLEU scores with respect to the primary system baseline using the bootstrap re-sampling normal approximation method (Noreen, 1989). The column "pair-CI" shows 95% confidence intervals relative to the primary system using the paired bootstrap re-sampling method (Koehn, 2004). The principle of the paired bootstrap method is to create a large number of corresponding virtual test sets by consistently selecting candidate translations with replacement from both the consensus and the primary system. The confidence interval is then estimated over the differences between the BLEU scores of corresponding virtual test sets. Improvements are considered to be significant if the left boundary of the confidence interval is larger than zero.

Oracle BLEU scores shown in Table 2 are computed by selecting the best translation among the three candidates. The oracle scores might indicate a larger potential of the MBR-like selection rule, and further gains could be expected if the candidate selection rule is combined with confidence measures.

Table 2 shows that it is important that all translation systems achieve nearly equal quality; combining high-performing systems with low-quality translations typically results in clear performance losses compared to the primary system, which is the case when combining, e.g., systems 01, 15, and 16.

[3] For more information see `http://www.nist.gov/speech/tests/mt/mt06eval_official_results.html`

Table 3: *BLEU scores and brevity penalty (BP) factors determined on the ZHEN-06 test set for the combination of multiple research systems using the MBR-like selection method with uniform and trained system prior weights. Prior weights are trained using 5-fold cross validation. The 95% confidence intervals realtive to uniform weights are computed using the paired bootstrap re-sampling method (Koehn, 2004).*

# systems	combination	uniform		ω opt. on dev.			ω opt. on test	
		BLEU	BP	BLEU	BP	pair-CI 95%	BLEU	BP
3	01 – 03	32.98	0.92	33.03	0.93	[-0.23, +0.34]	33.60	0.93
4	01 – 04	**33.44**	0.93	33.46	0.93	[-0.26, +0.29]	34.97	0.94
5	01 – 05	33.07	0.92	33.14	0.93	[-0.29, +0.43]	34.33	0.93
6	01 – 06	32.86	0.92	33.53	0.93	[+0.26, +1.08]	34.43	0.93
7	01 – 07	33.08	0.93	33.51	0.93	[+0.04, +0.82]	34.49	0.93
8	01 – 08	33.12	0.93	33.47	0.93	[-0.06, +0.75]	34.50	0.94
9	01 – 09	33.15	0.93	33.22	0.93	[-0.35, +0.51]	34.68	0.93
10	01 – 10	33.01	0.93	33.59	0.94	[+0.18, +0.96]	34.79	0.94
11	01 – 11	32.84	0.94	33.40	0.94	[+0.13, +0.98]	34.76	0.94
12	01 – 12	32.73	0.93	33.49	0.94	[+0.34, +1.18]	34.83	0.94
13	01 – 13	32.71	0.93	33.54	0.94	[+0.39, +1.26]	34.91	0.94
14	01 – 14	32.66	0.93	**33.69**	0.94	[+0.58, +1.47]	34.97	0.94
15	01 – 15	32.47	0.93	33.57	0.94	[+0.63, +1.57]	34.99	0.94
16	01 – 16	32.51	0.93	33.62	0.94	[+0.62, +1.59]	35.00	0.94

3.2 Non-Uniform System Prior Weights

As pointed out in Section 2.1, a useful property of the MBR-like system selection method is that system prior weights can easily be trained using the Minimum Error Rate Training (Och, 2003). In this section, we investigate the effect of using non-uniform system weights for the combination of multiple research systems. Since for each research system, only the first best translation candidate was provided, we used a five-fold cross validation scheme in order to train and evaluate the system prior weights. For this purpose, all research systems were consistently split into five random partitions of almost equal size. The partitioning procedure was document preserving, i.e., sentences belonging to the same document were guaranteed to be assigned to the same partition. Each of the five partitions played once the role of the evaluation set while the other four partitions were used as development data to train the system prior weights. Consensus systems were computed for each held out set using the system prior weights estimated on the respective development sets. The combination results determined on all held out sets were then concatenated and evaluated with respect to the ZHEN-06 reference translations. Table 3 shows the results for the combinations of up to 16 research systems using either uniform or trained system prior weights. System 01 achieved the highest BLEU score on all five constellations of development partitions and is therefore the primary system to which all results in Table 3 compare. In comparison to uniform weights, consensus translations using trained weights are more robust toward the integration of low performing systems into the combination scheme. The best combined system obtained with trained system prior weights (01-14) is, however, not significantly better than the best combined system using uniform weights (01-04), for which the 95% confidence interval yields $[-0.17, 0.66]$ according to the paired bootstrap re-sampling method.

Table 3 also shows the theoretically achievable BLEU scores when optimizing the system prior weights on the held out data. This provides an upper bound to what extent system combination might benefit if an ideal set of system prior weights were used.

3.3 Effect of Correlation on System Combination

The degree of correlation among input translation systems is a key factor which decides whether translation outputs can be combined such a way that the overall system performance improves. Correlation can be considered as a reciprocal measure of diversity: if the correlation is too large ($> 90\%$), there will be insufficient diversity among the input systems and the consensus system will at most be able to only marginally outperform the best indi-

Table 4: *BLEU scores obtained on ZHEN-05 with uniform prior weights and a 10-way system combination using the MBR-like candidate selection rule, word sausages, and the two-pass search algorithm together with their improved versions "sausages+" and "two-pass+", respectively for different sample sizes of the FBIS training corpus.*

sampling	**primary**			**mbr-like**		**sausages**		**sausages+**		**two-pass**		**two-pass+**	
[%]	**BLEU**	**CI 95%**	**BP**	**BLEU**	**BP**	**BLEU**	**BP**	**BLEU**	**BP**	**BLEU**	**BP**	**BLEU**	**BP**
5	27.82	(±0.65)	1.00	29.51	1.00	29.00	0.97	30.25	0.99	29.58	0.94	29.93	0.96
10	29.70	(±0.69)	1.00	31.42	1.00	30.74	0.98	31.99	0.99	31.30	0.95	31.75	0.97
20	31.37	(±0.69)	1.00	32.56	1.00	32.64	1.00	33.17	0.99	32.60	0.96	32.76	0.98
40	32.66	(±0.66)	1.00	33.52	1.00	33.23	0.99	33.98	1.00	33.65	0.97	33.88	0.99
80	33.67	(±0.66)	1.00	**34.17**	1.00	33.93	0.99	**34.38**	1.00	**34.20**	0.99	**34.35**	1.00
100	**33.90**	(±0.67)	1.00	34.03	1.00	**33.98**	1.00	34.02	1.00	33.90	1.00	34.08	1.00

vidual translation system. If the correlation is too low ($< 5\%$), there might be no consensus among the input systems and the quality of the consensus translations will hardly differ from a random selection of the candidates.

To study how correlation affects performance in system combination, we built a large number of systems trained on randomly sampled portions of the FBIS [4] training data collection. Sample sizes ranged between 5% and 100% with each larger data set doubling the size of the next smaller collection. For each sample size, we created 10 data sets, thus resulting in a total of 6×10 training corpora. On each data set, a new translation system was trained from scratch and used for decoding the ZHEN-05 test sentences. All 60 systems applied the MBR decision rule (Kumar and Byrne, 2004), which gave an additional 0.5% gain on average on top of using the *maximum a-posteriori* (MAP) decision rule. Systems trained on equally amounts of training data were incrementally combined. Figure 1 shows the evolution of the BLEU scores as a function of the number of systems as the sample size is increased from 5–100%. Table 4 shows the BLEU scores obtained with a 10-way system combination using the MBR-like candidate selection rule, word sausages, and the two-pass search algorithm together with their improved versions "sausages+" and "two-pass+", respectively. In order to measure the correlation between the individual translation systems, we computed the inter-system BLEU score matrix as shown exemplary

[4] LDC catalog number: LDC2003E14

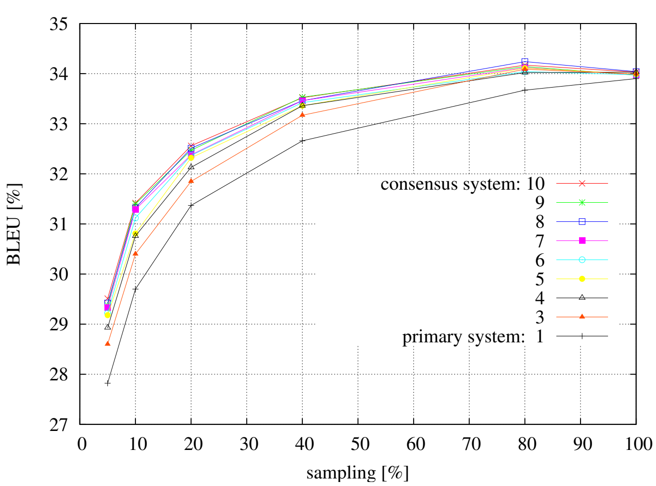

Figure 1: *Incremental system combination on ZHEN-05 using the MBR-like candidate selection rule and uniform prior weights. Systems were trained with different sample sizes of the FBIS data.*

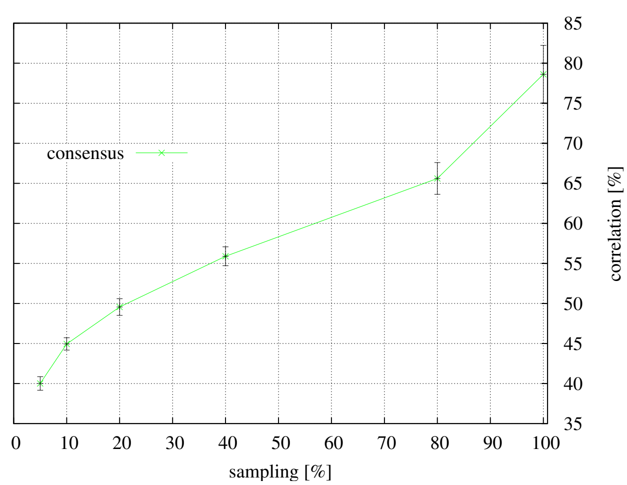

Figure 2: *Evolution of the correlation on ZHEN-05 averaged over 10 systems in the course of the sample size.*

Table 5: *Minimum, maximum, and average inter-system BLEU score correlations for (i) the primary systems of the 2006 NIST machine translation evaluation on the ZHEN-06 test data, (ii) different training corpus sizes (FBIS), and (iii) a greedy strategy which chooses 15 systems out of a pool of 200 translation systems.*

	ZHEN-6	**ZHEN-5**						**ZHEN-5**	**ZHEN-6**
	16 primary	FBIS sampling, 10 systems						15 systems	15 systems
	systems	5%	10%	20%	40%	80%	100%	greedy selection	ZHEN-5 selection
min	0.08	0.38	0.44	0.47	0.53	0.60	0.72	0.55	0.50
mean	0.18	0.40	0.45	0.50	0.56	0.66	0.79	0.65	0.61
median	0.19	0.40	0.45	0.49	0.56	0.64	0.78	0.63	0.58
max	0.28	0.42	0.47	0.53	0.58	0.70	0.88	0.85	0.83

in Table 6 for the 16 MT06 primary submissions. Figure 2 shows the evolution of the correlation averaged over 10 systems as the sample size is increased from 5–100%. Note that all systems were optimized using a non-deterministic implementation of the *Minimum Error Rate Training* described in (Och, 2003). Hence, using all of the FBIS corpus data does not necessarily result in fully correlated systems, since the training procedure may pick a different solution for same training data in order to increase diversity. Both Table 4 and Figure 1 clearly indicate that increasing the correlation (and thus reducing the diversity) substantially reduces the potential of a consensus system to outperform the primary translation system. Ideally, the correlation should not be larger than 30%.

Especially for low inter-system correlations and reduced translation quality, both the enhanced versions of the word sausage combination method and the two-pass search outperform the MBR-like candidate selection scheme. This advantage, however, diminishes as soon as the correlation increases and translations produced by the individual systems become more similar.

3.4 Toward Automatic System Generation and Selection

Sampling the training data is an effective means to investigate the effect of system correlation on consensus performance. However, this is done at the expense of the overall system quality. What we need instead is a method to reduce correlation without sacrificing system performance.

A simple, though computationally very expensive way to build an ensemble of low-correlated statistical machine translation systems from a single translation engine is to train a large pool of systems, in which each of the systems is trained with a slightly different set of parameters. Changing only few parameters at a time typically results in only small changes in system performance but may have a strong impact on system correlation. In our experiments we observed that changing parameters which affect the training procedure at a very early stage, are most effective and introduce larger diversity. For instance, changing the training procedure for word alignment models turned out to be most beneficial; for details see (Och and Ney, 2003). Other parameters that were changed include the maximum jump width in word re-ordering, the choice of feature function weights for the log-linear translation models, and the set of language models used in decoding.

Once a large pool of translation systems has been generated, we need a method to select a small ensemble of diverse translation outputs that are beneficial for computing consensus translations. Here, we used a greedy strategy to rank the systems with respect to their ability to improve system

Table 6: *Inter-system BLEU score matrix for primary systems of the NIST 2006 TIDES machine translation evaluation on the ZHEN-06 test data.*

Id	**01**	**02**	**03**	**04**	**05**	⋯	**14**	**15**	**16**
01	1.00	0.27	0.26	0.23	0.26	⋯	0.15	0.15	0.12
02	0.27	1.00	0.27	0.22	0.25	⋯	0.15	0.15	0.12
03	0.26	0.27	1.00	0.21	0.28	⋯	0.15	0.15	0.10
04	0.23	0.22	0.21	1.00	0.19	⋯	0.14	0.12	0.12
05	0.26	0.25	0.28	0.19	1.00	⋯	0.16	0.17	0.11
06	0.27	0.24	0.25	0.21	0.26	⋯	0.16	0.18	0.13
⋮						⋱			⋮
14	0.15	0.15	0.15	0.14	0.16	⋯	1.00	0.12	0.08
15	0.15	0.15	0.15	0.12	0.17	⋯	0.12	1.00	0.09
16	0.12	0.12	0.10	0.12	0.11	⋯	0.08	0.09	1.00

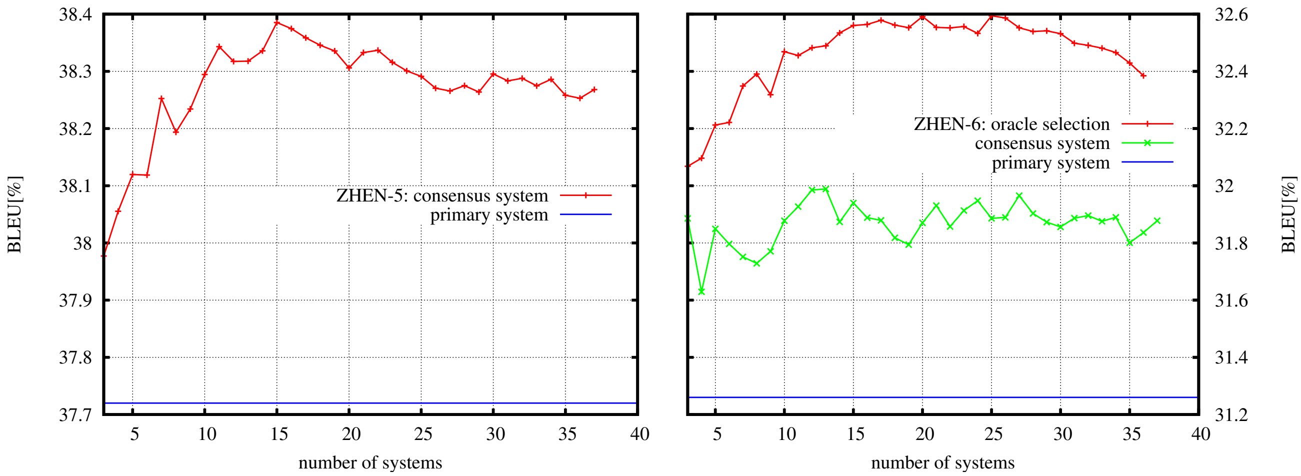

Figure 3: *BLEU score of the consensus translation as a function of the number of systems on the ZHEN-05 sentences (left) and ZHEN-06 sentences (right). The middle curve (right) shows the variation of the BLEU score on the ZHEN-06 data when the greedy selection of the ZHEN-05 is used.*

combination. Initially, the greedy strategy selected the best individual system and then continued by adding those systems to the ensemble, which gave the highest gain in terms of BLEU score according to the MBR-like system combination method. Note that the greedy strategy is not guaranteed to increase the BLEU score of the combined system when a new system is added to the ensemble of translation systems.

In a first experiment, we trained approximately 200 systems using different parameter settings in training. Each system was then used to decode both the ZHEN-05 and the ZHEN-06 test sentences using the MBR decision rule. The upper curve in Figure 3 (left) shows the evolution of the BLEU score on the ZHEN-05 sentences in the course of the number of selected systems. The upper curve in Figure 3 (right) shows the BLEU score of the consensus translation as a function of the number of systems when the selection is done on the ZHEN-06 set. This serves as an oracle. The middle curve (right) shows the function of the BLEU score when the system selection made on the ZHEN-05 set is used in order to combine the translation outputs for the ZHEN-06 data. Although system combination gave moderate improvements on top of the primary system, the greedy strategy still needs further refinements in order to improve generalization. While the correlation statistics shown in Table 5 indicate that changing the training parameters helps to substantially decrease system correlation, there is still need for additional methods in order to reduce the level of inter-system BLEU scores such that they fall within the range of [0.2, 0.3].

4 Conclusions

In this paper, we presented an empirical study on how different selections of translation outputs affect translation quality in system combination. Composite translations were computed using (i) a candidate selection method based on inter-system BLEU score matrices, (ii) an enhanced version of word sausage networks, and (iii) a novel two-pass search algorithm which determines and re-orders bags of words that build the constituents of the final consensus hypothesis. All methods gave statistically significant improvements.

We showed that both a high diversity among the original translation systems and a similar translation quality among the translation systems are essential in order to gain substantial improvements on top of the best individual translation systems.

Experiments were conducted on the NIST portion of the Chinese English text translation corpus used for the 2006 NIST machine translation evaluation. Combined systems were built from primary systems of up to 16 different research labs as well as systems derived from one of the best-ranked translation engines.

We trained a large pool of translation systems from a single translation engine and presented first experimental results for a greedy search to select an ensemble of translation systems for system combination.

References

S. Banerjee and A. Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization, 43th Annual Meeting of the Association of Computational Linguistics (ACL-2005)*, Ann Arbor, MI, USA, June.

S. Bangalore, G. Bodel, and G. Riccardi. 2001. Computing Consensus Translation from Multiple Machine Translation Systems. In *2001 Automatic Speech Recognition and Understanding (ASRU) Workshop*, Madonna di Campiglio, Trento, Italy, December.

T. Brants, A. Popat, P. Xu, F. Och, and J. Dean. 2007. Large Language Models in Machine Tranlation. In *Proceedings of the 2007 Conference on Empirical Methods in Natural Language Processing*, Prague, Czech Republic. Association for Computational Linguistics.

J. G. Fiscus. 1997. A Post-Processing System to Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduction (ROVER). In *Proceedings 1997 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 347–352, Santa Barbara, CA, USA, December.

S. Jayaraman and A. Lavie. 2005. Multi-Engine Machine Translation Guided by Explicit Word Matching. In *10th Conference of the European Association for Machine Translation (EAMT)*, pages 143–152, Budapest, Hungary.

P. Koehn, F. J. Och, and D. Marcu. 2003. Statistical Phrase-Based Translation. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 48–54, Edmonton, Canada. Association for Computational Linguistics.

P. Koehn. 2004. Statistical Significance Tests for Machine Translation Evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain, August. Association for Computational Linguistics.

S. Kumar and W. Byrne. 2004. Minimum Bayes-Risk Decoding for Statistical Machine Translation. In *Proc. HLT-NAACL*, pages 196–176, Boston, MA, USA, May.

E. Matusov, N. Ueffing, and H. Ney. 2006. Computing Consensus Translation from Multiple Machine Translation Systems Using Enhanced Hypotheses Alignment. In *11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 33–40, Trento, Italy, April.

T. Nomoto. 2004. Multi-Engine Machine Translation with Voted Language Model. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, pages 494–501, Barcelona, Spain, July.

E. W. Noreen. 1989. *Computer-Intensive Methods for Testing Hypotheses*. John Wiley & Sons, Canada.

F. J. Och and H. Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51.

F. J. Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *41st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 160–167, Sapporo, Japan, July.

K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2001. Bleu: a Method for Automatic Evaluation of Machine Translation. Technical Report RC22176 (W0109-022), IBM Research Division, Thomas J. Watson Research Center, Yorktown Heights, NY, USA.

M. Paul, T. Doi, Y. Hwang, K. Imamura, H. Okuma, and E. Sumita. 2005. Nobody is Perfect: ATR's Hybrid Approach to Spoken Language Translation. In *International Workshop on Spoken Language Translation*, pages 55–62, Pittsburgh, PA, USA, October.

F. Robert and S. Nirenburg. 1994. Three Heads are Better than One. In *Proceedings of the Fourth ACL Conference on Applied Natural Language Processing*, Stuttgart, Germany, October.

K. C. Sim, W. Byrne, M. Gales, H. Sahbi, and P.C. Woodland. 2007. Consensus network decoding for statistical machine translation system combination. In *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Honolulu, HI, USA, April.

M. Snover, B. J. Dorr, R. Schwartz, J. Makhoul, L. Micciulla, and R. Weischedel. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of Association for Machine Translation in the Americas*.