Advances in Czech – Signed Speech Translation*

Jakub Kanis and Luděk Müller

Univ. of West Bohemia, Faculty of Applied Sciences, Dept. of Cybernetics
Univerzitní 8, 306 14 Pilsen, Czech Republic
{jkanis,muller}@kky.zcu.cz

Abstract. This article describes advances in Czech – Signed Speech translation. A method using a new criterion based on minimal loss principle for log-linear model phrase extraction was introduced and it was evaluated against two another criteria. The performance of phrase table extracted with introduced method was compared with performance of two another phrase tables (manually and automatically extracted). A new criterion for semantic agreement evaluation of translations was introduced too.

Key words: machine translation; signed speech; phrase extraction

1 Introduction

In the scope of this paper, we are using the term Signed Speech (SS) for both the Czech Sign Language (CSE) and Signed Czech (SC). The CSE is a natural and adequate communication form and a primary communication tool of the hearing-impaired people in the Czech Republic. It is composed of the specific visual-spatial resources, i.e. hand shapes (manual signals), movements, facial expressions, head and upper part of the body positions (non-manual signals). It is not derived from or based on any spoken language. On the other hand the SC was introduced as an artificial language system derived from the spoken Czech language to facilitate communication between deaf and hearing people. SC uses grammatical and lexical resources of the Czech language. During the SC production, the Czech sentence is audibly or inaudibly articulated and simultaneously the CSE signs of all individual words of the sentence are signed.

2 Phrase-Based Machine Translation

The goal of the machine translation is to find the best translation $\hat{\mathbf{t}} = w_1, ..., w_I$ of the given source sentence $\mathbf{s} = w_1, ..., w_J$. The state of the art solution of this problem is using log-linear model [1]:

$$Pr(\mathbf{t}|\mathbf{s}) = p_{\lambda_1^M}(\mathbf{t}|\mathbf{s}) = \frac{\exp(\sum_{m=1}^{M} \lambda_m h_m(\mathbf{t}, \mathbf{s}))}{\sum_{\mathbf{t}'} \exp(\sum_{m=1}^{M} \lambda_m h_m(\mathbf{t}', \mathbf{s}))} \quad (1)$$

* This research was supported by the Grant Agency of Academy of Sciences of the Czech Republic, project No. 1ET101470416 and by the Ministry of Education of the Czech Republic, project No. MŠMT LC536.

There are feature models $h_m(\mathbf{t}, \mathbf{s})$, which model a relationship between the source and the target language and its weights λ_m. If we want to have the best translation we should choose the one with the highest probability, thus:

$$\hat{\mathbf{t}} = \underset{\mathbf{t}}{\operatorname{argmax}} \left\{ \frac{\exp(\sum_{m=1}^{M} \lambda_m h_m(\mathbf{t}, \mathbf{s}))}{\sum_{\mathbf{t}'} \exp(\sum_{m=1}^{M} \lambda_m h_m(\mathbf{t}', \mathbf{s}))} \right\} = \underset{\mathbf{t}}{\operatorname{argmax}} \left\{ \sum_{m=1}^{M} \lambda_m h_m(\mathbf{t}, \mathbf{s}) \right\}, \tag{2}$$

where we have disregarded the denominator of the Equation 2. In the log-linear model we can use a portion of different feature models. The source sentence $\mathbf{s}$ is segmented into a sequence of K phrases $\bar{s}_1, ..., \bar{s}_K$ which we call phrase alignment (all possible segmentations have the same probability) in the case of phrase-based translation. We define the phrase of a given **length l** as a continual word sequence: $\bar{s}_i = w_j, ..., w_{j+l}, j = 1, ..., J - l$. Each source phrase $\bar{s}_i, i = 1, ..., K$ is translated into a target phrase $\bar{t}_i$ in the decoding process. This particular ith translation is modeled by a probability distribution $\phi(\bar{s}_i|\bar{t}_i)$. The target phrases can be reordered to get more precise translation. The reordering of the target phrases can be modeled by a relative distortion probability distribution $d(a_i - b_{i-1})$ as in [3], where a_i denotes the start position of the source phrase which was translated into the ith target phrase, and b_{i-1} denotes the end position of the source phrase translated into the $(i-1)th$ target phrase. The basic feature models are: the both direction translation models ϕ, distortion model d, n-gram based language model p_{LM} and phrase p_{PhP} and word p_{WP} penalty models. The mostly used method for the weight adjustment is minimum error rate training (MERT) [2], where the weights are adjusted to minimize the error rate of the resulting translation:

$$\hat{\lambda}_1^M = \underset{\lambda_1^M}{\operatorname{argmin}} \left\{ \sum_{n=1}^{N} \sum_{k=1}^{K} E(r_n, \mathbf{t}_{n,k}) \delta(\hat{\mathbf{t}}(s_n, \lambda_1^M), \mathbf{t}_{n,k}) \right\} \tag{3}$$

$$\hat{\mathbf{t}}(\mathbf{s}_n, \lambda_1^M) = \underset{\mathbf{t} \in C_n}{\operatorname{argmax}} \left\{ \sum_{m=1}^{M} \lambda_m h_m(\mathbf{t}, \mathbf{s}_n) \right\} \tag{4}$$

$$\delta(\hat{\mathbf{t}}(\mathbf{s}_n, \lambda_1^M), \mathbf{t}_{n,k}) = \begin{cases} 1 \; if \quad \hat{\mathbf{t}}(\mathbf{s}_n, \lambda_1^M) = \mathbf{t}_{n,k} \\ 0 \; else \end{cases},$$

where N is number of sentence pairs in a training corpus, E error criterion which is minimized, r_n is reference translation of the source sentence $\mathbf{s}_n$ and $C_n = \{\mathbf{t}_{n,1}, ..., \mathbf{t}_{n,K}\}$ is a set of K different translations $\mathbf{t}_n$ of each source sentence $\mathbf{s}_n$.

3 Phrase Extraction Based on Minimal Loss Principle

The main source of the SMT system is a phrase table with bilingual pairs of phrases. State of the art methods for the phrase extraction are based on alignment modeling (especially on the word alignment modeling). The word alignment

can be modeled by probabilistic models of different complexity (Models 1 – 6 [7]). The model complexity directly influences the alignment error rate and thus the translation accuracy: the more complexity model, the better translations. However, more complicated models are computationally challenging. For example, the task of finding the Viterbi alignment for the Models 3 – 6 is an NP-complete problem [7]. Only a suboptimal solution can be found with usage of approximations. In addition, it was founded that the next reduction of word alignment errors does not have to lead to better translations [8]. Because of problems with word alignment models we have proposed using of the log-linear model for the phrase extraction, which can be optimized directly to the translation precision. Our solution is similar to the one in work [9] with some differences. Firstly, we are using different set of features without using of any alignment modeling. Secondly, we introduce a new criterion for phrase extraction based on a minimal loss principle.

Method Description Our task is to find for each source phrase $\bar{s}$ its translation, i.e. the corresponding target phrase $\bar{t}$. We suppose that we have a sentence aligned bilingual corpus (pairs of the source and target sentences). We start with the source sentence $\mathbf{s} = w_1, ..., w_J$ and the target sentence $\mathbf{t} = w_1, ..., w_I$ and generate a bag β of all possible phrases up to the given length l: $\beta\{\mathbf{s}\} = \{\bar{s_m}\}_{m=1}^{l}, \{\bar{s_m}\} = \{w_n, ..., w_{n+m-1}\}_{n=1}^{J-m+1}$, $\beta\{\mathbf{t}\} = \{\bar{t_m}\}_{m=1}^{l}, \{\bar{t_m}\} = \{w_n, ..., w_{n+m-1}\}_{n=1}^{I-m+1}$. The source phrases longer than one word are keeping for next processing only if they have been seen in the corpus at least as much as given threshold τ (reasonable threshold is five). All target phrases are keeping regardless of the number of their occurrence in the corpus. Each target phrase is considered to be a possible translation of each kept source phrase $\forall \bar{s} \in \beta\{\mathbf{s}\} : N(\bar{s}) \geq \tau : \bar{s} \rightarrow \beta\{\mathbf{t}\}$, where $N(\bar{s})$ is number of occurrences of phrase $\bar{s}$ in the corpus. Now for each possible translation pair $(\bar{s}, \bar{t}) : \bar{t} \in T(\bar{s}), T(\bar{s}) = \{\bar{t}\} : \bar{s} \rightarrow \tilde{t}$ we compute its corresponding score:

$$c(\bar{s}, \bar{t}) = \sum_{k=1}^{K} \lambda_k h_k(\bar{s}, \bar{t}), \tag{5}$$

where $h_k(\bar{s}, \bar{t}), k = 1, 2, ..., K$ is set of K features, which describe the relationship between the pair of phrases $(\bar{s}, \bar{t})$. The MERT training can be used for weights λ_k optimization. The resulting scores $\mathbf{c} = \{c\}$ are stored in a hash table, where the source phrase $\bar{s}$ is the key and all possible translations $\bar{t} \in T(\bar{s})$ with its score $c(\bar{s}, \bar{t})$ are the data. We process the whole training corpus and store the scores for all possible translation pairs.

The next step is choosing only "good" translations $\bar{t_G}$ from all possible translations $T(\bar{s})$ for each source phrase $\bar{s}$, i.e. we get a set of translations $T_G(\bar{s}) = \{\bar{t_G}\} : \bar{s} \rightarrow \bar{t_G}$. For each sentence pair we generate the bag of all phrases up to the given length l for both sentences. Then for each $\bar{s} \in \beta(\mathbf{s})$ we compute a **translation loss** $\mathbf{L_T}$ for each $\bar{t} \in T(\bar{s}) = \beta(\mathbf{t})$. The translation loss

L_T for the source phrase $\bar{s}$ and its possible translation $\bar{t}$ is defined as:

$$L_T(\bar{s},\bar{t}) = \frac{\sum_{\tilde{s}_i \in \beta(\mathbf{s}), \tilde{s}_i \neq \bar{s}} c(\tilde{s}_i,\bar{t})}{c(\bar{s},\bar{t})} \tag{6}$$

We compute how much probability mass we lost for the rest of source phrases from the bag $\beta(\mathbf{s})$ if we translate $\bar{s}$ as $\bar{t}$. For each $\bar{s}$ we store all translation losses $L_T(\bar{s},\bar{t})$ for all $\bar{t} \in \beta(\mathbf{t})$. The "good" translation $\bar{t_G}$ for $\bar{s}$ is the one (or more) with the lowest translation loss $L_T(\bar{s},\bar{t})$:

$$\bar{t_G} = \underset{\bar{t}}{\operatorname{argmin}}\, L_T(\bar{s},\bar{t}) \tag{7}$$

and all the other translations are discarded. We process all sentence pairs and get a new phrase table. This table comprises source phrases $\bar{s}$, corresponding "good" translations $\bar{t_G} \in T_G(\bar{s})$ only, and the numbers of how many times a particular translation $\bar{t}$ was determined as a "good" translation $\bar{t_G}$. These information can be then used for example for calculation of translation probabilities ϕ.

Used Features We used only features based on number of occurrences of translation pairs and particular phrases in the training corpus. We collect these numbers: number of occurrences of each considered source phrase $N(\bar{s})$, number of occurrences of each target phrase $N(\bar{t})$, number of occurrences of each possible translation pair $N(\bar{s},\bar{t})$ and number of how many times was given source or target phrase considered as translation $N_T(\bar{s})$ and $N_T(\bar{t})$ (it corresponds to the number of all phrases for which was given phrase considered as their possible translation in all sentence pairs). These numbers are used to compute the following features: translation probability ϕ, probability p_T that given phrase is a translation - all for both translation directions and translation probability p_{MI} based on mutual information. The translation probability ϕ is defined on base of relative frequencies as [3]:

$$\phi(\bar{s}|\bar{t}) = \frac{N(\bar{s},\bar{t})}{N(\bar{t})} \quad \phi(\bar{t}|\bar{s}) = \frac{N(\bar{s},\bar{t})}{N(\bar{s})} \tag{8}$$

Probability p_T, that given phrase is a translation, i.e. it appears together with considered phrase as its translation, is defined as:

$$p_T(\bar{s}|\bar{t}) = \frac{N(\bar{s},\bar{t})}{N_T(\bar{t})} \quad p_T(\bar{t}|\bar{s}) = \frac{N(\bar{s},\bar{t})}{N_T(\bar{s})} \tag{9}$$

Translation probability p_{MI} based on mutual information is defined as [10] (we can use both numbers N and N_T for computing):

$$p_{MI}(\bar{s},\bar{t}) = \frac{MI(\bar{s},\bar{t})}{\sum_{\bar{t}\in T(\bar{s})} MI(\bar{s},\bar{t})} \quad p_{MI_T}(\bar{s},\bar{t}) = \frac{MI_T(\bar{s},\bar{t})}{\sum_{\bar{t}\in T(\bar{s})} MI_T(\bar{s},\bar{t})} \tag{10}$$

$$MI(\bar{s},\bar{t}) = p(\bar{s},\bar{t}) \log \frac{p(\bar{s},\bar{t})}{p(\bar{s})\cdot p(\bar{t})} \quad MI_T(\bar{s},\bar{t}) = p_T(\bar{s},\bar{t}) \log \frac{p_T(\bar{s},\bar{t})}{p_T(\bar{s})\cdot p_T(\bar{t})} \tag{11}$$

$$p(\bar{s},\bar{t}) = \frac{N(\bar{s},\bar{t})}{N_S} \quad p_T(\bar{s},\bar{t}) = \frac{N(\bar{s},\bar{t})}{N_T} \tag{12}$$

$$p(\bar{s}) = \frac{N(\bar{s})}{N_S}\ p(\bar{t}) = \frac{N(\bar{t})}{N_S} \quad p_T(\bar{s}) = \frac{N_T(\bar{s})}{N_T}\ p_T(\bar{t}) = \frac{N_T(\bar{t})}{N_T}, \tag{13}$$

where N_S is the number of all sentence pairs in the corpus and N_T is the number of all possible considered translations, i.e. if source sentence length is five and target sentence length nine then we add 45 to N_T. Finally we have six features: $\phi(\bar{s}|\bar{t})$, $\phi(\bar{t}|\bar{s})$, $p_T(\bar{s}|\bar{t})$, $p_T(\bar{t}|\bar{s})$, $p_{MI}(\bar{s},\bar{t})$ and $p_{MI_T}(\bar{s},\bar{t})$ for the phrase extraction.

4 Tools and Evaluation Methodology

Data The main resource for the statistical machine translation is a parallel corpus which contains parallel texts of both the source and the target language. Acquisition of such corpus in case of SS is complicated by the absence of the official written form of both the CSE and the SC. Therefore we have used the Czech to Signed Czech (CSC) parallel corpus [4] for all experiments. For the purpose of experiments we have split the CSC corpus into training, development and testing part, which are described in Table 1 in more details.

Evaluation Criteria We have used the following well known criteria for evaluation of our experiments. The first criterion is the **BLEU** score: it counts modified n-gram precision for output translation with respect to the reference translation. The second criterion is the **NIST** score: it counts similarly as BLEU modified n-gram precision, but uses arithmetic mean and weighing by information gain of each n-gram. Next criterion is **Sentence Error Rate** (**SER**): it is a ratio of the number of incorrect sentence translations to the number of all translated sentences. The **Word Error Rate** (**WER**) criterion is adopted from ASR area: is defined as the Levensthein edit distance between the produced translation and the reference translation in percentage (a ratio of the number of all deleted, substituted and inserted produced words to the total number of reference words). The third error criterion is **Position-independent Word Error Rate** (**PER**): it compares two sentences without regard to their word order. These criteria however evaluate only lexical agreement between the reference and the resulting translation. But in the automatic translation we need to find out if two different word constructions have the same meaning, i.e. are semantically identical, because there are always equally correct different translations of each source sentence (for example there are mostly more reference

Table 1. Dividing of the CSC corpus into training, development and testing part.

	Training data		Development data		Testing data	
	CZ	SC	CZ	SC	CZ	SC
Sent. pairs	12 616		1 578		1 578	
# words	86 690	86 389	10 700	10 722	10 563	10 552
Vocab. size	3 670	2 151	1 258	800	1 177	748
# singletons	1 790	1 036	679	373	615	339
OOV(%)	–	–	240 (2.24)	122 (1.14)	208 (1.97)	105 (1.00)

translations of each source sentence in the corpus). We have proposed a new **Semantic Dimension Overlap** (**SDO**) criterion to evaluate semantic similarity of the translations between Czech and SC. The SDO criterion is based on the overlap between semantic annotation of the reference translation and semantic annotation of the resulting translation. The semantic annotation is created by HVS (Hidden Vector State) parser [5], which is trained on the CSC corpus data (the CSC corpus contains semantic annotation layer needed for the HVS parser training). A lower values of the three error criteria: SER, WER, PER and a higher values of the three precision criteria: BLEU, NIST, SDO indicates better, i.e. more precise translation.

Decoders Two different phrase-based decoders were used in our experiments. The first decoder is freely available state-of-the-art factored phrase-based beam-search decoder - **MOSES**[1] [6], which uses log-linear model (MERT training). The training tools for extraction of phrases from the parallel corpus are also available, i.e. the whole translation system can be constructed given a parallel corpus only. For the language modeling was used the SRILM[2] toolkit.

The second decoder is our implementation of monotone phrase-based decoder - **SiMPaD**, which already uses log-linear model (MERT training). The monotonicity means using the monotone reordering model only, i.e. no phrase reordering is permitted during the search. SiMPaD uses SRILM[2] language models and the Viterbi algorithm for the decoding, which defines generally n-gram dependency between translated phrases.

5 Experiments and Conclusion

Phrase Extraction Based on Minimal Loss Principle In the first experiment we compared the new criterion based on minimal loss principle (ML) proposed in Section 3 with two another criteria for the phrase extraction. All six features defined in Section 3 was used in log-linear model. The first one

[1] http://www.statmt.org/moses/
[2] http://www.speech.sri.com/projects/srilm/download.html

(BestG) is criterion used in the work [9] which selects all translation pairs for each sentence pair with score c higher than maximal score $c_m - \ threshold\ \tau$. The second one (BestL) criterion is criterion which selects only the translation pair with the highest score c_m for each source phrase in the sentence pair. The results are in Figure 1, where N means a number of first N best scores c selected for each source phrase. The ML criterion performs best (75.09), the second is BestL criterion (72.83) and the last is BestG criterion (72.31).

Phrase Table and Decoders Comparison In this experiment we have compared the translation accuracy of handcrafted (HPH) and automatically extracted phrases (phrases extracted by Moses (MPH) and phrases extracted by the method described in Section 3 (MLPH)). In the case of the MLPH table extraction we used additional techniques as a intersection of phrase tables for both translation directions and a subsequent filtration of the resulting table trough the training data translation. We compared both decoders too (M for MOSES, S for SiMPaD). The results in Table 2 are reported for testing data after MERT optimization on the BLEU criterion. The bootstrap method was used for acquisition of reliable results and confidence intervals (lower and upper indexes).

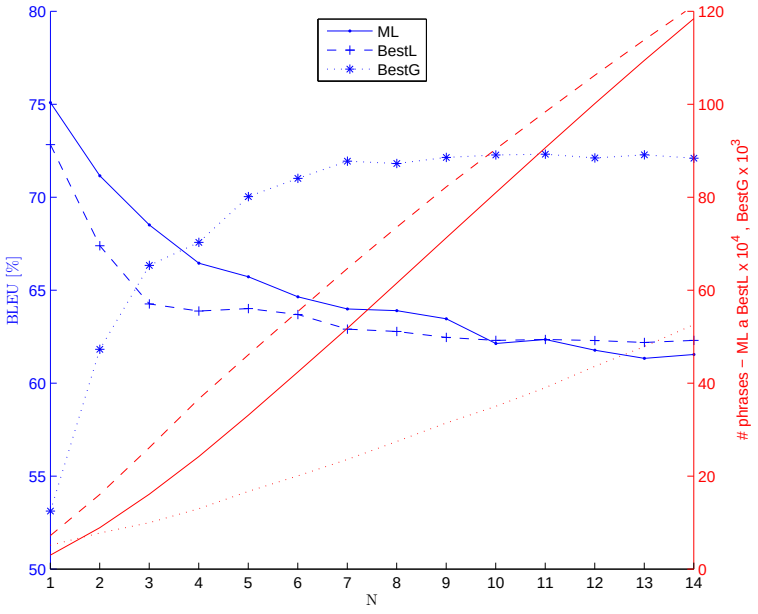

Fig. 1. Comparison of different criteria for the phrase extraction.

Table 2. Comparison of different phrase tables and decoders.

	HPH		MPH		MLPH	
Size	5 325		65 494		11 585	
	M	S	M	S	M	S
Bleu[%]	$\mathbf{81.29}^{1.27}_{1.29}$	$81.22^{1.31}_{1.31}$	$80.87^{1.31}_{1.31}$	$81.08^{1.27}_{1.32}$	$80.20^{1.28}_{1.33}$	$80.21^{1.32}_{1.36}$
NIST	$\mathbf{11.65}^{0.13}_{0.14}$	$\mathbf{11.65}^{0.13}_{0.13}$	$11.57^{0.13}_{0.14}$	$11.58^{0.14}_{0.14}$	$11.47^{0.14}_{0.14}$	$11.44^{0.14}_{0.14}$
SER[%]	$\mathbf{38.15}^{3.49}_{3.30}$	$38.53^{3.42}_{3.30}$	$38.21^{3.42}_{3.42}$	$38.59^{3.49}_{3.36}$	$40.56^{3.49}_{3.36}$	$42.90^{3.55}_{3.36}$
WER[%]	$13.14^{1.33}_{1.29}$	$\mathbf{13.06}^{1.32}_{1.25}$	$13.43^{1.36}_{1.31}$	$13.43^{1.31}_{1.25}$	$14.48^{1.41}_{1.35}$	$14.88^{1.42}_{1.33}$
PER[%]	$\mathbf{11.64}^{1.22}_{1.17}$	$11.72^{1.20}_{1.13}$	$11.85^{1.21}_{1.16}$	$11.93^{1.20}_{1.13}$	$12.95^{1.21}_{1.18}$	$13.24^{1.26}_{1.16}$
SDO[%]	$92.08^{1.95}_{2.37}$	$\mathbf{92.25}^{1.96}_{2.30}$	$92.12^{2.03}_{2.30}$	$92.11^{2.01}_{2.39}$	$90.84^{2.07}_{2.49}$	$90.82^{2.13}_{2.51}$

The results show that HPH and MPH tables perform equal while the MLPH table is about one to two percent depending on the criterion behind them. The main advantage of the HPH and MLPH tables is their smaller size in confrontation with the MPH table size. The HPH table is about twelve times and the MLPH table about five times smaller than the MPH table. The difference between results of both decoders is negligible too except the result for the SER criterion and the MLPH table. An explanation of this difference can be a good theme for a future examination.

References

1. F.J. Och, H. Ney. Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. In Proc. 40th Annual Meeting of the ACL, pages 295–302, Philadelphia, PA, July 2002.
2. F.J. Och. Minimum Error Rate Training in Statistical Machine Translation. In Proc. 41st Annual Meeting of the ACL, Sapporo, Japan, July 2003.
3. Koehn, P. et al., Statistical Phrase-Based Translation, HLT/NAACL, 2003.
4. Kanis, J. et al., Czech-Sign Speech Corpus for Semantic Based Machine Translation, In Lecture Notes in Artificial Intelligence, v.4188, pp.613-620, 2006.
5. F. Jurčíček el al., Extension of HVS semantic parser by allowing left-right branching. In International Conference on Acoustics, Speech, and Signal Processing, Las Vegas, USA, 2008.
6. Koehn, P. et al., Moses: Open Source Toolkit for Statistical Machine Translation. Annual Meeting of the ACL, Prague, Czech Republic, June 2007.
7. Och, F., J., Ney, H., A Systematic Comparison of Various Statistical Alignment Models, Computational Linguistics, volume 29, number 1, pp. 19-51 March 2003.
8. R. Zens. Phrase-based Statistical Machine Translation: Models, Search, Training. PhD thesis, RWTH Aachen University, Aachen, Germany, February 2008.
9. Y. Deng et al., Phrase Table Training for Precision and Recall: What Makes a Good Phrase and a Good Phrase Pair? In Proceedings of ACL-08: HLT, pages 81–88, Columbus, Ohio, June 2008.
10. C. Lavecchia el al., Phrase-Based Machine Translation based on Simulated Annealing. In Proceedings of the Sixth International Conference on Language Resources and Evaluation, Marrakech, Morocco, 2008. ELRA.