Paraphrasing Headlines by Machine Translation

Sentential paraphrase acquisition and generation using Google News

Sander Wubben, Antal van den Bosch and Emiel Krahmer

Tilburg Centre for Cognition and Communication
Tilburg University

Abstract

In this paper we investigate the automatic collection, generation and evaluation of sentential paraphrases. Valuable sources of paraphrases are news article headlines; they tend to describe the same event in various different ways, and can easily be obtained from the web. We describe a method for generating paraphrases by using a large aligned monolingual corpus of news headlines acquired automatically from Google News and a standard Phrase-Based Machine Translation (PBMT) framework. The output of this system is compared to a word substitution baseline. Human judges prefer the PBMT paraphrasing system over the word substitution system. We compare human judgements to automatic judgement measures and demonstrate that the BLEU metric correlates well with human judgements provided that the generated paraphrase is sufficiently different from the source sentence.

1 Introduction

Text-to-text generation is an increasingly studied subfield in natural language processing. In contrast with the typical natural language generation paradigm of converting concepts to text, in text-to-text generation a source text is converted into a target text that approximates the meaning of the source text. Text-to-text generation extends to such varied tasks as summarization (**?**), question-answering (**?**), Machine Translation, and paraphrase generation.

For text-to-text generation it is important to know which words and phrases are semantically close or exchangable in which contexts. While there are various resources available that capture such knowledge at the word level (e.g., synonymic knowledge in WordNet), this kind of information is much harder to get by at the phrase or even at the sentence level. The paraphrasing task extends from the word level up to the discourse level; a WordNet-like resource at the paraphrase level would be needed to generate paraphrases of new, unseen text. Therefore, paraphrase acquisition can be considered an important technology for producing resources for text-to-text generation. Paraphrase generation has already proven to be valuable for Question Answering (**?**, **?**), Machine Translation (**?**) and the evaluation thereof (**?**, **?**, **?**), but also for text simplification and explanation.

Paraphrase generation is the process of transforming a source sentence into a target sentence in the same language which differs in form from the source sentence, but approximates its meaning. Paraphrasing is often used as a subtask in more complex NLP applications to allow for more variation in text strings presented as input, for example to generate paraphrases of questions that in their original form cannot be answered (**?**, **?**), or to generate paraphrases of sentences that failed to translate (**?**). Paraphrasing has also been used in the evaluation of

Machine Translation system output (**?**, **?**, **?**). Adding certain constraints to paraphrasing allows for additional useful applications. When the constraint is specified that a paraphrase should be shorter than the input text, paraphrasing can be used for sentence compression (**?**, **?**). Another specific task that can be approached this way is text simplification for question answering or subtitle generation (**?**).

In this paper we regard the generation of sentential paraphrases as a monolingual Machine Translation task, where the source and target languages are the same (**?**). However, there are two problems that have to be dealt with to make this approach work, namely obtaining a sufficient amount of examples, and a proper evaluation methodology. As (**?**) argue, automatic evaluation of paraphrasing is problematic. The essence of paraphrasing is to generate a sentence that is structurally different from the source. Automatic evaluation metrics in related fields such as standard multilingual Machine Translation operate on a notion of similarity, while paraphrasing also centers around achieving dissimilarity. Besides the evaluation issue, another problem is that for an data-driven Machine Translation account of paraphrasing to work, a large collection of data is required. In this case, this would have to be pairs of sentences that are paraphrases of each other. So far, paraphrasing data sets of sufficient size have been mostly lacking. The work on paraphrasing has also mainly been focused on phrases as opposed to sentences. We argue that the headlines aggregated by Google News offer an attractive avenue.

2 Data Collection

Currently few resources are available for paraphrasing; one example is the Microsoft Paraphrase Corpus (MSR) (**?**, **?**), which is relatively small: it contains 139,000 aligned paraphrases. In this study we explore the use of a large, automatically acquired aligned paraphrase corpus. Where previous work has focused on aligning news-items at the paragraph and sentence level (**?**), we limit ourselves to aligning headlines of news articles. We think this approach will enable us to harvest reliable training material for paraphrase generation fast and efficiently, without having to worry too much about the problems that arise when trying to align complete news articles. Google News is such a vast resource that we already get a lot of data by looking at headlines alone.

For the development of our system we use data which was obtained in the DAESO-project. This project is an ongoing effort to build a Parallel Monolingual Treebank for Dutch (**?**) and will be made available through the Dutch HLT Agency. Part of the data in the DAESO-corpus consists of headline clusters crawled from Google News in the period April–August 2006. Google News uses clustering algorithms that consider the full text of each news article, as well as other features such as temporal and category cues, to produce sets of articles related topically. The crawler stored the headline and the first 150 characters of the article of each news article crawled from the Google News website. Roughly 13,000 Dutch clusters were retrieved, 450 MB in size. Table **??** shows part of a cluster. It is clear that although clusters deal roughly with one subject, the headlines can represent quite a different perspective on the content of the article. To obtain only paraphrase pairs,

Kamp : Veiligheid grootste probleem in Uruzgan *(Kamp: Security biggest problem in Uruzgan)*
Met gevechtsheli op Afghaanse theevisite *(With attack helicopter on Afghan tea-visit)*
Bevel overgedragen aan Nederlandse commandant *(Command transferred to Dutch commander)*
Nederlandse missie Uruzgan officieel begonnen *(Dutch mission Uruzgan officially started)* Nederlandse opbouwmissie in Afghanistan begint *(Dutch construction mission in Afghanistan begins)* Missie Uruzgan begonnen *(Mission Uruzgan had begun*
Soldaten opbouwmissie Uruzgan keren terug *(Soldiers construction mission return)* Eerste militairen komen terug uit Afghanistan *(First servicemen come back from Afghanistan)* Eerste groep militairen Afghanistan keert terug *(First group of servicemen return from Afghanistan)* Kwartiermakers keren terug uit Uruzgan *(Quartermasters return from Uruzgan)*
Opgelucht onthaal van militairen uit Uruzgan *(Relieved welcome of servicemen from Uruzgan)* Opgelucht onthaal van Uruzgan-gangers *(Relieved welcome of Uruzgan-goers)*

Table 1: Part of a sample headline cluster crawled in August 2006. The original headlines are displayed as they were clustered by the annotators.

the clusters need to be more coherent. To that end, in the DAESO project 865 clusters were manually subdivided into sub-clusters of headlines that show clear semantic overlap. Sub-clustering is no trivial task, however. Some sentences are very clearly paraphrases, but consider for instance the sentences in the example containing 'Afghanistan' or 'Uruzgan'. they can be seen as paraphrases of each other, but then the reader must now that Uruzgan' is a province in Afghanistan where the Dutch mission is stationed. Also, there are numerous headlines that can not be sub-clustered, such as the first three headlines shown in the example.

This annotated data is used to develop a method of automatically obtaining paraphrase pairs from headline clusters. We divide the annotated headline clusters in a development set of 40 clusters, while the remainder is used as test data. The headlines are stemmed using the porter stemmer for Dutch (**?**) Instead of a word overlap measure as used by Barzilay and Elhadad (**?**), we use a modified $TF*IDF$ word score as was suggested by Nelken and Shoeber (**?**). Each sentence is viewed

as a document, and each original cluster as a collection of documents. For each stemmed word i in sentence j, $TF_{i,j}$ is a binary variable indicating if the word occurs in the sentence or not. The $TF * IDF$ score can then be stated as follows:

$$\mathrm{TF.IDF_i} = TF_{i,j} \cdot \log \frac{|D|}{|\{d_j : t_i \in d_j\}|}$$

$|D|$ is the total number of sentences in the cluster and $|\{d_j : t_i \in d_j\}|$ is the number of sentences that contain the term t_i. These scores are used in a vector space representation. The similarity between headlines can be calculated by using a similarity function on the headline vectors, such as Cosine similarity.

2.1 Clustering

Our first approach is to use a clustering algorithm to cluster similar headlines. The original Google News headline clusters are reclustered into finer grained sub-clusters. We use the k-means implementation in the CLUTO[1] software package. The k-means algorithm is an algorithm that assigns k centers to represent the clustering of n points ($k < n$) in a vector space. The total intra-cluster variances is minimized by the function

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2$$

where μ_i is the centroid of all the points $x_j \in S_i$.

The PK1 cluster-stopping algorithm as proposed by Pedersen and Kulkarni (**?**) is used to find the optimal k for each sub-cluster:

$$PK1(k) = \frac{Cr(k) - mean(Cr[1...deltaK])}{std(Cr[1...deltaK])}$$

Here, Cr is a criterion function. As soon as $PK1(k)$ exceeds a threshold, $k-1$ is selected as the optimum number of clusters.

To find the optimal threshold value for cluster stopping, optimization is performed on the development data. Our optimization function is an F-score:

$$F_\beta = \frac{(1+\beta^2) \cdot (\mathrm{precision} \cdot \mathrm{recall})}{(\beta^2 \cdot \mathrm{precision} + \mathrm{recall})}$$

We evaluate the number of aligments between possible paraphrases. For instance, in a cluster of four sentences, $\binom{4}{2} = 6$ alignments can be made. In our case, precision is the number of alignments retrieved from the clusters which are relevant, divided by the total number of retrieved alignments. Recall is the number of relevant retrieved alignments divided by the total number of relevant alignments.

We use an F_β-score with a β of 0.25 as we favor precision above recall. We do not want to optimize on precision alone, because we still want to retrieve a

[1] http://glaros.dtc.umn.edu/gkhome/views/cluto/

Type	Precision	Recall
k-means clustering clusters only	0.91	0.43
k-means clustering all headlines	0.66	0.44
pairwise similarity all headlines	0.76	0.41

Table 2: Precision and Recall for both methods

fair amount of paraphrases and not only the ones that are very similar. Through optimization on our development set, we find an optimal threshold for the PK1 algorithm $th_{pk1} = 1$. For each original cluster, k-means clustering is then performed using the k found by the cluster stopping function. In each newly obtained cluster all headlines can be aligned with each other.

2.2 Pairwise similarity

Our second approach is to directly calculate similarities for each pair of headlines within a cluster. If the similarity exceeds a certain threshold, the pair is accepted as a paraphrase pair. If it is below the threshold, it is rejected. However, as Barzilay and Elhadad (**?**) have pointed out, sentence mapping in this way is only effective to a certain extent. Beyond that point, context is needed. With this in mind, we adopt two thresholds and the Cosine similarity function to calculate the similarity between two sentences:

$$cos(\theta) = \frac{V1 \cdot V2}{\|V1\|\|V2\|}$$

where $V1$ and $V2$ are the vectors of the two sentences being compared. If the similarity is higher than the upper threshold, it is accepted. If it is lower than the lower theshold, it is rejected. In the remaining case of a similarity between the two thresholds, similarity is calculated over the contexts of the two headlines, namely the text snippet that was retrieved with the headline. If this similarity exceeds the upper threshold, it is accepted. Threshold values as found by optimizing on the development data using again an $F_{0.25}$-score, are $Th_{lower} = 0.2$ and $Th_{upper} = 0.5$. An optional final step is to add alignments that are implied by previous alignments. For instance, if headline A is paired with headline B, and headline B is aligned to headline C, headline A can be aligned to C as well. We do not add these alignments, because particularly in large clusters when one wrong alignment is made, this process adds a large amount of incorrect alignments.

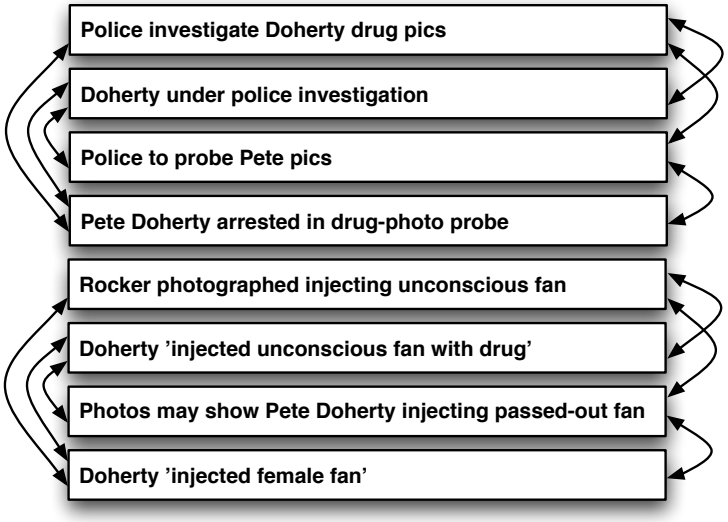

Figure 1: Part of a sample headline cluster, with aligned paraphrases

2.3 Results

The 825 clusters in the test set contain 1,751 sub-clusters in total. In these sub-clusters, there are 6,685 clustered headlines. Another 3,123 headlines remain unclustered. Table **??** displays the paraphrase detection precision and recall of our two approaches. It is clear that k-means clustering performs well when all unclustered headlines are artificially ignored. In the more realistic case when there are also items that cannot be clustered, the pairwise calculation of similarity with a back off strategy of using context performs better when we aim for higher precision.

2.4 Obtaining headline paraphrase pairs

We choose the pairwise similarity approach to extract paraphrasing headline pairs from our much larger extracted English dataset, consisting of roughly 30,000 English headlines that appeared in Google News over the period of April to September 2006, 3 GB in size. Using this method we end up with a collection of 7,400,144 pairwise alignments of 1,025,605 unique headlines[2]. An example of alignments created with this approach is in Figure fig:alignments,

[2]This list of aligned pairs will be made available online.

3 Paraphrase Generation

In our approach we use the collection of automatically obtained aligned headlines to train a paraphrase generation model using a Phrase-Based Machine Translation (PBMT) framework. We compare this approach to a word substitution baseline. The generated paraphrases along with their source headlines are presented to human judges, whose ratings are compared to the BLEU (**?**), METEOR (**?**) and ROUGE (**?**) automatic evaluation metrics.

3.1 Phrase-Based MT

We use the MOSES package to train a PBMT model (**?**). Such a statistical model normally finds a best translation $\tilde{e}$ of a text in language f to a text in language e by combining a translation model $p(f|e)$ with a language model $p(e)$:

$$\tilde{e} = arg \max_{e \in e^*} p(f|e)p(e)$$

GIZA++ is used to perform the word alignments, which are later combined into phrase alignments in the MOSES pipeline (**?**) to build the paraphrase model. We first tokenize our data before training a recaser. We then lowercase all data and use all unique headlines in the training data to train a language model with the SRILM toolkit (**?**). Then we invoke the GIZA++ aligner using the 7M training paraphrase pairs. We run GIZA++ with standard settings and we perform no optimization. Finally, we use the MOSES decoder to generate paraphrases for our test data.

Instead of assigning equal weights to language and translation model, we assign a larger weight of 0.7 to the language model to generate better formed (but more conservative) paraphrases. Because dissimilarity is a factor that is very important for paraphrasing but not implemented in a PBMT model, we perform post-hoc reranking based on dissimilarity. We clearly want our output to be different from out input after all. For each headline in the testset we generate the ten best paraphrases as scored by the decoder and then rerank them according to dissimilarity to the source using the Levenshtein distance measure modified to the word level. this means we look at insertion, deletion and substitution of words. The resulting headlines are recased using the previously trained recaser.

3.2 Word Substitution

The PBMT results are compared with a simple word substitution baseline. For each noun, adjective and verb in the sentence this model takes that word and its Part of Speech tag and retrieves from WordNet its most frequent synonym from the most frequent synset containing the input word. If no relevant alternative is found, the word is left unaltered. We use the Memory Based Tagger (**?**) trained on the Brown corpus to generate the POS-tags. The WordNet::QueryData[3] Perl module is used to query WordNet (**?**). Generated headlines and their source for both systems are given in Table **??**.

[3] http://search.cpan.org/dist/WordNet-QueryData/QueryData.pm

System	Headline
Source	Florida executes notorious serial killer
PBMT	Serial killer executed in Florida
Word Sub.	Florida executes ill-famed series slayer
Source	Dublin evacuates airport due to bomb scare
PBMT	Dublin airport evacuated after bomb threat
Word Sub.	Dublin evacuates airdrome due to bomb panic
Source	N. Korea blasts nuclear sanctions
PBMT	N. Korea nuclear blast of sanctions
Word Sub.	N. Korea blasts atomic sanctions
Source	Israeli raid in Lebanon kills 54
PBMT	Israeli raid kills 54 in Lebanon
Word Sub.	Israeli foray in Lebanon kills 54

Table 3: Examples of generated paraphrases

operation	sentences
single word replacement	80
word deletion or insertion	55
word/phrase reordering	18
phrase replacements	60
sentence rewriting	3

Table 4: Analysis of the generated paraphrases by the PBMT system indicating the number of sentences containing one or more of the specified edit operation.

system	mean	stdev.
PBMT	4.60	0.44
Word Substitution	3.59	0.64

Table 5: Results of human judgements (N = 10)

4 Evaluation

A human judgement study was set up to evaluate the generated paraphrases, and the human judges' ratings are compared to automatic evaluation measures in order to gain more insight in the automatic evaluation of paraphrasing.

4.1 Method

We randomly select 160 headlines from all headlines that meet the following criteria: the headline has to be comprehensible without reading the corresponding news article, both systems have to be able to produce a paraphrase for each headline, and there have to be a minimum of eight paraphrases for each headline. We need these paraphrases as multiple references for our automatic evaluation measures to account for the diversity in real-world paraphrases, as the aligned paraphrased headlines in Figure **??** witness.

The judges are presented with the 160 headlines, along with the paraphrases generated by both systems. The order of the headlines is randomized, and the order of the two paraphrases for each headline is also randomized to prevent a bias towards one of the paraphrases. The judges are asked to rate the paraphrases on a 1 to 7 scale, where 1 means that the paraphrase is very bad and 7 means that the paraphrase is very good. The judges were instructed to base their overall quality judgement on whether the meaning was retained, the paraphrase was grammatical and fluent, and whether the paraphrase was in fact different from the source sentence. Ten judges rated two paraphrases per headline, resulting in a total of 3,200 scores. All judges were blind to the purpose of the evaluation and had no background in paraphrasing research.

System	BLEU	ROUGE-1	ROUGE-2	ROUGE-SU4	METEOR	Lev.dist.	Lev. stdev.
PBMT	0.51	0.76	0.36	0.42	0.71	2.76	1.35
Wordsub.	0.25	0.59	0.22	0.26	0.54	2.67	1.50
Source	0.61	0.80	0.45	0.47	0.77	0	0

Table 6: Automatic evaluation and sentence Levenshtein scores

4.2 Results

The average scores assigned by the human judges to the output of the two systems are displayed in Table **??**. These results show that the judges rated the quality of the PBMT paraphrases significantly higher than those generated by the word substitution system ($t(18) = 4.11, p < .001$).

Results from the automatic measures as well as the Levenshtein distance are listed in Table **??**. We use a Levenshtein distance over tokens instead of characters. First, we observe that both systems perform roughly the same amount of edit operations on a sentence, resulting in a Levenshtein distance over words of 2.76 for the PBMT system and 2.67 for the Word Substitution system. BLEU, METEOR and three typical ROUGE metrics[4] all rate the PBMT system higher than the Word Substitution system. Notice also that the all metrics assign the highest scores to the original sentences, as is to be expected: because every operation we perform is in the same language, the source sentence is also a paraphrase of the reference sentences that we use for scoring our generated headline. If we pick a random sentence from the reference set and score it against the rest of the set, we obtain similar scores. This means that this score can be regarded as an upper bound score for paraphrasing. However, this also shows that these measures cannot be used directly as an automatic evaluation method of paraphrasing, as they assign the highest score to the "paraphrase" in which nothing has changed. The scores observed in Table **??** do indicate that the paraphrases generated by PBMT are less well formed than the original source sentence.

Table **??** shows a breakdown of the paraphrasing operations the PBMT approach has performed. The number indicates the amount of sentences out of the 160 that contain the specific edit operation. Phrase replacements should be interpreted as a replacement operating involving multi-word expressions. Sentence rewriting means that the sentence is fundamentally changed in its entirety, for instance changing from passive to active and vice versa. The first two sentences in Table **??** are examples of this.

There is an overall medium correlation between the BLEU measure and human judgements ($r = 0.41, p < 0.001$). We see a lower correlation between the various ROUGE scores and human judgements, with ROUGE-1 showing the highest correlation ($r = 0.29, p < 0.001$). Between the two lies the METEOR correlation ($r = 0.35, p < 0.001$). However, if we split the data according to Levenshtein distance, we observe that we generally get a higher correlation for all the tested metrics when the Levenshtein distance is higher, as visualized in Figure **??**. At Levenshtein distance 5, the BLEU score achieves a correlation of 0.78 with human judgements, while ROUGE-1 manages to achieve a 0.74 correlation. Beyond edit distance 5, data sparsity occurs.

[4]ROUGE-1, ROUGE-2 and ROUGE-SU4 are also adopted for the DUC 2007 evaluation campaign, `http://www-nlpir.nist.gov/projects/duc/duc2007/tasks.html`

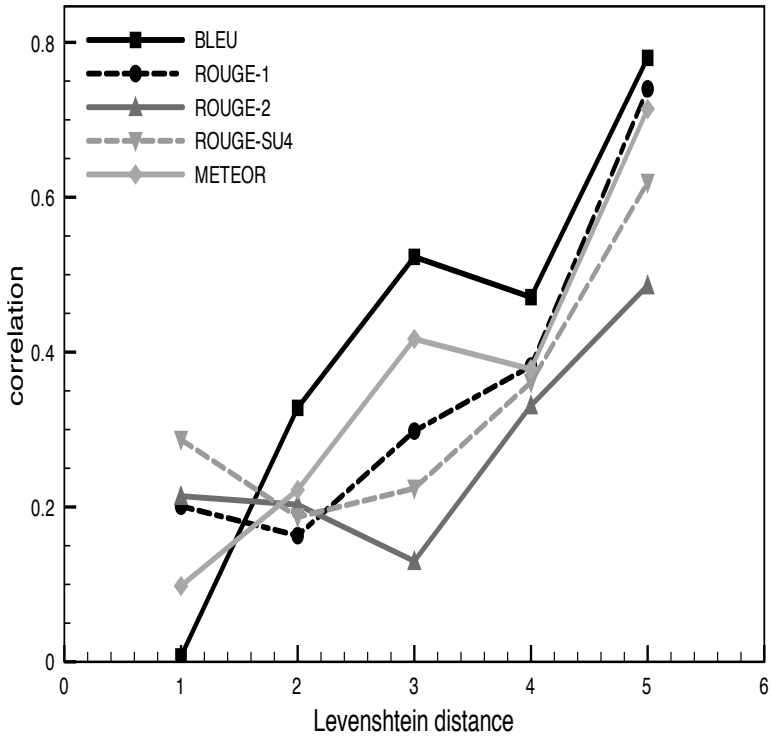

Figure 2: Correlations between human judgements and automatic evaluation metrics for various edit distances

5 Conclusion

In this paper we have shown that with an automatically obtained parallel monolingual corpus with several millions of paired examples, it is possible to develop a sentential paraphrase generation system based on a PBMT framework. We have described a method to align headlines extracted from Google News based on similarity between the two headlines. We have shown that a Cosine similarity function comparing headlines and using a back off strategy to compare context can be used to extract Dutch paraphrase pairs at a precision of 0.76. Although we could aim for a higher precision by assigning higher values to the thresholds, we still want to retain some recall and variation in our paraphrases.

The use of a PBMT framework to exploit this resource of aligned headlines is a feasible strategy; human judges preferred the output of our PBMT system over the output of a simple word substitution system. We have also addressed the

problem of automatic paraphrase evaluation. We measured BLEU, METEOR and ROUGE scores, and observed that these automatic scores correlate with human judgements to some degree, but that the correlation is highly dependent on edit distance. At low edit distances automatic metrics fail to properly assess the quality of paraphrases, whereas at edit distance 5 the correlation of BLEU with human judgements is 0.78, indicating that at higher edit distances these automatic measures can be utilized to rate the quality of the generated paraphrases. From edit distance 2, BLEU correlates best with human judgements, which suggests that Machine Translation evaluation metrics might be better for automatic paraphrase evaluation than summarization metrics.

6 Discussion and future work

The data we used for paraphrasing consists of headlines. Of course headlines use a special kind of language. In headlines articles and most forms of the verb ' to be' are often omitted. Most headlines are in simple present tense and written in telegraphic style and use a lot of abbreviations and metonyms to denote companies and organizations (i.e. 'Wall Street'). This means that the paraphrase patterns we learn are those used in headlines and possibly different from normal conversational language. The advantage of our approach is however that it paraphrases those parts of sentences that it can paraphrase, and leaves those parts that are unknown intact. This is different when we perform standard multilingual translation: if the unknown word is not a proper noun, it can not be left untranslated. It is straightforward to train a language model on in-domain text and use the translation model acquired from the headlines to generate paraphrases for other domains. We are of course also interested in capturing paraphrase patterns existing in other domains, but acquiring parallel paraphrase corpora for different domains is no trivial task.

Our plans for future work are twofold: on the one hand we wish to improve the automatic paraphrase generation process by augmenting the phrase alignment phase, using linguistic information in addition to the statistical models that are employed by GIZA++. We think that due to the monolingual nature of paraphrasing, linguistic information can be used with great effect. In addition, we plan to investigate if our paraphrase generation approach is applicable to comparable fields such as sentence compression and sentence simplification. Our goal is to develop a proper uniform paraphrasing model that is able to take into account different constraints we want to impose on our paraphrases during the decoding process. Such constraints can be dissimilarity for normal paraphrasing, but also simplicity or readability in the case of paraphrasing for sentence simplification and length for sentence compression. These constraints can be taken into account by adding scores for these constraints in addition to scores as provided by the language and translation model. On the topic of automatic evaluation, we aim to define an automatic paraphrase generation assessment score. An automatic paraphrase evaluation measure should be able to recognize that a good paraphrase is a well-formed sentence in the source language, yet at the same it is clearly dissimilar to the source.