

Clustering and Matching Headlines for Automatic Paraphrase Acquisition

Sander Wubben, Antal van den Bosch, Emiel Krahmer, Erwin Marsi

Tilburg centre for Creative Computing

Tilburg University

The Netherlands

{s.wubben, antal.vdnbosch, e.j.krahmer, e.c.marsi}@uvt.nl

Abstract

For developing a data-driven text rewriting algorithm for paraphrasing, it is essential to have a monolingual corpus of aligned paraphrased sentences. News article headlines are a rich source of paraphrases; they tend to describe the same event in various different ways, and can easily be obtained from the web. We compare two methods of aligning headlines to construct such an aligned corpus of paraphrases, one based on clustering, and the other on pairwise similarity-based matching. We show that the latter performs best on the task of aligning paraphrastic headlines.

1 Introduction

In recent years, text-to-text generation has received increasing attention in the field of Natural Language Generation (NLG). In contrast to traditional concept-to-text systems, text-to-text generation systems convert source text to target text, where typically the source and target text share the same meaning to some extent. Applications of text-to-text generation include summarization (Knight and Marcu, 2002), question-answering (Lin and Pantel, 2001), and machine translation.

For text-to-text generation it is important to know which words and phrases are semantically close or exchangeable in which contexts. While there are various resources available that capture such knowledge at the word level (e.g., synset knowledge in WordNet), this kind of information is much harder to get by at the phrase level. 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 (Lin and Pantel, 2001; Riezler et al.,

2007), Machine Translation (Callison-Burch et al., 2006) and the evaluation thereof (Russo-Lassner et al., 2006; Kauchak and Barzilay, 2006; Zhou et al., 2006), but also for text simplification and explanation.

In the study described in this paper, we make an effort to collect Dutch paraphrases from news article headlines in an unsupervised way to be used in future paraphrase generation. News article headlines are abundant on the web, and are already grouped by news aggregators such as Google News. These services collect multiple articles covering the same event. Crawling such news aggregators is an effective way of collecting related articles which can straightforwardly be used for the acquisition of paraphrases (Dolan et al., 2004; Nelken and Shieber, 2006). We use this method to collect a large amount of aligned paraphrases in an automatic fashion.

2 Method

We aim to build a high-quality paraphrase corpus. Considering the fact that this corpus will be the basic resource of a paraphrase generation system, we need it to be as free of errors as possible, because errors will propagate throughout the system. This implies that we focus on obtaining a high precision in the paraphrases collection process. Where previous work has focused on aligning news-items at the paragraph and sentence level (Barzilay and Elhadad, 2003), we choose to focus on aligning the headlines of news articles. We think this approach will enable us to harvest reliable training material for paraphrase generation quickly and efficiently, without having to worry too much about the problems that arise when trying to align complete news articles.

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 (Marsi

Placenta sandwich? No, urban legend!
Tom wants to make movie with Katie
Kate's dad not happy with Tom Cruise
Cruise and Holmes sign for eighteen million Eighteen million for Tom and Katie
Newest mission Tom Cruise not very convincing Latest mission Tom Cruise succeeds less well Tom Cruise barely succeeds with MI:3
Tom Cruise: How weird is he? How weird is Tom Cruise really?
Tom Cruise leaves family Tom Cruise escapes changing diapers

Table 1: Part of a sample headline cluster, with sub-clusters

and Kraahmer, 2007) 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 Netherlands in the period April–August 2006. For each news article, the headline and the first 150 characters of the article were stored. Roughly 13,000 clusters were retrieved. Table 1 shows part of a (translated) 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, the clusters need to be more coherent. To that end 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 last two sentences in the example. They do paraphrase each other to some extent, but their relation can only be understood properly with world knowledge. Also, there are numerous headlines that can not be sub-clustered, such as the first three headlines shown in the example.

We use these annotated clusters as development and test data in developing a method to automatically obtain 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 (Kraaij and Pohlmann, 1994).

Instead of a word overlap measure as used by Barzilay and Elhadad (2003), we use a modified $TF*IDF$ word score as was suggested by Nelken and Shieber (2006). 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 is then:

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

$|D|$ 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 CLUTO¹ 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 (2006) is used to find the optimal k for each sub-cluster:

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

Here, Cr is a criterion function, which measures the ratio of withincluster similarity to betweencluster similarity. 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:

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

¹<http://glaros.dtc.umn.edu/gkhome/views/cluto/>

We evaluate the number of alignments 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 favour precision over recall. We do not want to optimize on precision alone, because we still want to retrieve a 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 to each other.

2.2 Pairwise similarity

Our second approach is to calculate the similarity between pairs of headlines directly. 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 (2003) 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 threshold, 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

Type	Precision	Recall
k -means clustering clusters only	0.91	0.43
k -means clustering all headlines	0.66	0.44
pairwise similarity clusters only	0.93	0.39
pairwise similarity all headlines	0.76	0.41

Table 2: Precision and Recall for both methods

Playstation 3 more expensive than competitor
Playstation 3 will become more expensive than Xbox 360
Sony postpones Blu-Ray movies
Sony postpones coming of blu-ray dvds
Prices Playstation 3 known: from 499 euros
E3 2006: Playstation 3 from 499 euros
Sony PS3 with Blu-Ray for sale from November 11th
PS3 available in Europe from November 17th

Table 3: Examples of correct (above) and incorrect (below) alignments

well. We do not add these alignments, because in particular in large clusters when one wrong alignment is made, this process chains together a large amount of incorrect alignments.

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 2 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. Some examples of correct and incorrect alignments are given in Table 3.

4 Discussion

Using headlines of news articles clustered by Google News, and finding good paraphrases within these clusters is an effective route for obtaining pairs of paraphrased sentences with reasonable precision. We have shown that a cosine similarity function comparing headlines and using a back off strategy to compare context can be used to extract 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 some recall and variation in our paraphrases. Of course the coverage of our method is still somewhat limited: only paraphrases that have some words in common will be extracted. This is not a bad thing: we are particularly interested in extracting paraphrase patterns at the constituent level. These alignments can be made with existing alignment tools such as the GIZA++ toolkit.

We measure the performance of our approaches by comparing to human annotation of sub-clusterings. The human task in itself is hard. For instance, if we look at the incorrect examples in Table 3, the difficulty of distinguishing between paraphrases and non-paraphrases is apparent. In future research we would like to investigate the task of judging paraphrases. The next step we would like to take towards automatic paraphrase generation, is to identify the differences between paraphrases at the constituent level. This task has in fact been performed by human annotators in the DAESO-project. A logical next step would be to learn to align the different constituents on our extracted paraphrases in an unsupervised way.

Acknowledgements

Thanks are due to the Netherlands Organization for Scientific Research (NWO) and to the Dutch HLT Stevin programme. Thanks also to Wauter Bosma for originally mining the headlines from Google News. For more information on DAESO, please visit daeso.uvt.nl.

References

Regina Barzilay and Noemie Elhadad. 2003. Sentence alignment for monolingual comparable corpora. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 25–32.

Chris Callison-Burch, Philipp Koehn, and Miles Osborne. 2006. Improved statistical machine transla-

tion using paraphrases. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 17–24.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: exploiting massively parallel news sources. In *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, page 350.

David Kauchak and Regina Barzilay. 2006. Paraphrasing for automatic evaluation. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 455–462, June.

Kevin Knight and Daniel Marcu. 2002. Summarization beyond sentence extraction: a probabilistic approach to sentence compression. *Artif. Intell.*, 139(1):91–107.

Wessel Kraaij and Rene Pohlmann. 1994. Porters stemming algorithm for dutch. In *Informatiewetenschap 1994: Wetenschappelijke bijdragen aan de derde STINFON Conferentie*, pages 167–180.

Dekang Lin and Patrick Pantel. 2001. Dirt: Discovery of inference rules from text. In *KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 323–328.

Erwin Marsi and Emiel Krahmer. 2007. Annotating a parallel monolingual treebank with semantic similarity relations. In *the Sixth International Workshop on Treebanks and Linguistic Theories (TLT'07)*.

Rani Nelken and Stuart M. Shieber. 2006. Towards robust context-sensitive sentence alignment for monolingual corpora. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL-06)*, 3–7 April.

Ted Pedersen and Anagha Kulkarni. 2006. Automatic cluster stopping with criterion functions and the gap statistic. In *Proceedings of the 2006 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 276–279.

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

Grazia Russo-Lassner, Jimmy Lin, and Philip Resnik. 2006. A paraphrase-based approach to machine translation evaluation. Technical report, University of Maryland, College Park.

Liang Zhou, Chin-Yew Lin, and Eduard Hovy. 2006. Re-evaluating machine translation results with paraphrase support. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 77–84, July.