Learning Sentential Paraphrases from Bilingual Parallel Corpora for Text-to-Text Generation

Abstract

Previous work has shown that high quality *phrasal* paraphrases can be extracted from bilingual parallel corpora. However, it is not clear whether bitexts are an appropriate resource for extracting more sophisticated *sentential* paraphrases, which are more obviously learnable from monolingual parallel corpora. We extend bilingual paraphrase extraction to syntactic paraphrases and demonstrate its ability to learn a variety of general paraphrastic transformations, including passivization, dative shift, topicalization, etc. We discuss how our model can be adapted to many text generation tasks, by augmenting its feature set, development data and parameter estimation routine. We illustrate this adaptation by using our paraphrase model for the task of sentence compression and achieve results that improve on state-of-the-art compression systems.

1 Introduction

Paraphrases are alternative ways of expressing the same information (Culicover, 1968). Automatically generating and detecting paraphrases is a crucial aspect of many NLP tasks. In multi-document summarization, paraphrase detection is used to collapse redundancies (Barzilay et al., 1999; Barzilay, 2003). Paraphrase generation can be used for query expansion in information retrieval and question answering systems (McKeown, 1979; Anick and Tipirneni, 1999; Ravichandran and Hovy, 2002; Riezler et al., 2007). Finally, paraphrases allow for more flexible matching of system output against human references for tasks like MT and automatic summarization (Zhou et al., 2006; Kauchak and Barzilay, 2006; Madnani et al., 2007; Snover et al., 2010).

Broadly, we can distinguish two forms of paraphrases: *phrasal paraphrases* denote a set of surface text forms with the same meaning:

the committee's second proposal
the second proposal of the committee ,

while *syntactic paraphrases* augment the surface forms by introducing nonterminals (or *slots*) that are annotated with syntactic constraints:

the NP_1's NP_2
the NP_2 of the NP_1

It is evident that the latter have a much higher potential for generalization and for capturing interesting paraphrastic transformations.

A variety of different types of corpora (and semantic equivalence cues), have been used to automatically induce paraphrase collections for English (Madnani and Dorr, 2010). The perhaps most natural type of corpus for this task is a monolingual parallel text, from which paraphrases can be extracted by leveraging the fact that the given sentence pairs are perfect paraphrases of each other (Barzilay and McKeown, 2001; Pang et al., 2003). While rich syntactic paraphrases have been learned from such corpora, they suffer from very limited data availability and thus have poor coverage.

Other methods strive to obtain paraphrases from raw monolingual text, replacing the exact correspondence of sentences in monolingual parallel corpora with distributional similarity (Lin and Pantel, 2001; Bhagat and Ravichandran, 2008). While vast amounts of data are readily available for these approaches, the correspondency information they employ is weaker and suffers from problems such as mistaking cousin expressions or antonyms (such as $\{boy, girl\}$ or $\{rise, fall\}$) for paraphrases.

Abundantly available bilingual parallel corpora have been shown to address both these issues, obtaining paraphrases via a pivoting step over foreign language phrases (Bannard and Callison-Burch, 2005). The coverage of paraphrase lexica extracted from bitexts has been shown to outperform that obtained from other sources (Zhao et al., 2008a). While there have been efforts pursuing the extraction of more powerful paraphrases (Madnani et al., 2007; Callison-Burch, 2008; Cohn and Lapata, 2008; Zhao et al., 2008b), it is not yet clear to which extent sentential paraphrases can be induced from bitexts. In this paper we:

- Extend the bilingual pivoting approach to paraphrase induction to produce rich syntactic paraphrases.
- Perform a thorough analysis of the types of paraphrases we obtain, and discuss the paraphrastic transformations we are capable of capturing.
- Describe how design and training paradigms for syntactic/sentential paraphrase models should be tailored to different text-to-text generation tasks.
- Demonstrate our framework's suitability for a variety of text-to-text generation tasks on the example of sentence compression, obtaining state-of-the-art results.

2 Related Work

Madnani and Dorr (2010) survey a variety of data-driven paraphrasing techniques, categorizing them based on the type of data that they use. These include large monolingual texts (Lin and Pantel, 2001; Szpektor et al., 2004; Bhagat and Ravichandran, 2008), comparable corpora (Barzilay and Lee, 2003; Dolan et al., 2004), monolingual parallel corpora (Barzilay and McKeown, 2001; Pang et al., 2003), and bilingual parallel corpora (Bannard and Callison-Burch, 2005; Madnani et al., 2007; Zhao et al., 2008b). We focus on the latter type of data.

Paraphrase extraction using bilingual parallel corpora was proposed by Bannard and Callison-Burch (2005) who induced paraphrases using techniques from *phrase-based* statistical machine translation (Koehn et al., 2003). After extracting a bilingual phrase table, English paraphrases can be obtained by pivoting through foreign language phrases. Since many paraphrases can be extracted for a phrase, Bannard and Callison-Burch rank them using a paraphrase probability defined in terms of the translation model probabilities $p(f|e)$ and $p(e|f)$:

$$p(e_2|e_1) = \sum_f p(e_2, f|e_1) \quad (1)$$

$$= \sum_f p(e_2|f, e_1)p(f|e_1) \quad (2)$$

$$\approx \sum_f p(e_2|f)p(f|e_1) \quad (3)$$

Several subsequent efforts extended the bilingual pivoting technique, many of which introduced elements of more contemporary *syntax-based* approaches to statistical machine translation. Madnani et al. (2007) extended the technique to *hierarchical* phrase-based machine translation (Chiang, 2005), which is formally a synchronous context free grammar (SCFG) and thus can be thought of as a *paraphrase grammar*. The paraphrase grammar can paraphrase (or "decode") input sentences using an SCFG decoder, like the Hiero, Joshua or cdec MT systems (Chiang, 2007; Li et al., 2009; Dyer et al., 2010). Like Hiero, Madnani's model uses only a single nonterminal symbol X instead of linguistic non-terminals.

Three additional efforts incorporated linguistic syntax. Callison-Burch (2008) introduced syntactic constraints by labeling all phrases and paraphrases (even non-constituent phrases) with CCG slash categories (Steedman, 1999), an approach similar to Zollmann and Venugopal (2006)'s syntax-augmented machine translation (SAMT). Callison-Burch did not formally define a synchronous grammar, nor discuss decoding, since his presentation did not include hierarchical rules. Cohn and Lapata (2008) use the 'GHKM' extraction method (Galley et al., 2004), which is limited to constituent phrases and thus produces a reasonably small set of syntactic rules. Zhao et al. (2008b) added slots to bilingually-extracted paraphrase patterns that were labeled with part-of-speech tags, but not larger syntactic constituents.

Before the shift to statistical natural language pro-

cessing, paraphrasing was often treated as syntactic transformations, or by parsing and then generating from a semantic representation (McKeown, 1979; Muraki, 1982; Meteer and Shaked, 1988; Shemtov, 1996; Yamamoto, 2002). Indeed, some work generated paraphrases using (non-probabilistic) synchronous grammars (Shieber and Schabes, 1990; Dras, 1997; Dras, 1999; Kozlowski et al., 2003).

After the rise of statistical machine translation, a number of its techniques were re-purposed for paraphrasing. These include: sentence alignment (Gale and Church, 1993; Barzilay and Elhadad, 2003), word alignment and noisy channel decoding (Brown et al., 1990; Quirk et al., 2004), phrase-based models (Koehn et al., 2003; Bannard and Callison-Burch, 2005), hierarchical phrase-based models (Chiang, 2005; Madnani et al., 2007), log-linear models and minimum error rate training (Och, 2003a; Madnani et al., 2007; Zhao et al., 2008a), and here syntax-based machine translation (Wu, 1997; Yamada and Knight, 2001; Melamed, 2004; Quirk et al., 2005).

Beyond cementing the ties between paraphrasing and syntax-based statistical machine translation, the novel contributions of our paper are (1) an in-depth analysis of the types of structural and sentential paraphrases that can be extracted with bilingual pivoting, (2) a discussion on how our English-English paraphrase grammar should be adapted to specific text-to-text generation tasks (Zhao et al., 2009) with (3) a concrete example of the adaptation procedure for the task of paraphrase-based sentence compression (Knight and Marcu, 2002; Cohn and Lapata, 2008; Cohn and Lapata, 2009).

3 SCFGs in Translation

The model we use in our paraphrasing approach is a syntactically informed *synchronous context-free grammar* (SCFG). The SCFG formalism (Aho and Ullman, 1972) was re-popularized for statistical machine translation by Chiang (2005). Formally, a *probabilistic* SCFG $\mathcal{G}$ is defined by specifying

$$\mathcal{G} = \langle \mathcal{N}, \mathcal{T}_S, \mathcal{T}_T, \mathcal{R}, S \rangle,$$

where $\mathcal{N}$ is a set of nonterminal symbols, $\mathcal{T}_S$ and $\mathcal{T}_T$ are the source and target language vocabularies, $\mathcal{R}$ is a set of rules and $S \in \mathcal{N}$ is the root symbol. The

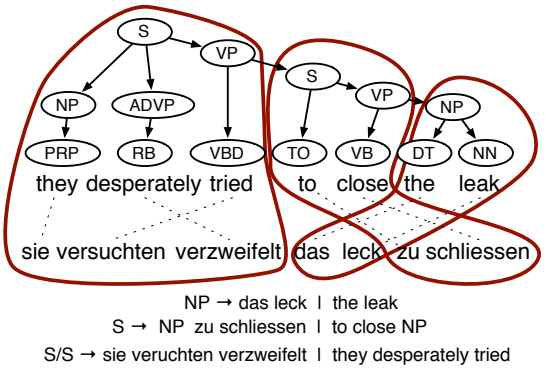


Figure 1: Synchronous grammar rules for translation are extracted from sentence pairs in a bixtext which have been automatically parsed and word-aligned. Extraction methods vary on whether they extract only minimal rules for phrases dominated by nodes in the parse tree, or more complex rules that include non-constituent phrases.

rules in $\mathcal{R}$ take the form:

$$C \rightarrow \langle \gamma, \alpha, \sim, w \rangle,$$

where the rule's left-hand side $C \in \mathcal{N}$ is a nonterminal, $\gamma \in (\mathcal{N} \cup \mathcal{T}_S)^*$ and $\alpha \in (\mathcal{N} \cup \mathcal{T}_T)^*$ are strings of terminal and nonterminal symbols with an equal number of nonterminals $c_{NT}(\gamma) = c_{NT}(\alpha)$ and:

$$\sim: \{1 \ldots c_{NT}(\gamma)\} \rightarrow \{1 \ldots c_{NT}(\alpha)\}$$

constitutes a one-to-one correspondency function between the nonterminals in γ and α. A non-negative weight $w \geq 0$ is assigned to each rule, reflecting the likelihood of the rule.

Rule Extraction Phrase-based approaches to statistical machine translation (and their successors) extract pairs of (e, f) phrases from automatically word-aligned parallel sentences. Och (2003b) described various heuristics for extracting phrase alignments from the Viterbi word-level alignments that are estimated using Brown et al. (1993) word-alignment models.

These phrase extraction heuristics have been extended so that they extract synchronous grammar rules (Galley et al., 2004; Chiang, 2005; Zollmann and Venugopal, 2006; Liu et al., 2006). Most of these extraction methods require that one side of the parallel corpus be parsed, typically this is done automatically with a statistical parser.

Figure 1 shows examples of rules obtained from a sentence pair. To extract a rule, we first choose a source side span f like *das leck*. Then we use phrase extraction techniques to find target spans e that are consistent with the word alignment (in this case *the leak* is consistent with our f). The nonterminal symbol that is the left-hand side of the SCFG rule is then determined by the syntactic constituent that dominates e (in this case *NP*). To introduce nonterminals into the righthand side of the rule, we can apply rules extracted over sub-phrases of f, synchronously substituting the corresponding nonterminal symbol for the sub-phrases on both sides. The synchronous substitution applied to f and e then yields the correspondency $\sim$.

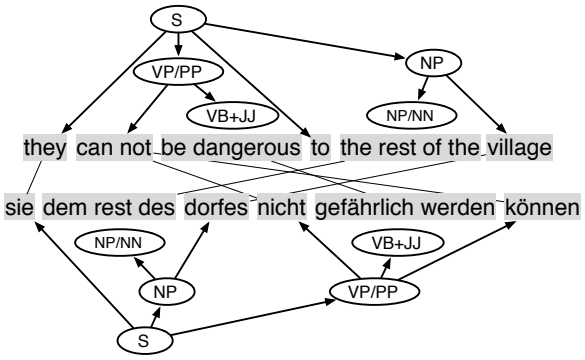
One significant differentiating factor between the competing ways of extracting SCFG rules is whether the extraction method generates rules only for constituent phrases that are dominated by a node in the parse tree (Galley et al., 2004; Cohn and Lapata, 2008) or whether they include arbitrary phrases, including non-constituent phrases (Zollmann and Venugopal, 2006; Callison-Burch, 2008). We adopt the extraction for all phrases, including non-constituents, since it allows us to cover a much greater set of phrases, both in translation and paraphrasing.

Feature Functions Rather than assigning a single weight w, we define a set of feature functions $\vec{\varphi} = \{\varphi_i\}$ that are combined in a log-linear model:

$$w = -\sum_{i}^{N} \lambda_i \log \varphi_i. \quad (4)$$

The weights $\vec{\lambda}$ of these feature functions are set to maximize some objective function like BLEU (Papineni et al., 2002) using a procedure called minimum error rate training (MERT), owing to Och (2003a). MERT iteratively adjusts the weights until the decoder produces output that best matches reference translations in a development set, according to the objective function. We will examine appropriate objective functions for text-to-text generation tasks in Section 6.3.

Typical features used in statistical machine translation include: phrase translation probabilities (calculated using maximum likelihood estimation over all phrase pairs that are enumerable in the parallel corpus), word-for-word lexical translation probabilities (which help to smooth the the phrase translation estimates), a 'rule application penalty' (which governs the system prefers fewer longer phrases or a greater number of shorter phrases), and a language model probability.



NP/NN → dem rest des | the rest of the
VB+JJ → gefährlich werden | be dangerous
NP → NP/NN dorfes | NP/NN village
VP/PP → nicht VB+JJ können | can not VB+JJ
S → sie NP VP/PP | they VP/PP to NP

Figure 2: An example derivation produced by a syntactic machine translation system. Although the synchronous trees are unlike the derivations found in the Penn Treebank, their yield is a good translation of the German.

Decoding Given an SCFG and an input source sentence, the decoder performs a search for the single most probable derivation via the CKY algorithm. In principle the best translation should be that English sentence e that is the most probable after summing over all $d \in D$ derivations, since many derivations yield the same e. In practice, we use a Viterbi approximation and return the translation that is the yield of the single best derivation:

$$\begin{aligned} \hat{e} &= \arg \max_{e \in Trans(f)} \sum_{d \in D(e,f))} p(d, e|f) \\ &\approx yield(\arg \max_{d \in D(e,f))} p(d, e|f)). \end{aligned} \quad (5)$$

Derivations are simply successive applications on the SCFG rules such as those given in Figure 2.

4 SCFGs in Paraphrasing

Rule Extraction To create a paraphrase grammar from a translation grammar, we extend the syntactically informed pivot approach of Callison-Burch (2008) to the SCFG model. For this purpose, we assume a grammar that translates from a given foreign language to English. For each pair of translation rules where the left-hand side C and foreign string γ match:

$$C \rightarrow \langle \gamma, \alpha_1, \sim_1, \vec{\varphi}_1 \rangle$$
$$C \rightarrow \langle \gamma, \alpha_2, \sim_2, \vec{\varphi}_2 \rangle,$$

we create a paraphrase rule:

$$C \rightarrow \langle \alpha_1, \alpha_2, \sim, \vec{\varphi} \rangle,$$

where the nonterminal correspondency relation $\sim$ has been set to reflect the combined nonterminal alignment:

$$\sim \;\; = \;\; \sim_1^{-1} \circ \sim_2 \; .$$

Feature Functions In the computation of the features $\vec{\varphi}$ from $\vec{\varphi}_1$ and $\vec{\varphi}_2$ we follow the approximation in Equation 3, which yields lexical and phrasal paraphrase probability features. Additionally, we add a boolean indicator for whether the rule is an identity paraphrase, $\delta_{identity}$. Another indicator feature, $\delta_{reorder}$, fires if the rule swaps the order of two nonterminals, which enables us to promote more complex paraphrases that require structural reordering.

Decoding With this, paraphrasing becomes an English-to-English translation problem which can be formulated similarly to Equation 5 as $\hat{e_2} \approx yield(\arg\max_{d \in D(e_2, e_1))} p(d, e_2|e_1))$. Figure 3 shows an example derivation produced as a result of applying our paraphrase rules in the decoding process. Another advantage of using the decoder from statistical machine translation is that n-gram language models, which have been shown to be useful in natural language generation (Langkilde and Knight, 1998), are already well integrated (Huang and Chiang, 2007).

5 Analysis

A key motivation for the use of syntactic paraphrases over their phrasal counterparts is their potential to capture meaning-preserving linguistic transformations in a more general fashion. In many cases a phrasal system will be limited to memorizing the fully lexicalized variants of such a transfomation in its paraphrase table, resulting in poor generalization capabilities. By contrast, a syntactic paraphrasing system intuitively should be able to address this issue and learn well-formed and generic patterns that can be easily applied to unseen data.

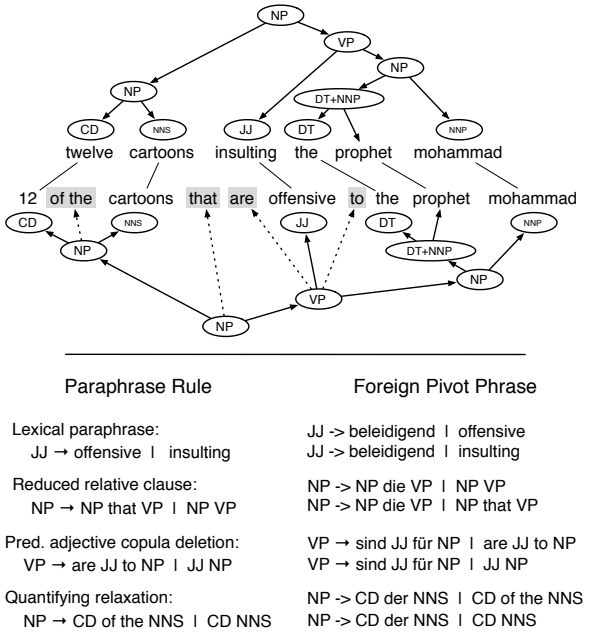


Figure 3: An example of a synchronous paraphrastic derivation. A few of the rules applied in the parse are show in the left column, with the pivot phrases that gave rise to them on the right.

To put this expectation to the test, we investigate how our grammar captures a number of well-known paraphrastic transformations.[1] Table 1 shows the transformations along with examples of the generic grammar rules our system learns to represent them. When given a transformation to extract a syntactic paraphrase for, we want to find rules that neither under- nor over-generalize. This means that, while replacing the maximum number of syntactic arguments with nonterminals, the rules ideally will both retain enough lexicalization to serve as sufficient evidence for the applicability of the transformation and

[1] The data and software used to extract the grammar we draw these examples from is described in Section 6.5.

Possessive rule	$NP \rightarrow$	the NN of the NNP	|	the NNP's NN
	$NP \rightarrow$	the NNS_1 made by NNS_2	|	the NNS_2's NNS_1
Dative shift	$VP \rightarrow$	give NN to NP	|	give NP the NN
	$VP \rightarrow$	provide NP_1 to NP_2	|	give NP_2 NP_1
Adv./adj. phrase move	$S/VP \rightarrow$	$ADVP$ they VBP	|	they VPB $ADVP$
	$S \rightarrow$	it is $ADJP$ VP	|	VP is $ADJP$
Verb particle shift	$VP \rightarrow$	VB NP up	|	VB up NP
Reduced relative clause	$SBAR/S \rightarrow$	although PRP VBP that	|	although PRP VBP
	$ADJP \rightarrow$	very JJ that S	|	JJ S
Quantificational variants	$NP \rightarrow$	CD of the NN	|	CD NN
	$NP \rightarrow$	all $DT\backslash NP$	|	all of the $DT\backslash NP$
Topicalization	$S \rightarrow$	NP, VP .	|	VP, NP .
Passivization	$SBAR \rightarrow$	that NP had VBN	|	which was VBN by NP
Light verbs	$VP \rightarrow$	take action $ADVP$	|	to act $ADVP$
	$VP \rightarrow$	TO take a decision PP	|	TO decide PP

Table 1: Examples of meaning-preserving transformations and syntactic paraphrases that our system extracts to capture them.

impose constraints on the nonterminals to ensure the arguments' well-formedness.

The paraphrases implementing the *possessive rule* and the *dative shift* shown in Table 1 are a good examples of this: the two noun-phrase arguments to the expressions are abstracted to nonterminals while the rules' lexicalization provides an appropriate frame of evidence for the transform. This is important for a good representation of the dative shift, which is a reordering transformation that fully applies to certain di-transitive verbs while other verbs are uncommon in one of the forms:

give *decontamination equipment* to *Japan*
give *Japan decontamination equipment*

provide *decontamination equipment* to *Japan*
? provide *Japan decontamination equipment*

Note how our system extracts a dative shift rule for *to give* and a rule that both shifts and substitutes a more appropriate verb when paraphrasing *to provide*.

The use of syntactic nonterminals in our paraphrase rules to capture complex transforms also makes it possible to impose constraints on their application. For comparison, as Madnani et al. (2007) do not impose any constraints on how the nonterminal X can be realized, their equivalent of the *topicalization* rule would massively overgeneralize:

$$S \rightarrow \quad X_1, X_2 \, . \quad | \quad X_2, X_1 \, .$$

Additional examples of transforms our use of syntax allows us to capture are the *adverbial phrase shift*, the *reduction of a relative clause*, as well as other phenomena listed in Table 1.

Unsurprisingly, syntactic information alone is not sufficient to capture all transformations. For instance it is hard to extract generic paraphrases for all instances of *passivization*, since our syntactic model currently has no means of representing the morphological changes that the verb undergoes:

the reactor *leaks* radiation
radiation *is leaking* from the reactor .

Still, for cases where the verb's morphology does not change, we manage to learn a rule:

the radiation that the reactor had *leaked*
the radiation which *leaked* from the reactor .

Another example of a deficiency in our synchronous grammar models are *light verb* constructs such as:

to take a *walk*
to *walk* .

Here, a noun is transformed into the corresponding verb – something our synchronous syntactic CFG approach is not able to capture except through memorization.

Overall our survey shows that we are able to extract appropriately generic representations for a surprising number of paraphrastic transformations, far exceeding the expressiveness of previous approaches to paraphrase extraction from bilingual parallel corpora.

6 Text-to-Text Applications

The core of many text-to-text generation tasks is sentential paraphrasing, augmented with specific constraints or goals. Since our model borrows much of its machinery from statistical machine translation – a sentential rewriting problem itself – it is straightforward to use our paraphrase grammars to generate new sentences using SMT's decoding and parameter optimization techniques. Our framework can be adapted to many different text-to-text generation tasks. These could include text simplification, sentence compression, poetry generation, query expansion, transforming declarative sentences into questions, deriving hypotheses for textual entailment, etc. Each individual text-to-text application requires that our framework be adapted in several ways, by specifying:

- A mechanism for extracting synchronous grammar rules (in this paper we argue that pivot-based paraphrasing is widely applicable).
- An appropriate set of rule-level features that capture information pertinent to the task (e.g. whether a rule simplifies a phrase).
- An appropriate 'objective function' that scores the output of the model, i.e. a task-specific equivalent to the BLEU metric in SMT.
- A development set with examples of the sentential transformations that we are modeling.
- Optionally, a way of injecting task-specific rules that were not extracted automatically.

In the remainder of this section, we illustrate how our bilingually extracted paraphrases can be adapted to perform sentence compression, which is the task of reducing the length of sentence while preserving its core meaning. Most previous approaches to sentence compression focused only on the deletion of a subset of words from the sentence (Knight and Marcu, 2002). Our approach follows Cohn and Lapata (2008), who expand the task to include substitutions, insertions and reorderings that are automatically learned from parallel texts.

6.1 Feature Design

In Section 4 we discussed phrasal probabilities. While these help quantify how good a paraphrase is in general, they do not make any statement on task-specific things such as the change in language complexity or text length. To make this information available to the decoder, we enhance our paraphrases with four compression-targeted features. We add the count features: c_{src} and c_{tgt}, indicating the number of words on either side of the rule as well as two difference features: $c_{dcount} = c_{tgt} - c_{src}$ and the analoguosly computed difference in the average word length in characters: c_{davg}.

6.2 Development Data

To tune the parameters of our paraphrase system for sentence compression, we need an appropriate corpus of reference compressions. Since our model is designed to compress by paraphrasing rather than deletion, the commonly used deletion-based compression data sets like the Ziff Davis corpus are not suitable. We have thus created a corpus of compression paraphrases. Beginning with 9570 tuples of parallel English-English sentences obtained from multiple reference translations for machine translation evaluation, we construct a parallel compression corpus by selecting the longest reference in each tuple as the source sentence and the shortest reference as the target sentence. We further retain only those sentence pairs where the compression rate cr is: $0.5 < cr \leq 0.8$. From these, we then randomly select 936 sentences for the development set, as well as 560 sentences for a test set that we use to gauge the performance of our system.

6.3 Objective Function

Given the nature of an SMT-based paraphrasing system, the most straightforward choice for parameter optimization is to optimize for BLEU over a set of paraphrases, for instance parallel English reference translations for a machine translation task (Madnani et al., 2007). Doing so naively, however, will result in a trivial paraphrasing system heavily biased towards producing identity "paraphrases". This is obviously not what we are looking for.

As an example of a suitable objective function for paraphrase training, we propose MRCBLEU, an ex-

tension of BLEU which takes into account the input i and penalizes the system when the changes in the output o are below a threshold λ:

$$\text{MRCBLEU}_\lambda(i,o) = \begin{cases} \frac{d_{lev}(i,o)}{\lambda} \cdot \text{BLEU}(o) & \text{if } d_{lev}(i,o) < \lambda \\ \text{BLEU}(o) & \text{otherwise} \end{cases},$$

where $d_{lev}(i,o)$ is the Levenshtein edit rate. It is straightforward to find similar adaptations for other tasks. For text simplification, for instance, the penalty term can include a readability metric. For text compression, we can intoduce an analoguous penalty if the output fails to produce the desired compression rate.

6.4 Grammar Augmentations

As we discussed in Section 5, the paraphrase grammar we induce is capable of representing a wide variety of transformations. However, the formalism and extraction method are not explicitely geared towards a compression application. For instance, the synchronous nature of our grammar does not allow us to perform deletions of constituents as done by Cohn and Lapata (2007)'s tree transducers. A possible way to extend the grammar's capabilities towards the requirements of a given task is by injecting additional rules designed to capture appropriate operations.

For the compression task, this can include adding rules that allow generic deletions of target-side nonterminals:

$$JJ \rightarrow JJ \mid \varepsilon$$

This, however, renders the grammar asynchronous and requires appropriate adjustments in the decoding process. Alternatively, we can generate rules that specifically delete particular adjectives from the corpus:

$$JJ \rightarrow \text{superfluous} \mid \varepsilon\ .$$

In our setup, it is more straightforward to adopt the latter approach.

6.5 Experimental Setup

We extracted a paraphrase grammar from the French-English Europarl corpus (v5). The bitext was aligned using the Berkeley aligner and the English side was parsed with the Berkeley parser. We obtained the initial translation grammar using the SAMT toolkit.

Grammar	# Rules
total	42,353,318
w/o identity	23,641,016
w/o complex constituents	6,439,923
w/o complex const. & identity	5,097,250

Table 2: Number and distribution of rules in our paraphrase grammar. Note the significant number of identity paraphrases and rules with complex nonterminal labels.

The grammars we extract tend to become extremely large. To keep their size manageable, we only consider translation rules that have been seen more than 3 times and whose translation probability exceeds 10^{-4} for pivot recombination. Additionally, we only retain the top 25 most likely paraphrases of each phrase, ranked by a uniformly weighted combination of phrasal and lexical paraphrase probabilities.

We tuned the parameters of our model via minimum error rate training using the Z-MERT toolkit (Zaidan, 2009). For decoding we used the Joshua package (Li et al., 2009). The language model used in both our paraphraser and the Clarke and Lapata (2008) baseline system is a Kneser-Ney discounted 5-gram model estimated on the Gigaword corpus using the SRILM toolkit (Stolcke, 2002).

With the publication of this paper, we will release our software for paraphrase grammar extraction, along with the grammars and our development and test data sets.

6.6 Evaluation

To assess the output quality of the resulting sentence compression system, we compare it to two state-of-the-art sentence compression systems. Specifically, we compare against an implementation of Clarke and Lapata (2008)'s compression model which uses a series of constraints in an integer linear programming (ILP) solver, and Cohn and Lapata (2007)'s tree transducer toolkit (T3) which learns a synchronous tree substitution grammar (STSG) from paired monolingual sentences. Unlike SCFGs, the STSG formalism allows changes to the tree topology. Cohn and Lapata argue that this is it a natural fit for sentence compression, since deletions intro-

Source	he also expected that he would have a role in the future at the level of the islamic movement across the palestinian territories , even if he was not lucky enough to win in the elections .
Reference	he expects to have a future role in the islamic movement in the palestinian territories if he is not successful in the elections .
Paraphrase	he also expected that he would have a role in the future of the islamic movement in the palestinian territories , although he was not lucky enough to win elections .
ILP	he also expected that he would have a role at the level of the islamic movement , even if he was not lucky enough to win in the elections .
T3	he also expected that he have a role .

Table 3: Example compressions produced by the three different systems for an input sentence from our test data.

duce structural mismatches. We trained the T3 software[2] on the 936 $\langle \text{full}, \text{compressed} \rangle$ sentence pairs that comprise our development set. This is equivalent in size to the training corpora that Cohn and Lapata (2007) used (their training corpora ranged from 882–1020 sentence pairs), and has the advantage of being in-domain with respect to our test set. Both these systems reported results outperforming previous systems such as McDonald (2006).

We solicit human judgments of the compressions along two five-point scales: grammaticality and meaning. Judges are instructed to decide how much the meaning from a reference translation is retained in the compressed sentence, with a score of 5 indicating that all of the important information is present, and 1 being that the compression does not retain any of the original meaning. Similarly, a grammar score of 5 indicates perfect grammaticality, and a grammar score of 1 is assigned to sentences that are entirely ungrammatical.

For a fair comparison, we tie the compression rate of the ILP system to the rate achieved by our own system.[3] This is done on the sentence level, leaving the ILP system a slack window of 1 token to not overly constrain its decisions. We were unable to set the compression rate for the T3 system; the resulting unfairly short output did not fare well in the evaluation. Additionally we remove all sentences from the evaluation on which either of the systems failed to produce an output sentence, leaving us with 303 sentence-compresssion pairs.

Table 3 shows an example sentence drawn from our test set and the compressions produced by the different systems. We see that both the paraphrase and ILP systems produce good quality results, with the paraphrase system retaining the meaning of the source sentence more accurately.

	CR	Meaning	Grammar
Reference	0.73	4.26	4.35
Paraphrase	0.81	**3.65**	3.38
ILP	0.79	3.45	3.54
T3	0.50	2.05	2.38
Random	0.50	1.94	1.57

Table 4: Results of the human evaluation: compression rate (CR), meaning and grammaticality scores.

The results for the human evaluation are shown in Table 4. We observe that, at similar compression rates, our paraphrase-based system significantly outperforms the ILP system in meaning retention[4], while the better grammaticality score of the ILP system is not statistically significant ($p < 0.088$). This means that our framework for text-to-text generation is performing as well as or better than specifically tailored state-of-the-art methods.

7 Conclusion

In this work we introduced a method to learn syntactically informed paraphrases from bilingual parallel texts. We discussed the expressive power and limitations of our formalism and outlined straightforward adaptation strategies for applications in text-to-text generation. We demonstrated our paraphrasing system, adapted to do sentence compression, to achieve results outperforming state-of-the-art compression systems with only minimal effort.

[2] www.dcs.shef.ac.uk/people/T.Cohn/t3/

[3] We have observed that evaluation quality correlates linearly with compression rate, and therefore the community-accepted practice of not comparing based on a closely tied compression rate is potentially subject to erroneous interpretation.

[4] This improvement is significant at $p < 0.0001$.

References

Alfred V. Aho and Jeffrey D. Ullman. 1972. *The Theory of Parsing, Translation, and Compiling*. Prentice Hall.

Peter G. Anick and Suresh Tipirneni. 1999. The paraphrase search assistant: terminological feedback for iterative information seeking. In *Proceedings of SIGIR*.

Colin Bannard and Chris Callison-Burch. 2005. Paraphrasing with bilingual parallel corpora. In *Proceedings of ACL*.

Regina Barzilay and Noemie Elhadad. 2003. Sentence alignment for monolingual comparable corpora. In *Proceedings of EMNLP*.

Regina Barzilay and Lillian Lee. 2003. Learning to paraphrase: An unsupervised approach using multiple-sequence alignment. In *Proceedings of HLT/NAACL*.

Regina Barzilay and Kathleen McKeown. 2001. Extracting paraphrases from a parallel corpus. In *Proceedings of ACL*.

Regina Barzilay, Kathleen R. McKeown, and Michael Elhadad. 1999. Information fusion in the context of multi-document summarization. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 550–557, Morristown, NJ, USA. Association for Computational Linguistics.

Regina Barzilay. 2003. *Information Fusion for Mutlidocument Summarization: Paraphrasing and Generation*. Ph.D. thesis, Columbia University, New York.

Rahul Bhagat and Deepak Ravichandran. 2008. Large scale acquisition of paraphrases for learning surface patterns. In *Proceedings of ACL/HLT*.

Peter Brown, John Cocke, Stephen Della Pietra, Vincent Della Pietra, Frederick Jelinek, Robert Mercer, and Paul Poossin. 1990. A statistical approach to language translation. *Computational Linguistics*, 16(2), June.

Peter Brown, Stephen Della Pietra, Vincent Della Pietra, and Robert Mercer. 1993. The mathematics of machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, June.

Chris Callison-Burch and Mark Dredze. 2010. Creating speech and language data with amazon's mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, NAACL-2010-address.

Chris Callison-Burch. 2008. Syntactic constraints on paraphrases extracted from parallel corpora. In *Proceedings of EMNLP*.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 263–270, Morristown, NJ, USA. Association for Computational Linguistics.

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228.

James Clarke and Mirella Lapata. 2008. Global inference for sentence compression: An integer linear programming approach. *Journal of Artificial Intelligence Research*, 31:273–381.

Trevor Cohn and Mirella Lapata. 2007. Large margin synchronous generation and its application to sentence compression. In *Proceedings of EMNLP-CoLing*.

Trevor Cohn and Mirella Lapata. 2008. Sentence compression beyond word deletion. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*.

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

P. W. Culicover. 1968. Paraphrase generation and information retrieval from stored text. *Mechanical Translation and Computational Linguistics*, 11(1-2):78–88.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th International Conference on Computational Linguistics*.

Mark Dras. 1997. Representing paraphrases using synchronous tree adjoining grammars. In *Proceedings of ACL*.

Mark Dras. 1999. *Tree Adjoining Grammar and the Reluctant Paraphrasing of Text*. Ph.D. thesis, Macquarie University, Australia.

Chris Dyer, Adam Lopez, Juri Ganitkevitch, Jonathan Weese, Ferhan Ture, Phil Blunsom, Hendra Setiawan, Vladimir Eidelman, and Philip Resnik. 2010. cdec: A decoder, alignment, and learning framework for finite-state and context-free translation models. In *Proceedings of ACL*.

William Gale and Kenneth Church. 1993. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1):75–90.

Michel Galley, Mark Hopkins, Kevin Knight, and Daniel Marcu. 2004. What's in a translation rule? In *Proceedings of HLT/NAACL*.

Liang Huang and David Chiang. 2007. Forest rescoring: Faster decoding with integrated language models. In *Proceedings of ACL*.

David Kauchak and Regina Barzilay. 2006. Paraphrasing for automatic evaluation. In *Proceedings of EMNLP*.

Kevin Knight and Daniel Marcu. 2002. Summarization beyond sentence extraction: A probabilistic approach to sentence compression. *Artificial Intelligence*, 139:91–107.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of HLT/NAACL*.

Raymond Kozlowski, Kathleen McCoy, and K. Vijay-Shanker. 2003. Generation of single-sentence paraphrases from predicate/argument structure using lexico-grammatical resources. In *Workshop On Paraphrasing*.

Irene Langkilde and Kevin Knight. 1998. The practical value of n-grams in generation. In *Workshop On Natural Language Generation*, Ontario, Canada.

Zhifei Li, Chris Callison-Burch, Chris Dyer, Juri Ganitkevitch, Sanjeev Khudanpur, Lane Schwartz, Wren Thornton, Jonathan Weese, and Omar Zaidan. 2009. Joshua: An open source toolkit for parsing-based machine translation. In *Proceedings of the Fourth Workshop on Statistical Machine Translation (WMT09)*.

Dekang Lin and Patrick Pantel. 2001. Discovery of inference rules from text. *Natural Language Engineering*, 7(3):343–360.

Yang Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-string alignment templates for statistical machine translation. In *Proceedings of the ACL/Coling*.

Nitin Madnani and Bonnie Dorr. 2010. Generating phrasal and sentential paraphrases: A survey of data-driven methods. *Computational Linguistics*, 36(3):341–388.

Nitin Madnani, Necip Fazil Ayan, Philip Resnik, and Bonnie Dorr. 2007. Using paraphrases for parameter tuning in statistical machine translation. In *Proceedings of the ACL Workshop on Statistical Machine Translation*.

Ryan McDonald. 2006. Discriminative sentence compression with soft syntactic evidence. In *Proceedings of EACL*.

Kathleen R. McKeown. 1979. Paraphrasing using given and new information in a question-answer system. In *Proceedings of ACL*.

Dan Melamed. 2004. Statistical machine translation by parsing. In *Proceedings of ACL*.

Marie W. Meteer and Varda Shaked. 1988. Strategies for effective paraphrasing. In *Proceedings of COLING*.

Kazunori Muraki. 1982. On a semantic model for multilingual paraphrasing. In *Proceedings of COLING*.

Franz Josef Och. 2003a. Minimum error rate training for statistical machine translation. In *Proceedings of ACL*.

Franz Josef Och. 2003b. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 160–167, Morristown, NJ, USA. Association for Computational Linguistics.

Karolina Owczarzak, Declan Groves, Josef Van Genabith, and Andy Way. 2006. Contextual bitext-derived paraphrases in automatic MT evaluation. In *Proceedings of the NAACL Workshop on Statistical Machine Translation*.

Bo Pang, Kevin Knight, and Daniel Marcu. 2003. Syntax-based alignment of multiple translations: Extracting paraphrases and generating new sentences. In *Proceedings of HLT/NAACL*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL*.

Chris Quirk, Chris Brockett, and William Dolan. 2004. Monolingual machine translation for paraphrase generation. In *Proceedings of EMNLP*.

Chris Quirk, Arul Menezes, and Colin Cherry. 2005. Dependency treelet translation: Syntactically informed phrasal smt. In *Proceedings of ACL*.

Deepak Ravichandran and Eduard Hovy. 2002. Learning sufrace text patterns for a question answering system. In *Proceedings of ACL*.

Stefan Riezler, Alexander Vasserman, Ioannis Tsochantaridis, Vibhu Mittal, and Yi Liu. 2007. Statistical machine translation for query expansion in answer retrieval. In *Proceedings of ACL*.

Hadar Shemtov. 1996. Generation of paraphrases from ambiguous logical forms. In *Proceedings of COLING*.

Stuart Shieber and Yves Schabes. 1990. Generation and synchronous tree-adjoining grammars. In *Workshop On Natural Language Generation*.

Matthew Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. 2010. Ter-plus: paraphrase, semantic, and alignment enhancements to translation edit rate. *Machine Translation*, 23(2-3):117–127.

Mark Steedman. 1999. Alternating quantifier scope in CCG. In *Proceedings of ACL*.

Andreas Stolcke. 2002. SRILM - an extensible language modeling toolkit.

Idan Szpektor, Hristo Tanev, Ido Dagan, and Bonaventura Coppola. 2004. Scaling web-based acquisition of entailment relations. In *Proceedings of EMNLP*, Proceedings of EMNLP.

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3).

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation model. In *Proceedings of ACL*.

Kazuhide Yamamoto. 2002. Machine translation by interaction between paraphraser and transfer. In *Proceedings of COLING*.

Omar F. Zaidan. 2009. 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.

Shiqi Zhao, Cheng Niu, Ming Zhou, Ting Liu, and Sheng Li. 2008a. Combining multiple resources to improve SMT-based paraphrasing model. In *Proceedings of ACL/HLT*.

Shiqi Zhao, Haifeng Wang, Ting Liu, and Sheng Li. 2008b. Pivot approach for extracting paraphrase patterns from bilingual corpora. In *Proceedings of ACL/HLT*.

Shiqi Zhao, Xiang Lan, Ting Liu, and Sheng Li. 2009. Application-driven statistical paraphrase generation. In *Proceedings of ACL*.

Liang Zhou, Chin-Yew Lin, Dragos Stefan Munteanu, and Eduard Hovy. 2006. Paraeval: Using paraphrases to evaluate summaries automatically. In *Proceedings of HLT/NAACL*.

Andreas Zollmann and Ashish Venugopal. 2006. Syntax augmented machine translation via chart parsing. In *Proceedings of the NAACL-2006 Workshop on Statistical Machine Translation (WMT06)*.