Top a Splitter:
Using Distributional Semantics for Improving Compound Splitting

Patrick Ziering **Stefan Müller**
Institute for Natural Language Processing
University of Stuttgart, Germany
{zierinpk,muellesn}
@ims.uni-stuttgart.de

Lonneke van der Plas
Institute of Linguistics
University of Malta, Malta
Lonneke.vanderPlas@um.edu.mt

Abstract

We present a flexible method that rearranges the ranked output of compound splitters (i.e., decomposers of one-word compounds such as the German *Kinderlied* ‘children’s song’) using a distributional semantics model. In an experiment, we show that our re-ranker improves the quality of various compound splitters.

1 Introduction

Closed nominal compounds (i.e., one-word compounds such as the German *Eidotter* ‘egg yolk’) are one of the most productive word formation types in Germanic languages such as German, Dutch or Swedish, and constitute a major class of multi-word expressions (MWEs). Baroni (2002) presents a German corpus study showing that almost half of the corpus types are compounds, while the token frequency of individual compounds is low. This makes it hard to process closed compounds with general-purpose statistical methods and necessitates automatic compound analysis as a principal part of many natural language processing tasks such as statistical machine translation (SMT).

Therefore, previous work has tried to tackle the task of compound splitting (e.g., decomposing *Eidotter* to *Ei* ‘egg’ and *Dotter* ‘yolk’). Most compound splitters follow a generate-and-rank procedure. Firstly, all possible candidate splits are generated, e.g., *Ei*|*dotter*, *Eid*|*otter*, ..., *Eidott*|*er* (Koehn and Knight, 2003) or a knowledge-rich morphological analyzer provides a set of plausible candidate splits (Fritzinger and Fraser, 2010). In a second step, the list of candidate splits is ranked according to statistical features such as constituent frequency (Stymne, 2008; Macherey et al., 2011; Weller and Heid, 2012) or frequency of morphological operations (Ziering and Van der Plas, 2016). By considering each constituent in isolation, approaches limited to frequency neglect the semantic compatibility between a compound and its constituents. For example, while *Eidotter* is usually understood as the yolk of an egg (i.e., *Ei*|*dotter*), the low frequency of *Dotter* often makes frequency-based splitters rank a less plausible interpretation higher: *Eid*|*otter* ‘oath otter’.

We try to tackle this pitfall by enriching the ranked output of various splitters with a semantic compatibility score. Our method is inspired by recent work on the prediction of compound compositionality using distributional semantics (Reddy et al., 2011; Schulte im Walde et al., 2013). The distributional measures that are used to predict the compositionality of compounds are in fact measuring the semantic similarity between the compound and its constituents. Our assumption is that they can therefore be used readily to rank the candidate constituents a splitter proposes and help to promote more plausible candidate splits (e.g., *Eidotter* is distributionally more similar to *Dotter* than to *Otter*). Previously, Weller et al., (2014) applied compositionality measures to compound splitting as a pre-processing step in SMT. Their intuition is that non-compositional compounds benefit less from splitting prior to SMT. However, they found no improvements in the extrinsic evaluation. Neither did they find improvements from applying distributional semantics directly to the unordered list of candidate splits. We will show in an intrinsic evaluation that distributional semantics, when combined with the initial ranked output of various splitters does lead to a statistically significant improvement in compound splitting.

Other works that used semantic information for compound splitting include Bretschneider and Zillner (2015), who developed a splitting approach relying on a semantic ontology of the medical do-

Proceedings of the 12th Workshop on Multiword Expressions, pages 50–55,
Berlin, Germany, August 7-12, 2016. ©2016 Association for Computational Linguistics

main. They disambiguated candidate splits using semantic relations from the ontology (e.g., *Beckenbodenmuskel* 'pelvic floor muscle' is binary split to *Beckenboden* | *muskel* using the `part_of` relation). As back-off strategy, if the ontology lookup fails, they used constituent frequency. We do not restrict to a certain domain and related ontology but use distributional semantics in combination with frequency-based split features for the disambiguation.

Daiber et al., (2015) developed a compound splitter based on semantic analogy (e.g., *bookshop* is to *shop* as *bookshelf* is to *shelf*). From word embeddings of compound and head word, they learned prototypical vectors representing the modification. During splitting, they determined the most suitable modifier by comparing the analogy to the prototypes. While Daiber et al., (2015) developed an autonomous splitter and focused on semantic analogy, we present a re-ranker that combines distributional similarity with additional splitting features.

Very recently, Riedl and Biemann (2016) developed a semantic compound splitter that uses a pre-compiled distributional thesaurus for searching semantically similar substrings of a compound subject to decomposition. While their stand-alone method focuses on knowledge-lean split point determination, our approach improves splitters including the task of constituent normalization.

Our contributions are as follows. We are the first to show that distributional semantics information as an additional feature helps in determining the best split among the candidate splits proposed by various compound splitters in an intrinsic evaluation. Moreover, we present an architecture that allows for the addition of distributional similarity scores to any compound splitter by re-ranking a system's output.

2 Re-ranking based on distributional semantics

2.1 Initial split ranking

Our method is applicable to any compound splitter that produces a ranked output of split options[1] with their corresponding ranking score.

For example, the target compound *Fischerzeugnis* 'fish product' is processed by a compound splitter yielding the output as given in Table 1. The top-ranked candidate split is the result from a falsely triggered normalization rule (i.e., *+er* is not a valid linking element for *Fisch*).

Ranking score	Candidate split	Correct?
14264	*Fisch* + *Zeugnis* 'fish certificate'	✗
9390	*Fisch* + *Erzeugnis* 'fish product'	✓
5387	*Fischer* + *Zeugnis* 'fisherman certificate'	✗

Table 1: Initial split ranking

2.2 Determination of distributional similarity

For each candidate split of a target compound (e.g., *Fisch* | *erzeugnis* given *Fischerzeugnis*), the cosine similarity between the target compound and each candidate constituent is determined as a standard measure that is used for computing the distributional similarity (DS). In a following step, these cosine values are used to predict the degree of semantic relatedness between the target compound and the candidate modifier (MOD) or head (HEAD), respectively. As proposed by Weller et al., (2014), a possible combination of the candidate constituents' cosine values is the geometric mean (GEO). For example, let $\cos(\overrightarrow{Fischerzeugnis}, \overrightarrow{Fisch})$ be 0.455 and $\cos(\overrightarrow{Fischerzeugnis}, \overrightarrow{Erzeugnis})$ be 0.10. The GEO DS score for the lexemes derived from *Fisch*|*erzeugnis* is $\sqrt{0.455 \cdot 0.10} \approx 0.22$.

2.3 Combination and re-ranking

In the next step, we multiply the DS scores with the initial split ranking scores and finally re-rank the splits according to the resulting product. Table 2 shows the result from re-ranking the output presented in Table 1 with GEO DS scores.

Re-ranking score	Candidate split	Correct?
9390 · 0.22 ≈ **2034**	*Fisch* + *Erzeugnis* 'fish product'	✓
14264 · 0.05 ≈ **709**	*Fisch* + *Zeugnis* 'fish certificate'	✗
5387 · 0.01 ≈ **70**	*Fischer* + *Zeugnis* 'fisherman certificate'	✗

Table 2: Split re-ranking with GEO DS scores

[1] Following Weller et al., (2014), we focus on true compounds and ignore non-split options.

3 Experiments

3.1 Data

We use the German Wikipedia[2] corpus comprising 665M words. We tokenize, lemmatize and PoS-tag using TreeTagger (Schmid, 1995). While we are aware of the fact that there are German corpora larger than Wikipedia which can boost the perfomance of distributional semantics methods, we decided to use the same corpora as used in previous work for the inspected compound splitters (Ziering and Van der Plas, 2016). By controlling for corpus size, we can contrast the differences in splitting performance with respect to information type (i.e., distributional similarity vs. frequency information) irrespective of corpus size.

3.2 Distributional model

In analogy to the distributional model of Weller et al., (2014), we adopt a setting whose parameters are tuned on a development set and prove best for compositionality (Schulte im Walde et al., 2013). It employs corpus-based co-occurrence information extracted from a window of 20 words to the left and 20 to the right of a target word. We restrict to the 20K most frequent nominal co-occurrents.

3.3 Distributional similarity modes

Inspired by Weller et al., (2014), the distributional similarity mode (DS MODE) refers to the selected cosine values, determined with our distributional model. We compare the distributional similarity of both individual constituents (i.e., modifier (MOD) and head (HEAD)) with the geometric mean of them (GEO). Moreover, we used standard arithmetic operations (Widdows, 2008; Mitchell and Lapata, 2010) and combine the vectors of modifier and head by vector addition (ADD), and multiplication (MULT) as shown to be beneficial in Schulte im Walde et al., (2013).

3.4 Rankings in comparison

We compare the performance of the initial ranking (INITIAL) of a compound splitter, based on all individual features, with the splitting performance after re-ranking by multiplying the selected DS value with the initial ranking score (RR_{ALL}). Our baseline (RR_{DS}) is inspired by the aggressive splitting mode (DIST) of Weller et al., (2014): we re-rank the unordered list of candidate splits proposed by a splitter according to the DS scores only.

3.5 Inspected compound splitters

We inspect three different types of German compound splitters, ranging from knowledge-lean to knowledge-rich. **Ziering and Van der Plas (2016)** developed a corpus-based approach, where morphological operations are learned automatically from word inflection. **Weller and Heid (2012)** used a frequency-based approach with a list of PoS-tagged lemmas and an extensive hand-crafted set of normalization rules. **Fritzinger and Fraser (2010)** combined the splitting output of the morphological analyzer SMOR (Schmid et al., 2004) with corpus frequencies.

3.6 Evaluation setup

While Weller at al., (2014) did not observe a difference in SMT performance between ranking candidate splits according to frequency and compositionality, we use an intrinsic evaluation measure actually revealing significant differences. We follow the evaluation approach of Ziering and Van der Plas (2016), who defined splitting accuracy[3] in terms of determining the correct split point (SPAcc) and correctly normalizing the resulting constituents (NormAcc), and use the GermaNet[4] gold standard developed by Henrich and Hinrichs (2011). We remove hyphenated compounds, which should be trivial splitting cases that do not need improvement by re-ranking. The final set comprises 51,230 compounds.

System	Test set size	Coverage
ZvdP_2016	51,194	99.9%
WH_2012	49.999	97.6%
FF_2010	47,940	93.6%

Table 3: Coverage of compound splitters

Some of the compound splitters described in Section 3.5 can only process a subset of the gold standard. For example, the approach of Fritzinger and Fraser (2010) is limited to a hand-crafted lexicon (i.e., it misses compounds with unknown constituents such as *Barbiepuppe* ‘Barbie doll’). Moreover, it uses the analyzer SMOR, which considers some gold standard compounds as cases of derivation which are not subject to decomposition (e.g., *Unterbesetzung* ‘understaffing’ is primarily derived from the verb *unterbesetzen* ‘to understaff’). Besides, for some compounds, there are

[2] de.wikipedia.org

[3] Accuracy refers to the top-ranked candidate split.

[4] sfs.uni-tuebingen.de/GermaNet

Accuracy	SPAcc					NormAcc				
DS MODE	MOD	HEAD	GEO	MULT	ADD	MOD	HEAD	GEO	MULT	ADD
ZIERING AND VAN DER PLAS (2016)										
INITIAL	97.5%					87.4%				
RR_{DS}	93.6%	94.6%	95.4%	92.7%	92.0%	75.9%	84.7%	77.8%	69.6%	61.2%
RR_{ALL}	97.5%	97.9%†	**98.0%**†	97.8%†	**98.0%**†	88.6%†	87.7%†	**89.0%**†	88.5%†	88.7%†
WELLER AND HEID (2012)										
INITIAL	98.1%					90.4%				
RR_{DS}	96.9%	97.0%	97.7%	96.9%	95.8%	86.5%	89.3%	87.1%	81.8%	75.3%
RR_{ALL}	98.2%†	98.2%†	**98.3%**†	98.2%†	**98.3%**†	**91.3%**†	90.5%†	91.1%†	90.9%†	90.9%†
FRITZINGER AND FRASER (2010)										
INITIAL	98.4%					94.9%				
RR_{DS}	97.9%	97.9%	98.4%	98.3%	98.2%	94.3%	94.3%	94.7%	94.5%	94.3%
RR_{ALL}	98.4%	98.3%	**98.5%**	98.4%	98.4%	94.8%	94.7%	**95.0%**	94.8%	94.7%

Table 4: Results of split re-ranking; † indicates significantly better than INITIAL

no binary splits in a system's ranking. These compounds are excluded from the respective splitter's test set. Table 3 shows the test set sizes and coverage of the inspected compound splitters.

4 Results and discussion

In the following section, we show results on splitting performance of various compound splitters before and after adding our re-ranking method. As shown in Table 3, the systems are evaluated on different test sets. It is not our goal to compare different splitting methods against each other, but to show the universal applicability of our re-ranker for different types of splitters.

4.1 General trends

Table 4 shows the performance numbers for all inspected compound splitters and all DS modes. A **first result** is that the INITIAL accuracy (both SPAcc and NormAcc) is always outperformed by re-ranking with DS scores as additional feature (RR_{ALL}) for at least one DS MODE.

The **baseline** of using pure DS scores (RR_{DS}) worsens the INITIAL performance. This is in line with previous work (Weller et al., 2014) and shows that isolated semantic information does not suffice but needs to be introduced as an additional feature. In an error analysis, we observed that the corpus frequency, which is missing for RR_{DS}, is a crucial feature for compound splitting and helps to demote analyses based on typographical errors or unlikely modifier normalization. For example, while RR_{ALL} analyzes the compound *Haarwasser* 'hair tonic' with the correct and highly frequent modifier *Haar* 'hair', RR_{DS} selects the morphologically plausible but yet unlikely and infrequent verbal modifier *haaren* 'to molt', which happens to have the higher cosine similarity to *Haarwasser*.

Another type of compound analysis that benefits from corpus frequency is binary splitting of left-branched tripartite compounds (i.e., bracketing). For example, the compound *Blinddarmoperation* 'appendix operation' (lit.: 'blind intestine operation') is frequency-based correctly split into *Blinddarm* | *operation* '[appendix] operation', whereas RR_{DS} prefers the right-branched splitting into *Blind* | *darmoperation* 'blind [intestine operation]'. Since the rightmost constituent *Operation* 'surgery/operation' is more ambiguous, it has a smaller cosine similarity to the entire compound than the right-branched compound *Darmoperation* 'intestinal operation'. In contrast, the high corpus frequency of the non-compositional *Blinddarm* 'appendix' and the head *Operation*, make a frequency-based splitter choose the correct structure. However, bracketing also benefits from cosine similarity. For example, using re-ranking by RR_{ALL}, the wrong compound split *Arbeits|platzmangel* 'labor [lack of space]' is corrected to *Arbeitsplatz|mangel* 'job scarcity'. As conclusion, we argue that the combination of corpus frequency and semantic plausibility (in terms of cosine similarity) is working best for splitting.

Comparing the accuracy types, we see that the determination of the correct split point is the easier task and achieves a SPAcc of 98.5%

(GEO@RR$_{\text{ALL}}$ for Fritzinger and Fraser's (2010) splitter). However, there is only a small benefit for SPAcc when adding semantic support. In contrast, constituent normalization (measured as NormAcc) can be improved by +1.6% (GEO@RR$_{\text{ALL}}$ for Ziering and Van der Plas' (2016) splitter).

Comparing the DS modes, we see that for NormAcc, the more demanding task that leads to the largest differences in performance between the different modes, the MOD mode outperforms the HEAD mode (for RR$_{\text{ALL}}$). However, the modes that combine head and modifier scores mostly outperform those based on heads or modifiers in isolation. This is in line with tendencies found in previous work on compositionality of compounds (Schulte im Walde et al., 2013). In addition, we find that for NormAcc, the GEO mode outperforms the modes based on vector arithmetic, whereas for SPAcc, the performance of GEO and the vector addition (ADD) is comparable.

4.2 Individual splitter improvement

Ziering and Van der Plas (2016) automatically learned constituent transformations taking place during compounding (e.g., *s*-suffixation) from word inflection. Based on corpus frequency and transformation plausibility, they produced a ranked list of candidate splits. However, misleading inflections can rank false splits high. For example, +*ge*, as in the participle *aufgewachsen* 'grown up' (*aufwachsen* 'grow up'), leads to the falsely top-ranked candidate split *Fu(ge)nk* | *elle* 'radio ulna' instead of *Fugen* | *kelle* 'filling trowel'. Re-ranking with RR$_{\text{ALL}}$ promotes the correct candidate split. We achieve significant[5] improvements for almost all DS MODEs.

Weller and Heid (2012) extended a frequency-based approach (Koehn and Knight, 2003) with a hand-crafted set of morphological rules. Even restricted to only valid constituent transformations, some rules are falsely triggered and lead to wrong splits. For example, the *er*-suffix (as in *Kinder* | *buch* 'children's book') is used for the compound *Text* | *erkennung* 'text recognition' and results in the false split *Text(er)* | *kennung* 'text ID'. Our re-ranking method (RR$_{\text{ALL}}$) again helps to promote the correct candidate split. In all DS MODES, the performance is improved significantly.

For the system of **Fritzinger and Fraser (2010)**, the GEO mode improves the INITIAL splitting accuracy (+0.1%), but we do not achieve statistically significant results. The main reason for this is due to the lexicon-based morphological analyzer SMOR. While having the smallest coverage on the gold standard, utilizing a hand-crafted lexicon results in only correctly triggered transformation rules. This leads to a smaller list of candidate splits. In fact, the average number of analyses provided by Fritzinger and Fraser (2010) is much smaller than for Ziering and Van der Plas (2016) as shown in Table 5.

System	Avg # candidate splits
ZvdP_2016	4.31
WH_2012	2.25
FF_2010	1.11

Table 5: Average number of candidate splits

As a consequence, re-ranking has only a limited impact on the splitting performance. We can conclude that a knowledge-rich morphological resource can mitigate the need for semantic support, however, at the expense of coverage.

5 Conclusion

We presented a flexible method for re-arranging the ranked output of a compound splitter, by adding a feature for the semantic compatibility between compound and potential constituents derived from a distributional semantics model. We showed that the addition of distributional similarity significantly improves different types of compound splitters.

Acknowledgments

We thank the anonymous reviewers for their helpful feedback. We also thank Marion Di Marco and Fabienne Cap for providing their splitting methods. This research was funded by the German Research Foundation (Collaborative Research Centre 732, Project D11).

References

Marco Baroni, Johannes Matiasek, and Harald Trost. 2002. Predicting the Components of German Nominal Compounds. In *ECAI*, pages 470–474. IOS Press.

Claudia Bretschneider and Sonja Zillner. 2015. Semantic Splitting of German Medical Compounds. In *Text, Speech, and Dialogue*. Springer International Publishing.

[5] Approximate randomization test (Yeh, 2000), $p < 0.05$

Joachim Daiber, Lautaro Quiroz, Roger Wechsler, and Stella Frank. 2015. Splitting Compounds by Semantic Analogy. *CoRR*.

Fabienne Fritzinger and Alexander Fraser. 2010. How to Avoid Burning Ducks: Combining Linguistic Analysis and Corpus Statistics for German Compound Processing. In *Proceedings of the ACL 2010 Joint 5th Workshop on Statistical Machine Translation and Metrics MATR*, pages 224–234.

Verena Henrich and Erhard W. Hinrichs. 2011. Determining Immediate Constituents of Compounds in GermaNet. In *RANLP 2011*, pages 420–426.

Philipp Koehn and Kevin Knight. 2003. Empirical methods for compound splitting. In *EACL*.

Klaus Macherey, Andrew M. Dai, David Talbot, Ashok C. Popat, and Franz Och. 2011. Language-independent Compound Splitting with Morphological Operations. In *ACL HLT 2011*.

Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive Science*, 34:1388–1429.

Siva Reddy, Diana McCarthy, and Suresh Manandhar. 2011. An Empirical Study on Compositionality in Compound Nouns. In *IJCNLP 2011*.

Martin Riedl and Chris Biemann. 2016. Unsupervised Compound Splitting With Distributional Semantics Rivals Supervised Methods. In *NAACL-HTL 2016*.

Helmut Schmid, Arne Fitschen, and Ulrich Heid. 2004. SMOR: A German Computational Morphology Covering Derivation, Composition, and Inflection. In *LREC 2004*, pages 1263–1266.

Helmut Schmid. 1995. Improvements in Part-of-Speech Tagging with an Application to German. In *ACL SIGDAT-Workshop*.

Sabine Schulte im Walde, Stefan Müller, and Stephen Roller. 2013. Exploring Vector Space Models to Predict the Compositionality of German Noun-Noun Compounds. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*.

Sara Stymne. 2008. German Compounds in Factored Statistical Machine Translation. In *GoTAL*.

Marion Weller and Ulrich Heid. 2012. Analyzing and Aligning German compound nouns. In *LREC 2012*.

Marion Weller, Fabienne Cap, Stefan Müller, Sabine Schulte im Walde, and Alexander Fraser. 2014. Distinguishing Degrees of Compositionality in Compound Splitting for Statistical Machine Translation. In *ComAComA 2014*.

Dominic Widdows. 2008. Semantic Vector Products: Some Initial Investigations. In *Proceedings of the Second AAAI Symposium on Quantum Interaction*.

Alexander Yeh. 2000. More Accurate Tests for the Statistical Significance of Result Differences. In *COLING 2000*.

Patrick Ziering and Lonneke van der Plas. 2016. Towards Unsupervised and Language-independent Compound Splitting using Inflectional Morphological Transformations. In *NAACL-HLT 2016*.