A Maximum Entropy Approach to Syntactic Translation Rule Filtering

Marcin Junczys-Dowmunt

Adam Mickiewicz University
Faculty of Mathematics and Computer Science
ul. Umultowska 87, 61-614 Poznań, Poland
junczys@amu.edu.pl

Abstract. In this paper we will present a maximum entropy filter for the translation rules of a statistical machine translation system based on tree transducers. This filter can be successfully used to reduce the number of translation rules by more than 70% without negatively affecting translation quality as measured by BLEU. For some filter configurations, translation quality is even improved.

Our investigations include a discussion of the relationship of *Alignment Error Rate* and *Consistent Translation Rule Score* with translation quality in the context of Syntactic Statistical Machine Translation.

1 Introduction

A crucial step when preparing a Syntactic Statistical Machine Translation system involves extracting a large set of translation rules from a bilingual word-aligned corpus. Even small errors in the alignment data may lead to the extraction of many wrong rules that can seriously affect translation quality. The majority of approaches designed to prevent "rogue rules" relies on methods that improve word alignments so they become more consistent with the given syntactic data, examples being [1,2]. As a result, the number of translation rules usually increases, but many of these rules are still incorrect or unlikely to be used in any translation. On the other hand, a reduction in the number of rules (e.g. by frequency thresholds or phrase length limitations) might cause a decrease in translation quality.[1] However, adhering to many possibly redundant translation rules results in greater requirements concerning resources and processing time.

Instead of tuning a single word alignment towards generating better rules, we extracted translation rules from several word alignments which have been created with different tools and combination methods. These rules were scored and discarded if they failed to achieve a predetermined threshold. This score is the probability that a rule represented by a set of features belongs to a class of correct rules as calculated by a Maximum Entropy (ME) model. This ME model learns to distinguish between correct and incorrect rules by being trained on a set

[1] This has been shown by [3] in the context of Phrase-Based, Hierarchical Phrased-Based and Syntax-Augmented SMT.

A. Gelbukh (Ed.): CICLing 2010, LNCS 6008, pp. 451–463, 2010.

of reference rules extracted from manually word-aligned sentences. Our decision to use many input word alignments instead of a chosen single word alignment is motivated by the increased coverage of correct rules that can be achieved this way. We show that it is possible to reduce the number of translation rules with this simple supervised machine-learning approach by 60–70% without sacrificing translation quality as measured by BLEU and NIST. Actually, for some filter settings, the translation quality is even higher than for the unfiltered rule sets.

Part of our investigations comprises a short discussion of the relationship of *Alignment Error Rate* (AER) and *Consistent Translation Rule Score* (CTRS) — a metric equivalent to *Consistent Phrase Error Rate* (CPER) [4] adapted to translation rules — with translation quality in the context of Syntactic SMT. Similar questions have been addressed by [4] in the context of Phrase-Based SMT, but only marginally for Syntactic SMT [1].

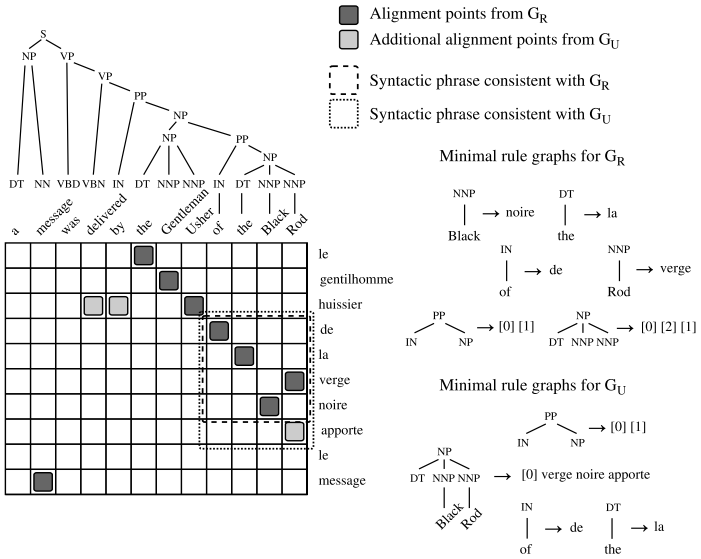
Section 2 reviews the process of translation rule extraction for Syntactic SMT from parallel corpora. Section 3 gives a short introduction to Maximum Entropy Models and details on the features used for the representation of translation rules. In Sect. 4 we compare automatically-created alignments as well as the rule sets generated from these alignments in terms of AER and CTRS. Section 5 gives the results of our filter measured in CTRS, BLEU and NIST. We finish the paper with a discussion of the presented findings.

2 Extraction of Translation Rules

The Syntactic SMT system used in our experiments — Bonsai — is described in [5] and is similar in function to the systems introduced by [6] and [7]. Formally, Bonsai is a tree-to-string transducer [8,9], which requires that the source language is syntactically parsed prior to translation. The parse tree is transformed by translation rules into a flat target language string. This process is guided by a set of probabilistic and heuristic rule features and one or more target language models.

For translation rule extraction, we applied the algorithm proposed by [10]. For a given word-aligned sentence pair and the parse tree of the source language sentence, this algorithm identifies syntactic constituents of the parse tree which are consistent with the word alignment and forms a set of minimal rule graphs. Complex rule graphs can be built from minimal graphs or smaller complex rule graphs by composing source tree fragments at shared nodes and concatenating the target sides of the composed rules. The number of minimal graphs used for the creation of a rule is denoted by k.

Figure 1 illustrates the rule extraction for a sentence pair from the Hansards parallel corpus [11] and two different alignments. These two alignments were created by training GIZA++ in both directions, after which the refined combination method from [12] (denoted by $\mathrm{G_R}$) and union (denoted by $\mathrm{G_U}$) were applied to the directed alignments. Dark gray alignment points belong to $\mathrm{G_R}$ and $\mathrm{G_U}$, while light gray points appear only in $\mathrm{G_U}$. The minimal rule graphs extracted from the marked phrases (dashed boxes) differ in number and form between both alignments, a fact which is caused by a single superfluous alignment point from $\mathrm{G_U}$.



Composed rules from G_R	**Alignment**	**Phrases**	**k**
PP(of NP[0]) → de [0]	(1,1)	(8,12,3,7), (9,12,4,7)	2
PP(of the NNP[0] NNP[1]) → de la [1] [0]	(1,1), (2,2)	(8,12,3,7), (10,11,6,7), (11,12,5,6)	4
PP(IN[0] the Black Rod) → [0] la verge noire	(2,2), (3,4), (4,3)	(8,12,3,7), (8,9,3,4)	5
Composed rules from G_U	**Alignment**	**Phrases**	**k**
PP(of NP[0]) → de [0]	(1,1)	(8,12,3,8), (9,12,4,8)	2
PP(IN[0] the Black Rod) → [0] la verge noire apporte	(2,2), (3,4), (4,3), (4,5)	(8,12,3,8), (8,9,3,4)	3

Fig. 1. Rule extraction and composition

A small sample of more complex rules[2] that can be created by composing the minimal graphs is given together with three types of parameters: rule-internal alignments for terminal symbols, rectangles describing phrases-pairs consistent with root nodes and nonterminal symbols, and the composition factor k.

3 The Maximum Entropy Filter

3.1 Maximum Entropy Models

Maximum entropy models estimate the probability $p(c|x)$ of a class c in a context x. Given a set of facts or constraints, a model is computed that follows all of these constraints but otherwise makes as few assumptions as possible [13].

Constraints are represented as feature functions, in most cases binary functions, $f_i : \mathcal{C} \times \mathcal{X} \rightarrow \{0, 1\}$, where $\mathcal{C}$ is the set of all classes and $\mathcal{X}$ denotes the set

[2] The translation rules used in our syntactic MT system differ slightly from the rules proposed in the majority of similar systems [10,6] as we ignore internal nodes and preserve only information about root nodes and leaves.

of all facts. Each feature function f_i is associated with a model parameter λ_i, the feature weight. Given a set of N feature functions $f_1, \ldots, f_N$, the probability of class c given a context x is equal to:

$$p(c|x) = \frac{1}{Z_x} \exp\left(\sum_{i=1}^{N} \lambda_i f_i(c, x)\right) \tag{1}$$

where Z_x is a normalization constant. The contribution (i.e. the weight λ_i) of each feature function to the final outcome can be computed with the *Generalized Iterative Scaling* (GIS) algorithm [14].

When maximum entropy models are used for hard classification, the class $\hat{c}$ that has the highest probability is chosen, i.e.

$$\hat{c} = \arg\max_{c} p(c|x). \tag{2}$$

For our described binary classification problem, we found it more convenient to take advantage of the whole probability distribution over both classes, using the probability of a chosen class as a threshold.

3.2 Rule Features

Translation rules are processed sentence-wise. Quantitative information that go beyond the scope of a single sentence pair are not available. For an approach to filtering based on the quantitative distribution of phrase-pairs in Phrase-Based SMT see [15]. For each sentence pair $(\mathbf{e}, \mathbf{f})$ one or more rule sets R_m exist, where $r \in R_m$ is a single translation rule. Each set R_m has been generated from an automatically created word alignment A_m. We define $\mathcal{R} = \{R_1, \ldots, R_n\}$ as the set of rule sets available for one sentence pair $(\mathbf{e}, \mathbf{f})$. The rule set R_{H} denotes the set of reference translation rules generated from the human-created word alignment. The filter is supposed to select the rules from the rule sets in $\mathcal{R}$ in such a way that the resulting rule set is closer to R_{H} than any of the input rule sets. The set of classes is $\mathcal{C} = \{\text{“good”}, \text{“bad”}\}$, where the respective classes denote the acceptance or rejection of a translation rule.

From the surface form of a translation rule r, the following features can be derived:

- **$\mathbf{R}_m$, RCount:** Whether r exists in a given rule set R_m and the number of rule sets from $\mathcal{R}$ it exists in.
- **SrcSymLen, TrgSymLen, SrcTrgDiff, SrcTrgEq:** The number of source language symbols (terminal and nonterminal) and target language symbols, their absolute difference and signed equality[3].

[3] We define signed equality as $x \gtreqless y = \begin{cases} -1 & \text{if } x < y, \\ 0 & \text{if } x = y, \\ 1 & \text{if } x > y \end{cases}$.

- **NtCount, SrcTrmCount, TrgTrmCount, SrcHasTrm, TrgHasTrm:** The number of nonterminal symbols, the number of source language (target language) terminal symbols, and whether there are source (target) language terminal symbols.
- **Lhs:** The left-hand side symbol of the rule.
- **NtDist$_j$:** For the j-th nonterminal symbol, the absolute distance between the source language position and the target language position in a rule.
- **SrcPuncCount, TrgPuncCount, SrcTrgPuncEq:** The number of punctuation symbols on the source (target) language side and their signed equality.

The following features are collected during the rule extraction process of r:

- **K:** The number k of minimal graphs used for the composition of rule r.
- **SrcSpan, TrgSpan, SrcTrgSpanDiff:** The number of symbols in the source (target) language span and their absolute difference.
- **Align$_m(i,j)$:** For each rule set R_m[4], all alignment points (i,j) from $A(r)$, where $A(r)$ is the set of internal alignments of a rule r. i is the position of the source language symbol, and j the position of the target language symbol in the rule.
- **SrcAligned, SrcUnaligned, TrgAligned, TrgUnaligned:** The number of aligned and unaligned source (target) language words.

The combination of features and feature values results in a large number of feature functions. For the English-French test set there are more than 1,300 different feature functions, while the Polish-French set has over 1,100. The corresponding model parameters λ_i are learned using the *The OpenNLP Maximum Entropy Package*[5].

4 Alignment Data, Rule Sets, and Metrics

The quality of the described filter is evaluated for two language pairs, English-French and Polish-French. The English-French data was made available at the HLT-NAACL 2003 workshop on "Building and Using Parallel Texts: Data Driven Machine Translation and Beyond" [16] and comprises a subset of the Canadian Hansards [11] and a separate test set of 447 manually word-aligned sentences [12]. For the Polish-French experiments we used a subset of the *Directorate-General for Translation – Translation Memory*[6]. A small subset of 294 sentences from this corpus was set apart and manually annotated with the correct word alignments.[7]

[4] As mentioned before, the rule sets have been generated from different alignments. Rules with the same surface may have different internal alignments for different m.

[5] Available at `http://maxent.sourceforge.net`

[6] Available at `http://langtech.jrc.it/DGT-TM.html`

[7] By the moment this paper is published, manual annotation is still work in progress. The data will be made available once the task is finished. To our knowledge this will be the first word-aligned test set with Polish.

Table 1. Data used for filter training

(a) Word-aligned test data

Languages	Sentences	Source	Rules
English-French	497	HLT/NAACL 2003 and [12]	36,846
Polish-French	294	DGT Translation Memory	25,709

(b) Training data for automatic word alignments

Languages	Sentences	Source
English-French	1,130,550	Hansards [11]
Polish-French	748,734	DGT Translation Memory

The data from Tab. 1b is used to compute several automatic word alignments listed in Tab. 2. Apart from GIZA++ and the BerkeleyAligner [1], we also use a close implementation of the supervised word alignment combination method (ACME) proposed by [18], which has been trained on the human-created word alignments and three automatically created alignments (the two directed alignments and BA). In order to reduce data sparseness introduced by the rich morphology of the Polish language, word alignment computation was carried out for a lemmatized version of the Polish-French corpus. The English-French corpus was not preprocessed in this way.

A translation rule set that was created from a given word alignment is identified by the same symbol as its underlying alignment. It should follow from the context whether we refer to the underlying alignment or the generated rule set. English source language parses of all English-French data have been produced with the Stanford Parser [19]. Polish parse trees for the Polish-French data have been created with the internal parser of the Bonsai Syntactic SMT system.

The purpose of the manually word-aligned sentences from Tab. 1a is twofold. Firstly, for each language pair these sentences are used to measure the AER of the automatically created alignments. Secondly, they serve as the basis for the extraction of the reference rule set R_{H} that will be used to train the described maximum entropy model as well as for its evaluation.

Table 2. Automatically created word alignments

Symbol		Description
$\mathrm{G_{EF}}$	$\mathrm{G_{FE}}$	Directed en-fr and fr-en alignments created with GIZA++
$\mathrm{G_{PF}}$	$\mathrm{G_{FP}}$	Directed pl-fr and fr-pl alignments created with GIZA++
$\mathrm{G_I}$		Intersection of the directed word alignments
$\mathrm{G_R}$		Refined [12] combination of the directed word alignments
$\mathrm{G_G}$		Grow-Diag-Final [17] combination of the directed word alignments
$\mathrm{G_U}$		Union of the directed word alignments
BA		BerkeleyAligner [1] joint word alignment model
ACME		A supervised word alignment combination method [18]

Table 3. Comparison of AER for both language pairs

(a) en-fr

Align	Pr	Rc	AER
G_I	98.25	80.16	10.47
G_R	92.39	91.88	7.82
G_G	86.98	94.13	10.33
G_U	85.47	94.85	11.10
BA	90.74	95.99	7.24
ACME	95.47	94.72	4.84

(b) pl-fr

Align	Pr	Rc	AER
G_I	95.60	50.04	34.31
G_R	83.98	64.46	27.06
G_G	76.07	67.41	28.52
G_U	74.11	68.84	28.62
BA	82.98	63.89	27.81
ACME	86.54	75.49	19.36

4.1 Word Alignment Error Rate

The standard metric *Alignment Error Rate* (AER) proposed by [12] is used to evaluate the quality of the introduced input word alignments. AER is calculated as follows:

$$\mathrm{Pr} = \frac{|A \cap P|}{|A|} \qquad \mathrm{Rc} = \frac{|A \cap S|}{|S|} \qquad \mathrm{AER} = 1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|} \tag{3}$$

where P is the set of possible alignment points in the reference alignment, S is the set of sure alignments in the reference alignment ($S \subset P$), and A is the evaluated word alignment.

The results for all alignment methods have been compiled into Tab. 3. There are large qualitative differences concerning the alignment error rate between both language pairs, which are probably caused by the greater dissimilarity of Polish and French as well as by the characteristics of the utilised test sets. The relative number of possible alignments in the English-French test set is much greater than in its Polish-French counterpart. This makes the English-French test set more forgiving of erroneous alignments.

4.2 Consistent Translation Rule Score

So far we have not defined a formal way to measure the quality of a set of translation rules against the reference rule set R_H. For this purpose, we have adapted the *Consistent Phrase Error Rate* (CPER) from [4] to the needs of syntactic translation rules.[8] To emphasize the application of CPER to syntactic translation rules we have renamed it to *Consistent Translation Rule Score* (CTRS) and calculate it as follows:

$$\mathrm{Pr} = \frac{|R \cap R_H|}{|R|} \qquad \mathrm{Rc} = \frac{|R \cap R_H|}{|R_H|} \qquad \mathrm{CTRS} = \frac{2 \cdot \mathrm{Pr} \cdot \mathrm{Rc}}{\mathrm{Pr} + \mathrm{Rc}} \tag{4}$$

where R_H is a rule set consistent with a human-created alignment and R a rule set consistent with an automatically generated word alignment. The original

[8] The same approach has been proposed by [1] to show that the syntactic HMM word alignment models implemented in the BerkeleyAligner allow to create better and more general tree transducer rules. An evaluation of MT quality was not given.

Table 4. Comparison of CTRS for input alignments

(a) en-fr

Align	Pr	Rc	CTRS	Rules
G_I	35.49	33.94	34.70	35,115
G_R	38.32	46.89	42.17	44,977
G_G	43.99	44.20	44.09	36,972
G_U	45.39	42.43	43.86	34,403
BA	41.16	50.82	45.49	45,364
ACME	44.02	55.45	**49.08**	46,285

(b) pl-fr

Align	Pr	Rc	CTRS	Rules
G_I	31.09	27.00	28.90	22,291
G_R	34.80	32.62	33.67	24,017
G_G	39.97	28.98	33.60	18,708
G_U	41.57	27.45	33.07	17,065
BA	37.45	36.21	36.82	24,793
ACME	60.99	50.44	**55.22**	21,245

CPER is calculated as $1 - \text{F-score}$, for CTRS we find F-score more appropriate since an increase in F-score can be directly interpreted as an increase in the quality of a rule set.

According to [4], CPER penalizes incorrect or missing alignment links more severely than AER. When AER is computed, one incorrect alignment link reduces the number of correct alignments by one, which results in slight decreases in precision and recall, while missing alignment links result in a small decrease in recall only. For CPER, incorrect or missing links may result in the elimination or addition of more than one phrase pair and thus have a stronger influence on precision and recall. This is even truer of CTRS and Syntactic SMT, where many translation rules can be created from one phrase pair.

Table 4 depicts the CTRS results of the rule sets generated from the input alignments. The absolute number of rules generated for the test set is also given. For the GIZA++ derived rule sets, a reverse trend can be seen when CTRS is compared to AER: the rule sets based on recall-oriented alignments yielded a higher precision, while the rule sets created from alignments with a higher precision had higher values for CTRS recall. The observed changes in balance between precision and recall for the majority of the data sets can be explained by the way phrases and translation rules are built from alignments. Recall-oriented alignments generally result in a smaller number of phrases since the presence of more alignment points forces the creation of longer phrase pairs. In syntax-based MT, however, these phrases must be consistent with the provided parse trees, otherwise no rules are created.

5 Experiments and Evaluation

5.1 Filter Parameters

Choice of Input Rule Sets. During training, the rule sets generated from all discussed input alignments were used and the filter achieved a CTRS of 56.16%. Removing any single rule set from the training set resulted in drops in CTRS, e.g. the filter's performance dropped to 52.54% CTRS if the ACME rule set was removed. The distance of ca. 7% to the remaining best performing single rule

Table 5. Probabilities for some example rules

Translation Rule	**Probability**
PP(of NP[0]) → de [0]	0.7920
PP(of the NNP[0] NNP[1]) → de la [1] [0]	0.7060
PP(IN[0] the Black Rod) → [0] la verge noire	0.7729
PP(IN[0] the Black Rod) → [0] la verge noire apporte	0.1642

set BA (45.49%) persisted. Removing only the BA rule set showed no significant impact on CTRS (56.00%). However, if both ACME and BA were removed, CTRS decreased to 49.04%. Similar results were obtained if other rule sets were removed from the training data.

A second matter of interest concerns the filter performance when only single rule sets are given as input data. Using only G_U for training yielded a CTRS of 45.75% compared to 43.86% for the unfiltered version. Repeating the experiment for ACME alone resulted in 53.23% CTRS compared to 49.08% for the input rule set. The application of the filter to a single rule set changes the balance between precision and recall. For the ACME rule set precision now amounts to 66.84% and recall to 44.22% compared to a precision of 44.02% and a recall of 55.45% for the original rule set. The number of rules was reduced by roughly 50%.

Feature Selection and Partitioning. For the English-French language pair a CTRS of 56.16% was reached when all of features described in 3.2 were used. Removing single features resulted in only small changes, while the impact of removing groups of related features was more significant. The greatest impact was observed if all alignment-related features (**Align**$_m(i,j)$**, SrcAligned, SrcUnaligned, TrgAligned, TrgUnaligned**) were discarded, CTRS dropped then to 54.34%.

For their maximum entropy based word-alignment combination method[9], [4] observed that a partition into distinct models based on the values of selected features may result in improvements. We tested this approach for different rule features and feature combinations and found that a partition based on the following features works best: **SrcSymLen** (56.60%), **K** (56.54%), and a combination of both features (56.65%).

Balancing Precision and Recall via Manually Set Thresholds. Let r be a translation rule and x_r the context or feature set representing r. Then we define $\mathrm{F}(t)$ as the rule set generated by the filter at a threshold t as

$$\mathrm{F}(t) = \{\ r : p(\text{“good”}|x_r) \geq t\ \} \tag{5}$$

where $p(c|x)$ is the probability distribution defined in (1). Table 5 contains the probabilities for the example rules from Fig. 1. The last rule, created due to an incorrect alignment link, would be discarded for an appropriate threshold.

[9] The alignment method ACME is an implementation of this method.

Table 6. Comparison of rule quality according to the test set

(a) en-fr

Filter	Pr	Rc	CTRS	Rules
F(0.2)	54.87	55.67	55.27	36,745
F(0.3)	63.48	50.94	**56.52**	29,068
F(0.4)	70.46	45.34	55.17	23,306
F(0.5)	73.92	40.28	52.14	19,735
F(0.6)	77.28	33.59	46.82	15,743
F(0.7)	82.58	24.47	37.75	10,732

(b) pl-fr

Filter	Pr	Rc	CTRS	Rules
F(0.2)	57.59	59.22	58.39	24,967
F(0.3)	66.79	54.24	**59.86**	19,716
F(0.4)	74.21	49.39	59.31	16,160
F(0.5)	78.32	45.04	57.19	13,960
F(0.6)	80.61	39.56	53.08	11,915
F(0.7)	83.50	30.50	44.69	8,869

For the machine translation task, we decided to chose six thresholds, from 0.2 to 0.7 with a step of 0.1, and present the CTRS for both test sets in Tab. 6. All results were obtained using 5-fold cross-validation for the respective test sets. As defined in (5), the symbol F(t) denotes the rule set generated by the filter at a threshold t. The extreme values for precision and recall differ between the rule sets by roughly 30%.

5.2 MT Evaluation

In this section, we will give the machine translation results for all introduced rule sets — alignment-based and filtered. Translation quality is measured with lowercased BLEU-4 and NIST. All rule sets have been generated from the first 100,000 sentences of the two previously described training corpora. This size limit is purely technical since we have to deal with 17 distinct rule sets for each language pair. Machine translation test sets for both language pairs comprise the last 1,500 sentences from the respective corpora, while the development sets (1,000 sentences each) have been taken from the middle of the same corpora. Translation model weights of the decoder for each rule set have been optimised on the development set with Z-MERT [20].

The machine translation results are described in Tab. 7 (English-French) and Tab. 8 (Polish-French). For the English-French language pair, G_U performed best among the unfiltered rule sets and G_G reached the second best scores for all metrics. Rather surprising are the weak MT results for ACME, BA and G_R since the underlying alignments of these rule sets scored best in terms of AER and the first two rule sets showed the best CTRS results. A very similar situation can be observed for the unfiltered Polish-French rule sets.

For the English-French filtered rule sets, F(0.4) showed the best BLEU score and F(0.5) the best scores for NIST among all evaluated rule sets. The rule sets F(0.2) to F(0.5) outperformed the best unfiltered rule set for all three metrics, F(0.6) had better results for NIST. The number of unique rules in each rule set is also given. F(0.4) consists of roughly 75% fewer rules than ACME and 60% fewer than G_U. Negative effects of data sparseness seem to manifest somewhere between a filter threshold of 0.6 and 0.7. Results for the Polish-French language pair are similar though less significant.

Table 7. MT scores for English-French language pair

(a) Input rule sets

Align	BLEU	NIST	Rules
G_I	0.1918	5.2983	5,023,457
G_R	0.1738	4.9788	6,148,095
G_G	0.2006	5.3016	4,576,837
G_U	**0.2049**	**5.3678**	4,182,497
BA	0.1923	5.2168	6,037,889
ACME	0.1891	5.1620	6,269,929

(b) Filtered rule sets

Filter	BLEU	NIST	Rules
F(0.2)	0.2079	5.4456	3,533,210
F(0.3)	0.2093	5.5091	2,253,812
F(0.4)	**0.2127**	5.7492	1,584,581
F(0.5)	0.2090	**5.8208**	1,243,441
F(0.6)	0.2031	5.7821	926,183
F(0.7)	0.1570	4.7648	605,946

Table 8. MT scores for Polish-French language pair

(a) Input rule sets

Align	BLEU	NIST	Rules
G_I	0.2955	6.1520	4,327,075
G_R	0.3031	6.1183	4,572,431
G_G	0.3200	6.4454	3,136,140
G_U	**0.3218**	**6.5050**	2,768,452
BA	0.3060	6.2897	4,562,305
ACME	0.2989	6.1283	3,625,969

(b) Filtered Rule Sets

Filter	BLEU	NIST	Rules
F(0.2)	0.3138	6.3625	3,855,027
F(0.3)	0.3144	6.4079	2,624,104
F(0.4)	0.3246	6.5865	1,894,711
F(0.5)	**0.3301**	**6.7269**	1,505,139
F(0.6)	0.3168	6.6654	1,161,098
F(0.7)	0.2656	5.6743	744,163

It is worth mentioning that the best MT results were reached for both language pairs at thresholds close to 0.5. In terms of the used maximum entropy model, this means that we could revert from thresholding strategies and return to hard classification as defined in (2). If this could be shown to be a generally valid result, it would confirm that the rules classified as "good" — and therefore more similar to those generated from a manually-created word alignment — are indeed well suited for Syntactic SMT. Since thresholds were chosen arbitrarily, we cannot say whether a threshold exists that would yield better MT quality. Hence, one direction for further research should include threshold optimization in terms of BLEU scores on a given development set.

6 Discussion

Previous work [4] has shown that improved results for AER and CPER (or CTRS in this work) are not good indicators for Phrase-Based SMT quality. In the context of Syntactic SMT, these findings can be confirmed for the alignments generated by the BerkeleyAligner (BA) and especially the supervised alignment method ACME. The MT results for both rule sets are significantly worse than for G_U and G_G although they show superior AER and CTRS scores. Similarly, the filtered rule sets with the highest CTRS do not reach the best MT scores, but are exceeded by filters with higher thresholds. However, the differences in CTRS between these rule sets are rather small. All rule sets that reached high

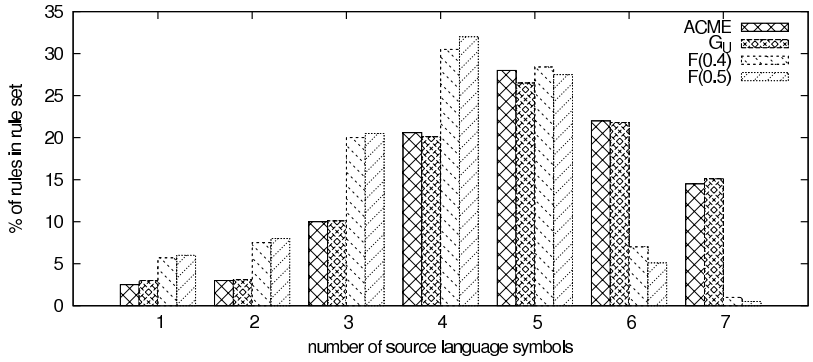


Fig. 2. Histogram of rule lengths for chosen unfiltered and filtered rule sets

MT results maintain a relatively high CTRS and prefer CTRS precision over CTRS recall. This is equally true for the unfiltered and filtered rule sets. High CTRS precision is generally connected with high AER recall.

For the alignment-based rule sets, the worst performing sets have the greatest number of rules and vice versa. The same is true for the filtered rule sets if we disregard F(0.7). The chart in Fig. 2 allows us to compare the distribution of rule lengths (the number of source language symbols) for four chosen rule sets: ACME, G_U, F(0.4), and F(0.5). There are no significant differences between the two unfiltered rule sets or between both filtered rule sets. However, when comparing the filtered rule sets to the unfiltered ones, we can see that the majority of rules longer than 5 symbols has been discarded. The decrease in number of long rules is the main factor behind the size reduction of the filtered sets. Since long rules will only be used in specific construction, it is possible that the final effect of the filtration is in some degree equivalent to the effects of significance testing described by [15] for Phrase-Based SMT, which might be an explanation for the better MT results obtained by the filtered rule sets.

We have shown that a maximum entropy model trained on a reference rule set generated from manual alignments can improve machine translation quality and reduce the number of translation rules at the same time. This simple approach could improve CTRS several percent over the best unfiltered rule set even if only one rule set is used. The findings of other researchers that AER is not necessarily related to MT quality have been confirmed; for CTRS, however, a relationship between better MT results and higher CTRS precision seems to exist. From this, it follows that alignment combination methods that aim for recall seem to be better suited for Syntactic SMT than precision-oriented methods, a result that contradicts those presented by [4] for Phrase-Based SMT.

Acknowledgements

This paper is based on research funded by the Polish Ministry of Science and Higher Education (Grant No. 003/R/T00/2008/05).

References

1. DeNero, J., Klein, D.: Tailoring word alignments to syntactic machine translation. In: Proceedings of ACL, pp. 17–24 (2007)
2. Fossum, V., Knight, K., Abney, S.: Using syntax to improve word alignment precision for syntax-based machine translation. In: Proceedings of ACL Workshop on Statistical Machine Translation, pp. 44–52 (2008)
3. Zollmann, A., Venugopal, A., Och, F., Ponte, J.: A systematic comparison of phrase-based, hierarchical and syntax-augmented statistical MT. In: Proceedings of ACL-COLING, pp. 1145–1152 (2008)
4. Ayan, N.F., Dorr, B.J.: Going beyond AER: an extensive analysis of word alignments and their impact on MT. In: Proceedings of ACL-COLING, pp. 9–16 (2006)
5. Junczys-Dowmunt, M.: It's all about the trees — towards a hybrid syntax-based MT system. In: Proceedings of IMCSIT, pp. 219–226 (2009)
6. Huang, L.: Statistical syntax-directed translation with extended domain of locality. In: Proceedings of AMTA, pp. 66–73 (2006)
7. Liu, Y., Liu, Q., Lin, S.: Tree-to-string alignment template for statistical machine translation. In: Proceedings of ACL, pp. 609–616 (2006)
8. Aho, A.V., Ullman, J.D.: Translations on a context-free grammar. Information and Control 19, 439–475 (1971)
9. Graehl, J., Knight, K.: Training tree transducers. In: Proceedings of HLT-NAACL, pp. 105–112 (2004)
10. Galley, M., Hopkins, M., Knight, K., Marcu, D.: What's in a translation rule. In: Proceedings of HLT-NAACL, pp. 273–280 (2004)
11. Germann, U.: Aligned hansards of the 36th parliament of Canada (2001)
12. Och, F.J., Ney, H.: A systematic comparison of various statistical alignment models. Computational Linguistics 29, 19–51 (2003)
13. Berger, A.L., Della Pietra, V.J., Della Pietra, S.A.: A maximum entropy approach to natural language processing. Computational Linguistics 22, 39–71 (1996)
14. Darroch, J., Ratcliff, D.: Generalized iterative scaling for log-linear models. The Annals of Mathematical Statistics 43, 1470–1480 (1972)
15. Johnson, H., Martin, J., Foster, G., Kuhn, R.: Improving translation quality by discarding most of the phrasetable. In: Proceedings of EMNLP-CoNLL, pp. 967–975 (2007)
16. Mihalcea, R., Pedersen, T.: An evaluation exercise for word alignment. In: Proceedings of HLT-NAACL, pp. 1–10 (2003)
17. Koehn, P., Och, F., Marcu, D.: Statistical phrase-based translation. In: Proceedings of HLT-NAACL, pp. 48–54 (2003)
18. Ayan, N.F., Dorr, B.J.: A maximum entropy approach to combining word alignments. In: Proceedings of HLT-NAACL, pp. 96–103 (2006)
19. Klein, D., Manning, C.D.: Accurate unlexicalized parsing. In: Proceedings of ACL, pp. 423–430 (2003)
20. Zaidan, O.F.: Z-MERT: A fully configurable open source tool for minimum error rate training of machine translation systems. The Prague Bulletin of Mathematical Linguistics 91, 79–88 (2009)