Sentence Fusion via Dependency Graph Compression

Katja Filippova and **Michael Strube**
EML Research gGmbH
Schloss-Wolfsbrunnenweg 33
69118 Heidelberg, Germany
http://www.eml-research.de/nlp

Abstract

We present a novel unsupervised sentence fusion method which we apply to a corpus of biographies in German. Given a group of related sentences, we align their dependency trees and build a dependency graph. Using integer linear programming we compress this graph to a new tree, which we then linearize. We use GermaNet and Wikipedia for checking semantic compatibility of co-arguments. In an evaluation with human judges our method outperforms the fusion approach of Barzilay & McKeown (2005) with respect to readability.

1 Introduction

Automatic text summarization is a rapidly developing field in computational linguistics. Summarization systems can be classified as either extractive or abstractive ones (Spärck Jones, 1999). To date, most systems are extractive: sentences are selected from one or several documents and then ordered. This method exhibits problems, because input sentences very often overlap and complement each other at the same time. As a result there is a trade-off between *non-redundancy* and *completeness* of the output. Although the need for abstractive approaches has been recognized before (e.g. McKeown et al. (1999)), so far almost all attempts to get closer to abstractive summarization using scalable, statistical techniques have been limited to sentence compression.

The main reason why there is little progress on abstractive summarization is that this task seems to require a conceptual representation of the text which is not yet available (see e.g. Hovy (2003, p.589)). Sentence fusion (Barzilay & McKeown, 2005), where a new sentence is generated from a group of related sentences and where complete semantic and conceptual representation is not required, can be seen as a middle-ground between extractive and abstractive summarization. Our work regards a corpus of biographies in German where multiple documents about the same person should be merged into a single one. An example of a fused sentence (3) with the source sentences (1,2) is given below:

(1) Bohr studierte an der Universität Kopenhagen
Bohr studied at the University Copenhagen
und erlangte dort seine Doktorwürde.
and got there his PhD
'Bohr studied at the University of Copenhagen and got his PhD there'

(2) Nach dem Abitur studierte er Physik und
After the school studied he physics and
Mathematik an der Universität Kopenhagen.
mathematics at the University Copenhagen
'After school he studied physics and mathematics at the University of Copenhagen'

(3) Nach dem Abitur studierte Bohr Physik und
After the school studied Bohr physics and
Mathematik an der Universität Kopenhagen
mathematics at the University Copenhagen
und erlangte dort seine Doktorwürde.
and got there his PhD
'After school Bohr studied physics and mathematics at the University of Copenhagen and got his PhD there'

Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pages 177–185,
Honolulu, October 2008. ©2008 Association for Computational Linguistics

Having both (1) and (2) in a summary would make it redundant. Selecting only one of them would not give all the information from the input. (3), fused from both (1) and (2), conveys the necessary information without being redundant and is more appropriate for a summary.

To this end, we present a novel sentence fusion method based on dependency structure alignment and semantically and syntactically informed phrase aggregation and pruning. We address the problem in an unsupervised manner and use integer linear programming (ILP) to find a globally optimal solution. We argue that our method has three important advantages compared to existing methods. First, we address the grammaticality issue empirically by means of knowledge obtained from an automatically parsed corpus. We do not require such resources as subcategorization lexicons or hand-crafted rules, but decide to retain a dependency based on its syntactic importance score. The second point concerns integrating semantics. Being definitely important, *”this source of information remains relatively unused in work on aggregation*[1] *within NLG”* (Reiter & Dale, 2000, p.141). To our knowledge, in the text-to-text generation field, we are the first to use semantic information not only for alignment but also for aggregation in that we check coarguments’ compatibility. Apart from that, our method is not limited to sentence fusion and can be easily applied to sentence compression. In Filippova & Strube (2008) we compress English sentences with the same approach and achieve state-of-the-art performance.

The paper is organized as follows: Section 2 gives an overview of related work and Section 3 presents our data. Section 4 introduces our method and Section 5 describes the experiments and discusses the results of the evaluation. The conclusions follow in the final section.

2 Related Work

Most studies on text-to-text generation concern sentence compression where the input consists of exactly one sentence (Jing, 2001; Hori & Furui, 2004; Clarke & Lapata, 2008, inter alia). In such setting, redundancy, incompleteness and compatibility issues do not arise. Apart from that, there is no obvious way of how existing sentence compression methods can be adapted to sentence fusion.

Barzilay & McKeown (2005) present a sentence fusion method for multi-document news summarization which crucially relies on the assumption that information appearing in many sources is important. Consequently, their method produces an intersection of input sentences by, first, finding the centroid of the input, second, augmenting it with information from other sentences and, finally, pruning a predefined set of constituents (e.g. PPs). The resulting structure is not necessarily a tree and allows for extraction of several trees, each of which can be linearized in many ways.

Marsi & Krahmer (2005) extend the approach of Barzilay & McKeown to do not only *intersection* but also *union* fusion. Like Barzilay & McKeown (2005), they find the best linearization with a language model which, as they point out, often produces inadequate rankings being unable to deal with word order, agreement and subcategorization constraints. In our work we aim at producing a valid dependency tree structure so that most grammaticality issues are resolved *before* the linearization stage.

Wan et al. (2007) introduce a global revision method of how a novel sentence can be generated from a set of input words. They formulate the problem as a search for a maximum spanning tree which is incrementally constructed by connecting words or phrases with dependency relations. The grammaticality issue is addressed by a number of hard constraints. As Wan et al. point out, one of the problems with their method is that the output built up from dependencies found in a corpus might have a meaning different from the intended one. Since we build our trees from the input dependencies, this problem does not arise with our method. Apart from that, in our opinion, the optimization formulation we adopt is more appropriate as it allows to integrate many constraints without complex rescoring rules.

3 Data

The comparable corpus we work with is a collection of about 400 biographies in German gathered from

[1] We follow Barzilay & McKeown (2005) and refer to aggregation within text-to-text generation as sentence fusion.

the Internet[2]. These biographies describe 140 different people, and the number of articles for one person ranges from 2 to 4, being 3 on average. Despite obvious similarities between articles about one person, neither identical content nor identical ordering of information can be expected.

Fully automatic preprocessing in our system comprises the following steps: sentence boundaries are identified with a Perl CPAN module[3]. Then the sentences are split into tokens and the TnT tagger (Brants, 2000) and the TreeTagger (Schmid, 1997) are used for tagging and lemmatization respectively. Finally, the biographies are parsed with the CDG dependency parser (Foth & Menzel, 2006). We also identify references to the biographee (pronominal as well as proper names) and temporal expressions (absolute and relative) with a few rules.

4 Our Method

Groups of related sentences serve as input to a sentence fusion system and thus need to be identified first (4.1). Then the dependency trees of the sentences are modified (4.2) and aligned (4.3). Syntactic importance (4.4) and word informativeness (4.5) scores are used to extract a new dependency tree from a graph of aligned trees (4.6). Finally, the tree is linearized (4.7).

4.1 Sentence Alignment

Sentence alignment for comparable corpora requires methods different from those used in machine translation for parallel corpora. For example, given two biographies of a person, one of them may follow the timeline from birth to death whereas the other may group events thematically or tell only about the scientific contribution of the person. Thus one cannot assume that the sentence order or the content is the same in two biographies. Shallow methods like word or bigram overlap, (weighted) cosine or Jaccard similarity are appealing as they are cheap and robust. In particular, Nelken & Schieber (2006) demonstrate the efficacy of a sentence-based *tf*idf* score when applied to comparable corpora. Following them, we define the similarity of two sentences *sim*(s_1, s_2) as

$$\frac{S_1 \cdot S_2}{|S_1| \cdot |S_2|} = \frac{\sum_t w_{S_1}(t) \cdot w_{S_2}(t)}{\sqrt{\sum_t w^2_{S_1}(t) \sum_t w^2_{S_2}(t)}} \qquad (1)$$

where S is the set of all lemmas but stop-words from s, and $w_S(t)$ is the weight of the term t:

$$w_S(t) = S(t)\frac{1}{N_t} \qquad (2)$$

where $S(t)$ is the indicator function of S, N_t is the number of sentences in the biographies of one person which contain t. We enhance the similarity measure by looking up synonymy in GermaNet (Lemnitzer & Kunze, 2002).

We discard identical or nearly identical sentences ($sim(s_1, s_2) > 0.8$) and greedily build sentence clusters using a hierarchical groupwise-average technique. As a result, one sentence may belong to one cluster at most. These sentence clusters serve as input to the fusion algorithm.

4.2 Dependency Tree Modification

We apply a set of transformations to a dependency tree to emphasize its important properties and eliminate unimportant ones. These transformations are necessary for the compression stage. An example of a dependency tree and its modifed version are given in Fig. 1.

PREP preposition nodes (*an, in*) are removed and placed as labels on the edges to the respective nouns;

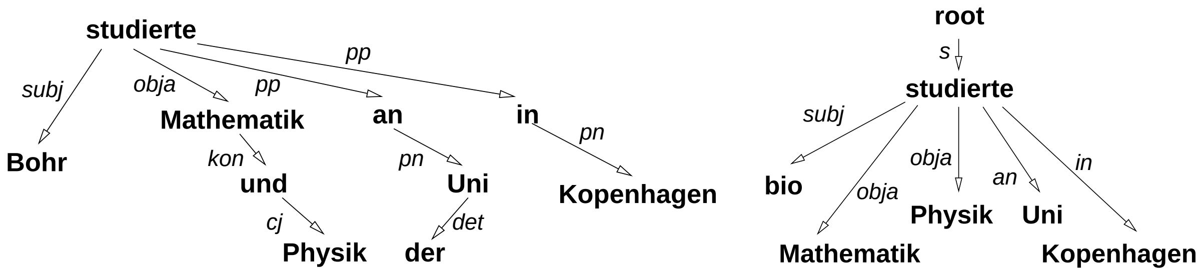
CONJ a chain of conjuncts (*Mathematik und Physik*) is split and each node is attached to the parent node (*studierte*) provided they are not verbs;

APP a chain of words analyzed as appositions by CDG (*Niels Bohr*) is collapsed into one node;

FUNC function words like determiners (*der*), auxiliary verbs or negative particles are removed from the tree and memorized with their lexical heads (memorizing negative particles preserves negation in the output);

[2] http://de.wikipedia.org, http://home.datacomm.ch/biografien, http://biographie.net/de, http://www.weltchronik.de/ws/bio/main.htm, http://www.brockhaus-suche.de/suche

[3] http://search.cpan.org/~holsten/Lingua-DE-Sentence-0.07/Sentence.pm



(a) Dependency tree

(b) Modified tree

Figure 1: The dependency tree of the sentence *Bohr studierte Mathematik und Physik an der Uni in Kopenhagen (Bohr studied mathematics and physics at university in Copenhagen)* as produced by the parser (a) and after all transformations applied (b)

ROOT every dependency tree gets an explicit root which is connected to every verb node;

BIO all occurrences of the biographee (*Niels Bohr*) are replaced with the *bio* tag.

4.3 Node Alignment

Once we have a group of two to four strongly related sentences and their transformed dependency trees, we aim at finding the best node alignment. We use a simple, fast and transparent method and align any two words provided that they

1. are content words;
2. have the same part-of-speech;
3. have identical lemmas or are synonyms.

In case of multiple possibilities, which are extremely rare in our data, the choice is made randomly. By merging all aligned nodes we get a dependency graph which consists of all dependencies from the input trees. In case it contains a cycle, one of the alignments from the cycle is eliminated.

We prefer this very simple method to bottom-up ones (Barzilay & McKeown, 2005; Marsi & Krahmer, 2005) for two main reasons. Pursuing local subtree alignments, bottom-up methods may leave identical words unaligned and thus prohibit fusion of complementary information. On the other hand, they may force alignment of two unrelated words if the subtrees they root are largely aligned. Although in some cases it helps discover paraphrases, it considerably increases chances of generating ungrammatical output which we want to avoid at any cost.

4.4 Syntactic Importance Score

Given a dependency graph we want to get a new dependency tree from it. Intuitively, we want to retain obligatory dependencies (e.g. *subject*) while removing less important ones (e.g. *adv*). When deciding on pruning an argument, previous approaches either used a set of hand-crafted rules (e.g. Barzilay & McKeown (2005)), or utilized a subcategorization lexicon (e.g. Jing (2001)). The hand-crafted rules are often too general to ensure a grammatical argument structure for different verbs (e.g. *PPs can be pruned*). Subcategorization lexicons are not readily available for many languages and cover only verbs. E.g. they do not tell that the noun *son* is very often modified by a PP using the preposition *of*, as in *the son of Niels Bohr*, and that the NP without a PP modifier may appear incomplete.

To overcome these problems, we decide on pruning an edge by estimating the conditional probability of its label given its head, $P(l|h)$[4]. For example, $P(subj|studieren)$ – the probability of the label *subject* given the verb *study* – is higher than $P(in|studieren)$, and therefore the subject will be preserved whereas the prepositional label and thus the whole PP can be pruned, if needed. Table 1 presents the probabilities of several labels given that the head is *studieren* and shows that some prepositions are more important than other ones. Note that if we did not apply the PREP modification we would be unable to distinguish between different prepositions and could only calculate $P(pp|studieren)$

[4] The probabilities are calculated from a corpus of approx. 3,000 biographies from Wikipedia which we annotated automatically as described in Section 3.

which would not be very informative.

subj	obja	in	an	nach	mit	zu
0.88	0.74	0.44	0.42	0.09	0.02	0.01

Table 1: Probabilities of *subj, obja(ccusative), in, at, after, with, to* given the verb *studieren* (*study*)

4.5 Word Informativeness Score

We also want to retain informative words in the output tree. There are many ways in which word importance can be defined. Here, we use a formula introduced by Clarke & Lapata (2008) which is a modification of the significance score of Hori & Furui (2004):

$$I(w_i) = \frac{l}{N} \cdot f_i \log \frac{F_A}{F_i} \tag{3}$$

w_i is the topic word (either noun or verb), f_i is the frequency of w_i in the aligned biographies, F_i is the frequency of w_i in the corpus, and F_A is the sum of frequencies of all topic words in the corpus. l is the number of clause nodes above w and N is the maximum level of embedding of the sentence which w belongs to. By defining word importance differently, e.g. as relatedness of a word to the topic, we could apply our method to topic-based summarization (Krahmer et al., 2008).

4.6 New Sentence Generation

We formulate the task of getting a tree from a dependency graph as an optimization problem and solve it with ILP[5]. In order to decide which edges of the graph to remove, for each directed dependency edge from head h to word w we introduce a binary variable $x^l_{h,w}$, where *l* stands for the label of the edge:

$$x^l_{h,w} = \begin{cases} 1 & \text{if the dependency is preserved} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

The goal is to find a subtree of the graph which gets the highest score of the objective function (5) to which both the probability of dependencies ($P(l|h)$) and the importance of dependent words ($I(w)$) contribute:

$$f(X) = \sum_x x^l_{h,w} \cdot P(l|h) \cdot I(w) \tag{5}$$

The objective function is subject to four types of constraints presented below (W stands for the set of graph nodes minus root, i.e. the set of words).

STRUCTURAL constraints allow to get a tree from the graph: (6) ensures that each word has one head at most. (7) ensures connectivity in the tree. (8) is optional and restricts the size of the resulting tree to α words ($\alpha = \min(0.\bar{6} \cdot |W|, 10)$).

$$\forall w \in W, \sum_{h,l} x^l_{h,w} \leq 1 \tag{6}$$

$$\forall w \in W, \sum_{h,l} x^l_{h,w} - \frac{1}{|W|} \sum_{u,l} x^l_{w,u} \geq 0 \tag{7}$$

$$\sum_x x^l_{h,w} \leq \alpha \tag{8}$$

SYNTACTIC constraints ensure the syntactic validity of the output tree and explicitly state which arguments should be preserved. We have only one syntactic constraint which guarantees that a subordinating conjunction (*sc*) is preserved (9) if and only if the clause it belongs to serves as a subordinate clause (*sub*) in the output.

$$\forall x^{sc}_{w,u}, \sum_{h,l} x^{sub}_{h,w} - x^{sc}_{w,u} = 0 \tag{9}$$

SEMANTIC constraints restrict coordination to semantically compatible elements. The idea behind these constraints is the following (see Fig. 2). It can be that one sentence says *He studied math* and another one *He studied physics*, so the output may unite the two words under coordination: *He studied math and physics*. But if the input sentences are *He studied physics* and *He studied sciences*, then one should not unite both, because *sciences* is the generalization of *physics*. Neither should one unite two unrelated words: *He studied with pleasure* and *He studied with Bohr* cannot be fused into *He studied with pleasure and Bohr*.

To formalize these intuitions we define two functions *hm(w,u)* and *rel(w,u)*: *hm(w,u)* is a binary function, whereas *rel(w,u)* returns a value from $[0, 1]$. We

[5] We use lp_solve in our implementation http://sourceforge.net/projects/lpsolve.

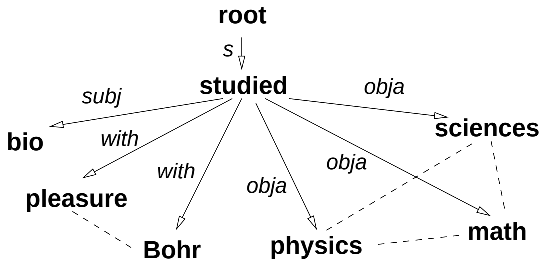


Figure 2: Graph obtained from sentences *He studied sciences with pleasure* and *He studied math and physics with Bohr*

also introduce additional variables $y^l_{w,u}$ (represented by dashed lines in Fig. 2):

$$y^l_{w,u} = \begin{cases} 1 & \text{if } \exists h, l : x^l_{h,w} = 1 \wedge x^l_{h,u} = 1 \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

For two edges sharing a head and having identical labels to be retained we check in GermaNet and in the taxonomy derived from Wikipedia (Kassner et al., 2008) that their dependents are not in the *h*yponymy or *m*eronymy relation (11). We prohibit verb coordination unless it is found in one of the input sentences. If the dependents are nouns, we also check that their semantic relatedness as measured with WikiRelate! (Strube & Ponzetto, 2006) is above a certain threshold (12). We empirically determined the value of $\beta = 0.36$ by calculating an average similarity of coordinated nouns in the corpus.

$$\forall y^l_{w,u}, \texttt{hm}(w,u) \cdot y^l_{w,u} = 0 \quad (11)$$

$$\forall y^l_{w,u}, (\texttt{rel}(w,u) - \beta) \cdot y^l_{w,u} \geq 0 \quad (12)$$

(11) prohibits that *physics* (or *math*) and *sciences* appear together since, according to GermaNet, *physics* (*Physik*) is a hyponym of *science* (*Wissenschaft*). (12) blocks taking both *pleasure* (*Freude*) and *Bohr* because *rel(Freude,Bohr)* $= 0.17$. *math* and *physics* are neither in *ISA*, nor *part-of* relation and are sufficiently related (*rel(Mathematik, Physik)* $= 0.67$) to become conjuncts.

META constraints (equations (13) and (14)) guarantee that $y^l_{w,u} = x^l_{h,w} \times x^l_{h,u}$ i.e. they ensure that the semantic constraints are applied only if both the labels from h to w and from h to u are preserved.

$$\forall y^l_{w,u}, x^l_{h,w} + x^l_{h,u} \geq 2y^l_{w,u} \quad (13)$$

$$\forall y^l_{w,u}, 1 - x^l_{h,w} + 1 - x^l_{h,u} \geq 1 - y^l_{w,u} \quad (14)$$

4.7 Linearization

The "overgenerate-and-rank" approach to statistical surface realization is very common (Langkilde & Knight, 1998). Unfortunately, in its simplest and most popular version, it ignores syntactical constraints and may produce ungrammatical output. For example, an inviolable rule of German grammar states that the finite verb must be in the second position in the main clause. Since it is hard to enforce such rules with an ngram language model, syntax-informed linearization methods have been developed for German (Ringger et al., 2004; Filippova & Strube, 2007). We apply our recent method to order constituents and, using the CMU toolkit (Clarkson & Rosenfeld, 1997), build a trigram language model from Wikipedia (approx. 1GB plain text) to find the best word order within constituents. Some constraints on word order are inferred from the input. Only interclause punctuation is generated.

5 Experiments and Evaluation

We choose Barzilay & McKeown's system as a nontrivial baseline since, to our knowledge, there is no other system which outperforms theirs (Sec. 5.1). It is important for us to evaluate the fusion part of our system, so the input and the linearization module of our method and the baseline are identical. We are also interested in how many errors are due to the linearization module and thus define the readability upper bound (Sec. 5.2). We further present and discuss the experiments (Sec. 5.3 and 5.5).

5.1 Baseline

The algorithm of Barzilay & McKeown (2005) proceeds as follows: Given a group of related sentences, a dependency tree is built for each sentence. These trees are modified so that grammatical features are eliminated from the representation and memorized; noun phrases are flattened to facilitate alignment. A locally optimal pairwise alignment of modified dependency trees is recursively found with WordNet and a paraphrase lexicon. From the alignment costs the centroid of the group is identified. Then this tree is augmented with information from other trees given that it appears in at least half of the sentences from this group. A rule-based pruning module prunes optional constituents, such as PPs or relative clauses. The linearization of the resulting tree (or graph) is done with a trigram language model.

To adapt this system to German, we use the GermaNet API (Gurevych & Niederlich, 2005) instead of WordNet. We do not use a paraphrase lexicon, because there is no comparable corpus of sufficient size available for German. We readjust the alignment parameters of the system to prevent dissimilar nodes from being aligned. The input to the algorithm is generated as described in Sec. 4.1. The linearization is done as described in Sec. 4.7. In cases when there is a graph to linearize, all possible trees covering the maximum number of nodes are extracted from it and linearized. The most probable string is selected as the final output with a language model. For the rest of the reimplementation we follow the algorithm as presented.

5.2 Readability Upper Bound

To find the upper bound on readability, we select one sentence from the input randomly, parse it and linearize the dependency tree as described in Sec. 4.7. This way we obtain a sentence which may differ in form from the input sentences but whose content is identical to one of them.

5.3 Experiments

It is notoriously difficult to evaluate generation and summarization systems as there are many dimensions in which the quality of the output can be assessed. The goal of our present evaluation is in the first place to check whether our method is able to produce sensible output.

We evaluated the three systems (GRAPH-COMPRESSION, BARZILAY & MCKEOWN and READABILITY UB) with 50 native German speakers on 120 fused sentences generated from 40 randomly drawn related sentences groups (3×40). In an online experiment, the participants were asked to read a fused sentence preceded by the input and to rate its readability (*read*) and informativity in respect to the input (*inf*) on a five point scale. The experiment was designed so that every participant rated 40 sentences in total. No participant saw two sentences generated from the same input. The results are presented in Table 2. *len* is an average length in words of the output.

	read	*inf*	*len*
READABILITY UB	4.0	3.5	12.9
BARZILAY & MCKEOWN	3.1	3.0	15.5
GRAPH-COMPRESSION	3.7	3.1	13.0

Table 2: Average readability and informativity on a five point scale, average length in words

5.4 Error Analysis

The main disadvantage of our method, as well as other methods designed to work on syntactic structures, is that it requires a very accurate parser. In some cases, errors in the preprocessing made extracting a valid dependency tree impossible. The poor rating of READABILITY UB also shows that errors of the parser and of the linearization module affect the output considerably.

Although the semantic constraints ruled out many anomalous combinations, the limited coverage of GermaNet and the taxonomy derived from Wikipedia was the reason for some semantic oddities in the sentences generated by our method. For example, it generated phrases like *aus England und Großbritannien* (*from England and Great Britain*). A larger taxonomy would presumably increase the recall of the semantic constraints which proved helpful. Such errors were not observed in the output of the baseline because it does not fuse within NPs.

Both the baseline and our method made subcategorization errors, although these are more common for the baseline which aligns not only synonyms but also verbs which share some arguments. Also, the baseline pruned some PPs necessary for a sentence to be complete. For example, it pruned *an der Atombombe* (*on the atom bomb*) and generated an incomplete sentence *Er arbeitete* (*He worked*). For the baseline, alignment of flattened NPs instead of words caused generating very wordy and redundant sentences when the input parse trees were incorrect. In other cases, our method made mistakes in linearizing constituents because it had to rely on a language model whereas the baseline used unmodified constituents from the input. Absense of intra-clause commas caused a drop in readability in some otherwise grammatical sentences.

5.5 Discussion

A paired t-test revealed significant differences between the readability ratings of the three systems ($p = 0.01$) but found no significant differences between the informativity scores of our system and the baseline. Some participants reported informativity hard to estimate and to be assessable for grammatical sentences only. The higher readability rating of our method supports our claim that the method based on syntactic importance score and global constraints generates more grammatical sentences than existing systems. An important advantage of our method is that it addresses the subcategorization issue directly without shifting the burden of selecting the right arguments to the linearization module. The dependency structure it outputs is a tree and not a graph as it may happen with the method of Barzilay & McKeown (2005). Moreover, our method can distinguish between more and less obligatory arguments. For example, it knows that *at* is more important than *to* for *study* whereas for *go* it is the other way round. Unlike our differentiated approach, the baseline rule states that PPs can generally be pruned.

Since the baseline generates a new sentence by modifying the tree of an input sentence, in some cases it outputs a compression of this sentence. Unlike this, our method is not based on an input tree and generates a new sentence without being biased to any of the input sentences.

Our method can also be applied to non-trivial sentence compression, whereas the baseline and similar methods, such as Marsi & Krahmer (2005), would then boil down to a few very general pruning rules. We tested our method on the English compression corpus[6] and evaluated the compressions automatically the same way as Clarke & Lapata (2008) did. The results (Filippova & Strube, 2008) were as good as or significantly better than the state-of-the-art, depending on the choice of dependency parser.

6 Conclusions

We presented a novel sentence fusion method which formulates the fusion task as an optimization problem. It is unsupervised and finds a globally optimal solution taking semantics, syntax and word informativeness into account. The method does not require hand-crafted rules or lexicons to generate grammatical output but relies on the syntactic importance score calculated from an automatically parsed corpus. An experiment with native speakers demonstrated that our method generates more grammatical sentences than existing systems.

There are several directions to explore in the future. Recently query-based sentence fusion has been shown to be a better defined task than generic sentence fusion (Krahmer et al., 2008). By modifying the word informativeness score, e.g. by giving higher scores to words semantically related to the query, one could force our system to retain words relevant to the query in the output. To generate coherent texts we plan to move beyond sentence generation and add discourse constraints to our system.

Acknowledgements: This work has been funded by the Klaus Tschira Foundation, Heidelberg, Germany. The first author has been supported by a KTF grant (09.009.2004). Part of the data has been used with a permission of Bibliographisches Institut & F. A. Brockhaus AG, Mannheim, Germany. We would like to thank the participants in our online evaluation. We are also grateful to Regina Barzilay and the three reviewers for their helpful comments.

References

Barzilay, Regina & Kathleen R. McKeown (2005). Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–327.

Brants, Thorsten (2000). TnT – A statistical Part-of-Speech tagger. In *Proceedings of the 6th Conference on Applied Natural Language Processing*, Seattle, Wash., 29 April – 4 May 2000, pp. 224–231.

Clarke, James & Mirella Lapata (2008). Global inference for sentence compression: An integer linear programming approach. *Journal of Artificial Intelligence Research*, 31:399–429.

Clarkson, Philip & Ronald Rosenfeld (1997). Statistical language modeling using the CMU-Cambridge toolkit. In *Proceedings of the 5th European Conference on Speech Communication and Technology,*

[6] The corpus is available from `http://homepages.inf.ed.ac.uk/s0460084/data`.

Rhodes, Greece, 22-25 September 1997, pp. 2707–2710.

Filippova, Katja & Michael Strube (2007). Generating constituent order in German clauses. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics,* Prague, Czech Republic, 23–30 June 2007, pp. 320–327.

Filippova, Katja & Michael Strube (2008). Dependency tree based sentence compression. In *Proceedings of the 5th International Conference on Natural Language Generation,* Salt Fork, Ohio, 12–14 June 2008, pp. 25–32.

Foth, Kilian & Wolfgang Menzel (2006). Hybrid parsing: Using probabilistic models as predictors for a symbolic parser. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics,* Sydney, Australia, 17–21 July 2006, pp. 321–327.

Gurevych, Iryna & Hendrik Niederlich (2005). Accessing GermaNet data and computing semantic relatedness. In *Companion Volume to the Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics,* Ann Arbor, Mich., 25–30 June 2005, pp. 5–8.

Hori, Chiori & Sadaoki Furui (2004). Speech summarization: An approach through word extraction and a method for evaluation. *IEEE Transactions on Information and Systems*, E87-D(1):15–25.

Hovy, Eduard (2003). Text summarization. In Ruslan Mitkov (Ed.), *The Oxford Handbook of Computational Linguistics*, pp. 583–598. Oxford, U.K.: Oxford University Press.

Jing, Hongyan (2001). *Cut-and-Paste Text Summarization*, (Ph.D. thesis). Computer Science Department, Columbia University, New York, N.Y.

Kassner, Laura, Vivi Nastase & Michael Strube (2008). Acquiring a taxonomy from the German Wikipedia. In *Proceedings of the 6th International Conference on Language Resources and Evaluation,* Marrakech, Morocco, 26 May – 1 June 2008.

Krahmer, Emiel, Erwin Marsi & Paul van Pelt (2008). Query-based sentence fusion is better defined and leads to more preferred results than generic sentence fusion. In *Companion Volume to the Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics,* Columbus, Ohio, 15–20 June 2008, pp. 193–196.

Langkilde, Irene & Kevin Knight (1998). Generation that exploits corpus-based statistical knowledge. In *Proceedings of the 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics,* Montréal, Québec, Canada, 10–14 August 1998, pp. 704–710.

Lemnitzer, Lothar & Claudia Kunze (2002). GermaNet – representation, visualization, application. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation,* Las Palmas, Canary Islands, Spain, 29–31 May 2002, pp. 1485–1491.

Marsi, Erwin & Emiel Krahmer (2005). Explorations in sentence fusion. In *Proceedings of the European Workshop on Natural Language Generation,* Aberdeen, Scotland, 8–10 August, 2005, pp. 109–117.

McKeown, Kathleen R., Judith L. Klavans, Vassileios Hatzivassiloglou, Regina Barzilay & Eleazar Eskin (1999). Towards multidocument summarization by reformulation: Progress and prospects. In *Proceedings of the 16th National Conference on Artificial Intelligence,* Orlando, Flo., 18–22 July 1999, pp. 453–460.

Nelken, Rani & Stuart Schieber (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,* Trento, Italy, 3–7 April 2006, pp. 161–168.

Reiter, Ehud & Robert Dale (2000). *Building Natural Language Generation Systems*. Cambridge, U.K.: Cambridge University Press.

Ringger, Eric, Michael Gamon, Robert C. Moore, David Rojas, Martine Smets & Simon Corston-Oliver (2004). Linguistically informed statistical models of constituent structure for ordering in sentence realization. In *Proceedings of the 20th International Conference on Computational Linguistics,* Geneva, Switzerland, 23–27 August 2004, pp. 673–679.

Schmid, Helmut (1997). Probabilistic Part-of-Speech tagging using decision trees. In Daniel Jones & Harold Somers (Eds.), *New Methods in Language Processing*, pp. 154–164. London, U.K.: UCL Press.

Spärck Jones, Karen (1999). Automatic summarizing: Factors and directions. In Inderjeet Mani & Mark T. Maybury (Eds.), *Advances in Automatic Text Summarization*, pp. 1–12. Cambridge, Mass.: MIT Press.

Strube, Michael & Simone Paolo Ponzetto (2006). WikiRelate! Computing semantic relatedness using Wikipedia. In *Proceedings of the 21st National Conference on Artificial Intelligence,* Boston, Mass., 16–20 July 2006, pp. 1419–1424.

Wan, Stephen, Robert Dale, Mark Dras & Cecile Paris (2007). Global revision in summarization: Generating novel sentences with Prim's algorithm. In *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics,* Melbourne, Australia, 19–21 September, 2007, pp. 226–235.