Evaluating Compound Splitters Extrinsically with Textual Entailment

Glorianna Jagfeld **Patrick Ziering**
Institute for Natural Language Processing
University of Stuttgart
{jagfelga,zierinpk}
@ims.uni-stuttgart.de

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

Abstract

Traditionally, compound splitters are evaluated intrinsically on gold-standard data or extrinsically on the task of statistical machine translation. We explore a novel way for the extrinsic evaluation of compound splitters, namely recognizing textual entailment. Compound splitting has great potential for this novel task that is both transparent and well-defined. Moreover, we show that it addresses certain aspects that are either ignored in intrinsic evaluations or compensated for by task-internal mechanisms in statistical machine translation. We show significant improvements using different compound splitting methods on a German textual entailment dataset.

1 Introduction

Closed compounding, i.e., the formation of a one-word unit composing several lexemes, is a common linguistic phenomenon in several languages such as German, Dutch, Greek, and Finnish. The goal of compound splitting is to obtain the constituents of a compound to increase its semantic transparency. For example, for the German compound *Apfelsaft* 'apple$_1$ juice$_2$' the desired output of a compound splitter is *Apfel*$_1$ *Saft*$_2$.

Intrinsic evaluation of compound splitting measures the correctness of the determined split point (Riedl and Biemann, 2016) and the resulting lemmas by means of precision, recall, F_1-score and accuracy (e.g., Koehn and Knight (2003)). In *extrinsic evaluation* setups, compound splitting is applied to the input data of an external natural language processing (NLP) task that benefits from split compounds. As closed compounding introduces semantic opaqueness and vastly increases the vocabulary size of a language, many NLP tasks benefit from compound splitters. Still, previous work that evaluates compound splitting with extrinsic evaluation methods mostly focuses on statistical machine translation (SMT) (e.g., Nießen and Ney (2000), Koehn and Knight (2003)). Some other external tasks such as information retrieval (Kraaij and Pohlmann, 1998) or speech recognition (Larson et al., 2000) have been shown to benefit from prior compound splitting, yet these works have not compared the extrinsic performance of different compound splitting methods.

Interestingly, the performance found in intrinsic evaluations does not automatically propagate to performance in downstream evaluations as shown in (Fritzinger and Fraser, 2010) for SMT, where oversplit compounds are simply learned as phrases (Dyer, 2009; Weller et al., 2014). Oversplitting is an example of a feature that might not be measured in intrinsic evaluations, because some available gold standards contain positive examples only (Ziering and van der Plas, 2016). It is highly relevant to increase the number of extrinsic tasks for the evaluation of compound splitting to be able to evaluate features that intrinsic evaluations and known extrinsic evaluations ignore.

In this paper we investigate the suitability of Recognizing Textual Entailment (RTE) for the task of compound splitting, inspired by the fact that previous work in RTE underlined the potential benefits of compound splitting for this task (Zeller, 2016). Textual Entailment (TE) is a directional relationship between an entailing text fragment T and an entailed hypothesis, H, saying that the meaning of T entails (or implies) the meaning of H. This relation holds if 'typically, a human, reading T, would infer that H is most likely true' (Dagan et al., 2006). The following is an example of an entailing T-H pair:

T: Yoko Ono unveiled a bronze statue of her late husband, John Lennon.
H: Yoko Ono is John Lennon's widow.

Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 58–63
Vancouver, Canada, July 30 - August 4, 2017. ©2017 Association for Computational Linguistics
https://doi.org/10.18653/v1/P17-2010

We opted for exploring the use of RTE as an extrinsic evaluation for compound splitting for three main reasons: first, in contrast to SMT systems, most RTE systems are less complex. In fact, we deliberately chose an RTE system that reaches good performance with a method that is transparent, i.e., a method that allows for exploring the effect of compounding.[1] It is not our goal to reach state-of-the-art performance for the RTE task. We aim to find a suitable alternative extrinsic evaluation for compound splitting. Second, human agreement on the binary RTE decisions is very high, e.g., on the dataset used in our experiments, an average agreement rate of 87.8% with a κ level of 0.75 was reported (Giampiccolo et al., 2007). Third, the potential benefits for RTE are large. According to Zeller (2016, p. 182) the number of T/H pairs in their phenomenon-specific RTE dataset would rise by about 16 percentage points by compound splitting. In the dataset we use in our experiments, about three-quarters of the T-H pairs contain at least one closed compound.

2 Relevance of Compound Splitting for RTE

The approach to RTE taken in this paper follows the *Lexical Overlap Hypothesis* (LOH), which states that the higher the number of lexical matches between T and H, the more likely the T-H pair is entailing rather than non-entailing (Zeller, 2016). In other words, H is more likely to be entailed by T if most of its lexical content also occurs in T. While this hypothesis is a simplification of the TE problem, it has been shown to perform reasonably well for some datasets (Noh et al., 2015). We argue that the brittleness of the chosen LOH-based RTE system may actually be a strength in terms of evaluation, since it will penalize oversplitting more severely than, e.g., an RNN-based RTE system or a phrase-based MT method that can recover from systematic oversplitting by chunking the splits.

Under the LOH, the problem caused by the opacity of closed compounds becomes evident. As shown in the example below, missing information on the constituents of closed compounds hinders the matching of words from T in H1. Conversely, compound splitting also helps to detect non-entailing T-H pairs. By compound splitting, we increase the number of uncovered tokens in H2, which makes a non-entailment decision more likely[2].

T: Kinder lieben Frucht**säfte**$_1$ aus **Äpfeln**$_2$ ‘*Children love fruit **juices**$_1$ made of **apples**$_2$*’

H1: Peters Sohn liebt **Apfel**$_3$**saft**$_4$ ‘Peter’s son loves **apple**$_3$ **juice**$_4$’

H2: Peters Sohn liebt **Apfel**$_5$**kuchen**$_6$**stücke**$_7$ ‘Peter’s son loves **pieces**$_7$ of **apple**$_5$ **pie**$_6$’

3 Materials and Methods

In this section we explain the splitters and the RTE framework used in our experiments.

3.1 Inspected Compound Splitters

Our proposed extrinsic evaluation approach for compound splitting is language-independent as we do not use any language-specific parameters. However, in the present work we test it on the most prominent closed-compounding language, German (Ziering and van der Plas, 2014). We inspect the impact of three different types of automatic compound splitting[3] methods that follow a generate-and-rank principle, where the candidate splits are ranked according to the geometric mean of the constituents’ frequencies in a given training corpus (Koehn and Knight, 2003).

FF2010 The compound splitter by Fritzinger and Fraser (2010) relies on the output of the German morphological analyzer SMOR (Schmid et al., 2004) to generate several plausible compound splits (e.g., due to word sense ambiguity).

WH2012 As an alternative method, we use the statistical approach presented in Weller and Heid (2012) for German compound splitting. Instead of using the knowledge-rich SMOR, it includes an extensive list of hand-crafted transformation rules that allows to map constituents to corpus lemmas (e.g., by truncating linking morphemes) to generate all possible splits with up to four constituents per compound. Moreover, misleading lemmas are removed from the training corpus using hand-crafted filters.

[1] We did not opt for neural RTE systems (Bowman et al., 2015), albeit state-of-the-art, in this first study because of the opacity of the models and the inclusion of phrase-level information, which will make interpretation of the effect harder.

[2] Note that we need to apply lemmatization prior to determining the lexical matches between T and H.

[3] The compound splitters are designed to split compounds with any content word as head, i.e., noun compounds (*Hunde*|*hütte* ‘doghouse’), verb compounds (*eis*|*laufen* ‘to ice-skate’) and adjective compounds (*hunde*|*müde* ‘dog-tired’) and disregard constructions with a functional modifier (as in the particle verb *auf*|*stehen* ‘to stand up’).

System	Acc	Entailment			Non-entailment		
		P	R	F$_1$	P	R	F$_1$
INIT	64.13	62.50	74.57	68.00	66.67	53.20	59.18
manual splitting ⋆	67.88	65.08	80.20	71.85	72.64	54.99	62.59
ZvdP2016	66.63	64.55	77.02	70.23	69.87	55.75	62.02
FF2010 ⋆	**67.38**	**65.48**	76.53	**70.58**	**70.19**	**57.80**	**63.39**
WH2012	66.00	63.73	**77.75**	70.04	69.77	53.71	60.69

Table 1: Results on RTE performance without (INIT) and with prior compound splitting. ⋆: significant difference of the performance in comparison to INIT

ZvdP2016 Finally, the method using least language-specific knowledge was proposed by Ziering and van der Plas (2016). Instead of using a morphological analyzer or manually compiling a hand-crafted list of rules, they recursively generate all possible binary splits by learning constituent transformations from regular inflection derived from a monolingual lemmatized corpus, e.g., the *s*-suffix in the case of a genitive marker is often used as linking morpheme. The recursion stops if a non-splitting (atomic) analysis is ranked highest.

Additionally, to provide an upper bound, we manually split development and test data.

3.2 RTE Framework

We conduct our RTE experiments using the open-source Excitement Open Platform (EOP) (Padó et al., 2015; Magnini et al., 2014), which provides comprehensive implementations of algorithms and lexical resources for textual inference. We use the alignment-based algorithm P1EDA (Noh et al., 2015) in all our experiments as it has been shown to be simple and transparent while yielding relatively good results. P1EDA is based on the LOH for RTE explained in Section 2. The algorithm works in three steps: First, it extracts all possible alignments between sequences of identical lemmas in T and H. Then, it extracts various features[4] from the alignments. Finally, these features are given as input to a multinomial logistic regression classifier which is trained on annotated data. For the sake of simplicity, for now we only use one basic aligner which aligns (sequences of) words in T and H that consist of identical lemmas. We will investigate the impact of prior compound splitting given additional lexical resources (such as a derivational morphology lexicon (Zeller et al., 2013)) in future work. We use TreeTagger (Schmid, 1995) as integrated in EOP to provide tokenization, lemmatization and Part-of-Speech tagging as linguistic preprocessing.

We train and evaluate all models on the German translation of the RTE-3 dataset (Dagan et al., 2006; Magnini et al., 2014). The training and test dataset contain 800 T-H pairs each. In both sets, entailing and non-entailing T-H pairs are equally distributed (chance baseline of 50% accuracy).

We apply a compound splitter on the RTE training and test dataset *before* we input the data to the EOP pipeline. We replace all compounds by their constituents, separated by white-space. Thus, they are subsequently treated as individual words by EOP and the lexical aligner can benefit from the increased transparency of the compounds.

4 Results and Discussion

Table 1 shows accuracy, precision, recall and F$_1$-score for the entailment and non-entailment class on the RTE-3 dataset. As reflected in the results, reducing the opacity of compounds via the application of a compound splitter improves the subsequent RTE performance. This holds for all compound splitters that we used in our experiments. It is also noticeable that the different compound splitters yield different results in the downstream task, with FF2010 being the most beneficial and significantly[5] outperforming the initial RTE setup without prior compound splitting (INIT) by up to four percentage points in accuracy and F$_1$-score.

As expected, manual splitting performs best overall. The performance difference with FF2010 is however not statistically significant. This is not surprising because FF2010 reaches an accuracy of around 90% in intrinsic evaluations (Ziering and van der Plas, 2016) and the small underperfor-

[4] We use a similar feature set as Noh et al. (2015), namely the ratio of aligned vs. unaligned words in H with respect to all words, content words, and named entities.

[5] McNemar test (McNemar, 1947), $p < 0.05$

mance is leveled out by the small size of the test set. Moreover, manual inspections revealed that FF2010 has a higher recall than manual splitting in the non-entailment class due to its undersplitting which results in less lexical overlap between T and H, pointing to the non-entailment class.

When we compare these results from the extrinsic evaluation with intrinsic evaluation results (in terms of splitting accuracy) reported in Ziering and van der Plas (2016), we see the same performance ordering with respect to the three compound splitters, while the current extrinsic evaluation on RTE differentiates between the best system (FF2010) and the two others in that only the former reached statistically significant improvements over the INIT baseline.

To analyze the possible causes of difference in performance between the systems and to see the benefits of using RTE for compound splitting evaluation we performed a manual error analysis. First, we examined all entailment classifications that were correct using FF2010 and incorrect when using the INIT baseline. Using FF2010, the classifier was able to correctly classify an additional 36 entailing and 25 non-entailing T-H pairs. As expected, most of the hypotheses in these pairs contained correctly split compounds where the RTE system could benefit from the increased transparency. Conversely, we also examined the 28 T-H pairs that the classifier missed to identify as entailing while they were correct in INIT. Most of the examples were cases in which there was almost no lexical overlap between T and H even with compound splitting.

Furthermore, we compared the correct entailment classifications of FF2010 with the other two splitters. For ZvdP2016, most errors can be attributed to oversplitting. Precisely, 25 out of its 37 (67.5%) misclassifications compared to FF2010 can be attributed to this problem. For example, ZvdP2016 oversplit the name *Landowska* into *Line Dow Ska*[6] that appeared in both T and H in an non-entailing pair, which artificially increased the coverage ratio of words in H and therefore pointed to the incorrect entailment classification. For WH2012, oversplitting is also a major contributor of RTE errors, however it appeares not as predominant as for ZvdP2016. 10 out of its 29 (34.5%) misclassifications compared to FF2010 can be attributed to oversplitting, while 4 (13.8%) missclassifications are due to undersplitting. For example, in an entailing T-H pair WH2012 correctly split *Amazonas-Regenwald* 'Amazon rainforest' in H into *Amazonas Regen Wald*, however it oversplit *Amazonas* in T into *Amazon As* 'Amazon ace' and thus, *Amazonas* in H remained unaligned. To the contrary, FF2010 did not split *Amazonas* in T, which lead to a higher token coverage ratio in H. Again, in the H *Die EU senkt die Fangquoten* 'The EU lowers the fishing quota' of another entailing T-H pair, WH2010 correctly split *Fang*$_1$*quoten*$_2$ 'fishing quota' in H into *fangen*$_1$ *Quote*$_2$ but failed to split *EU-Quote* in T, failing to cover both *EU* and *Quote* in H.

Our closer inspections also showed that compound splitting does not always suffice to reveal a lexical match between T and H as shown in the following example:

T: Ben **fährt**$_1$ einen **Mercedes**$_2$ '*Ben **drives**$_1$ a **Mercedes**$_2$*'

H: Ben ist **Auto**$_3$**fahrer**$_4$ 'Ben is a **car**$_3$ **driver**$_4$'

Given a correct splitting of *Autofahrer* to *Auto Fahrer*, a derivational morphology resource (Zeller et al., 2013) would be required to discover the relationship between *fahren* and *Fahrer* and a synonym database to find that *Mercedes* is a hyponym of *Auto*. This does not weaken the claim that RTE is useful for evaluating compound splitters. It just shows that deeper, semantic compound analysis could improve RTE further.

Besides, the error analysis shed some light on the treatment of compound heads and modifiers. It seems advisable to weight the compound head and modifiers differently when computing the ratio of aligned tokens in H. As illustrated by the following example, coverage of the head should be more important for the entailment decision than of the modifiers. Given a correct split of *Kinder*$_1$*buch*$_2$ into *Kind*$_1$ *Buch*$_2$, H1 and H2 have the same token coverage ratio while only H1 is entailed by T.

T: Yuki kauft ein **Kinder**$_1$**buch**$_2$ '*Yuki buys a **children's**$_1$ **book**$_2$*'

H1: Yuki kauft ein **Buch** 'Yuki buys a **book**'

H2: Yuki ist ein **Kind** 'Yuki is a **child**'

It should be noted that the transparency gain using compound splitting is limited to closed compounds that are compositional with respect to at least one constituent. Splitting compounds in

[6] Misleading knowledge about verbal inflection automatically derived from a lemmatized corpus is responsible for the oversplitting by ZvdP2016.

H that are fairly non-compositional with respect to all constituents (e.g., *Maulwurf* 'mole' (lit. 'mouth throw')) is counterproductive. However, since most compounds (in particular ad-hoc productions) are compositional, this is only a side issue. In fact, we did not observe any cases of non-compositional compounds in the course of our error analysis.

In summary, compound splitting is a complex task that comprises many subtasks. The multiple evaluation methods available, both intrinsic and extrinsic, vary in their suitability to evaluate them. One of these subtasks concerns the ability of compound splitters to determine whether to split or not, which is an integral part of compound analysis. While aspects such as oversplitting were not consistently evaluated in previous intrinsic evaluations, or compensated for by task-internal mechanisms in SMT, RTE proved more strict in this respect. Moreover, the transparency of the models made it possible to better estimate the impact of splitting. Despite the small size of the dataset, we were able to show significant differences, partly due to the clear definition of this binary classification task.

On a side note, to the best of our knowledge, the result we obtained using the FF2010 compound splitter is the best result on the German RTE-3 dataset that has been reported using EOP. Notably, we obtain an accuracy which is almost three percentage points higher than the results of Noh et al. (2015), although they include further (language-specific) linguistic knowledge.

5 Conclusion

Inspired by the potential benefits of compound splitting from the RTE literature and supported by the transparency of the models and the clear definition of this binary classification task, we set out to explore whether RTE is a suitable method to extrinsically evaluate the performance of compound splitting. We compared several compound splitters on a German textual entailment dataset and found that compound splitting is helpful for RTE across the board. More importantly, we found that certain aspects of compound splitters, neglected in previous evaluations, such as oversplitting, had a large impact on this task and nicely differentiated the systems tested. We conclude that RTE represents a suitable alternative to SMT for the extrinsic evaluation of compound splitters.

In future work, we would like to investigate the interaction between additional lexical resources (such as GermaNet (Hamp and Feldweg, 1997; Henrich and Hinrichs, 2010) or DErivBase (Zeller et al., 2013)) and compound splitting, and the impact on the RTE performance.

Acknowledgments

We thank Sebastian Padó and Britta Zeller for their inspiring ideas about using compound splitting for improving RTE, and for their support in using the RTE framework. We are also grateful to the anonymous reviewers for their helpful feedback. This research was funded by the German Research Foundation (Collaborative Research Centre 732, Project D11).

References

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL Recognising Textual Entailment Challenge. In *Proceedings of the First International Conference on Machine Learning Challenges: Evaluating Predictive Uncertainty Visual Object Classification, and Recognizing Textual Entailment*. Springer-Verlag, Berlin, Heidelberg, MLCW'05.

Chris Dyer. 2009. Using a Maximum Entropy Model to Build Segmentation Lattices for MT. In *Proceedings of NAACL-HLT 2009*. NAACL '09.

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*.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The Third PASCAL Recognizing Textual Entailment Challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*. Stroudsburg, PA, USA, RTE '07.

Birgit Hamp and Helmut Feldweg. 1997. GermaNet - a Lexical-Semantic Net for German. In *Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*.

Verena Henrich and Erhard Hinrichs. 2010. GernEdiT - The GermaNet Editing Tool. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente

Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*. European Language Resources Association (ELRA), Valletta, Malta.

Philipp Koehn and Kevin Knight. 2003. Empirical Methods for Compound Splitting. In *EACL*.

Wessel Kraaij and Renée Pohlmann. 1998. Comparing the Effect of Syntactic vs. Statistical Phrase Index Strategies for Dutch. In *Proceedings ECDL'98*.

Martha Larson, Daniel Willett, Joachim Köhler, and Gerhard Rigoll. 2000. Compound splitting and lexical unit recombination for improved performance of a speech recognition system for German parliamentary speeches. In *Sixth International Conference on Spoken Language Processing, ICSLP / INTERSPEECH*.

Bernardo Magnini, Roberto Zanoli, Ido Dagan, Kathrin Eichler, Günter Neumann, Tae-Gil Noh, Sebastian Pado, Asher Stern, and Omer Levy. 2014. The Excitement Open Platform for Textual Inferences. In *Proceedings of the ACL 2014 System Demonstrations*.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika* 12(2).

Sonja Nießen and Hermann Ney. 2000. Improving SMT quality with morpho-syntactic analysis. In *COLING 2000*.

Tae-Gil Noh, Sebastian Padó, Vered Shwartz, Ido Dagan, Vivi Nastase, Kathrin Eichler, Lili Kotlerman, and Meni Adler. 2015. Multi-Level Alignments As An Extensible Representation Basis for Textual Entailment Algorithms. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics, *SEM 2015*. Denver, Colorado, USA.

Sebastian Padó, Tae-Gil Noh, Asher Stern, Rui Wang, and Roberto Zanoli. 2015. Design and Realization of a Modular Architecture for Textual Entailment. *Natural Language Engineering* 21(2).

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

Helmut Schmid. 1995. Improvements In Part-of-Speech Tagging With an Application To German. In *In Proceedings of the ACL SIGDAT-Workshop*.

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

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*.

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

Britta Dorothee Zeller. 2016. *Induction, Semantic Validation and Evaluation of a Derivational Morphology Lexicon for German*. Ph.D. thesis, Heidelberg, Germany.

Britta Dorothee Zeller, Jan Snajder, and Sebastian Padó. 2013. DErivBase: Inducing and Evaluating a Derivational Morphology Resource for German. In *ACL (1)*. The Association for Computer Linguistics.

Patrick Ziering and Lonneke van der Plas. 2014. What good are 'Nominalkomposita' for 'noun compounds': Multilingual Extraction and Structure Analysis of Nominal Compositions using Linguistic Restrictors. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*.

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