Compound or Term Features? Analyzing Salience in Predicting the Difficulty of German Noun Compounds across Domains

Anna Hätty[1,2], Julia Bettinger[2], Michael Dorna[1], Jonas Kuhn[2], Sabine Schulte im Walde[2]
[1]Robert Bosch GmbH, Corporate Research, Renningen, Germany
[2]Institute for Natural Language Processing, University of Stuttgart, Germany
{anna.haetty, michael.dorna}@de.bosch.com,
{julia.bettinger,jonas.kuhn,schulte}@ims.uni-stuttgart.de

Abstract

Predicting the difficulty of domain-specific vocabulary is an important task towards a better understanding of a domain, and to enhance the communication between lay people and experts. We investigate German closed noun compounds and focus on the interaction of compound-based lexical features (such as frequency and productivity) and terminology-based features (contrasting domain-specific and general language) across word representations and classifiers. Our prediction experiments complement insights from classification using (a) manually designed features to characterise termhood and compound formation and (b) compound and constituent word embeddings. We find that for a broad binary distinction into *easy* vs. *difficult* general-language compound frequency is sufficient, but for a more fine-grained four-class distinction it is crucial to include contrastive termhood features and compound and constituent features.

1 Introduction

In times of a constant growth of domain-specific data, it is more important than ever to analyse characteristics of domain-specific vocabulary. Domains are topically restricted subject fields containing domain-specific vocabulary that encode domain knowledge. The more technical the terminology in the domain vocabulary, the more difficult it is perceived by lay people unfamiliar with the domain. Predicting the difficulty of domain-specific vocabulary is therefore an important task for enhancing the communication between lays and experts. A prominent example in this respect is the medical domain, where the prediction of difficulty of medical terms can enhance the communication between doctors and patients, e.g. by simplifying medical texts (Abrahamsson et al., 2014; Grabar and Hamon, 2014; Wandji Tchami and Grabar, 2014). While the medical domain represents a well-researched focus, the problem of miscommunication appears across domains.

Previous research on automatic term difficulty prediction already explored a large number of parameters, but as to our knowledge there is yet no study that investigated how difficulty can be attributed to complex phrase formation processes (a language phenomenon) in interaction with domain specialization (a domain phenomenon). The current study investigates these aspects, goes beyond domain peculiarities (such as Latin words in the medical domain), and performs analyses across three rather different domains: *Cooking, DIY ('do-it-yourself')* and *Automotive*.

While we choose a diverse set of domains, we otherwise focus on a special phenomenon within domain-specific vocabulary: German closed compounds. Closed compounds are complex expressions that consist of several lexemes and are written in a single string of characters. An example is *Bremsflüssigkeit* 'brake fluid', which is composed of the two simple words *Bremse* 'brake' and *Flüssigkeit* 'fluid'. By focusing on closed compounds, the boundaries of the phrases to pre-extract in text are unambiguous, and feature analysis will not be biased by how the extraction method is designed. Furthermore, closed compounds are a frequent phenomenon in German: Baroni et al. (2002) found that 47% of the word types in a general-language corpus in German are compounds, and according to Clouet and Daille (2014) compounding is even more productive in specialized domains. The interaction of domain features and lexical features can be easily demonstrated at the examples of closed compounds: For example, the compound *Hydraulikleitung* 'hydraulic line' is considered difficult because it contains the rather technical constituent 'hydraulic'. In contrast, the compound *Blaukochen* (lit: 'blue boiling', a special kind of boiling fish by adding acid) only contains con-

Proceedings of the 10th Conference on Lexical and Computational Semantics, pages 252–262
August 5–6, 2021, Bangkok, Thailand (online)

stituents that are well-known to lay people but is nevertheless difficult for them because the compound is not semantically transparent regarding its constituent 'blue', i.e. it is not obvious what the constituent contributes to the meaning of the compound. In sum, the difficulty of a compound cannot be derived from only compound attributes; in addition, it is influenced by the role and properties of the constituents.

In this study, we want to empirically investigate how phrase formation and domain-specific termhood[1] attributes interact in the automatic prediction of compound difficulty. In order to train predictive models, we use a German compound dataset with a total of 1,030 compounds across the above-mentioned three domains. Based on two settings of the gold standard dataset (a four-class and a binary version) we apply a decision tree classifier using manually designed features to characterize termhood and compound formation, and neural classifiers using word embeddings.

2 Related Work

Term difficulty prediction (also referred to as term familiarity or term technicality prediction) can be seen as a subtask of automatic term extraction. For automatic term extraction, a major strand of methodologies are contrastive techniques, where a term candidate's distribution in a domain-specific text corpus is compared to the distribution in a reference corpus, for example a general-language corpus (Ahmad et al., 1994; Rayson and Garside, 2000; Drouin, 2003; Kit and Liu, 2008; Bonin et al., 2010; Kochetkova, 2015; Lopes et al., 2016; Mykowiecka et al., 2018, i.a.). Many term difficulty prediction studies rely on some variant of contrastive approaches, mostly frequency-based; notable exceptions are Zeng-Treitler et al. (2008), who apply a contextual network, and Bouamor et al. (2016), who use a likelihood ratio test based on two language models. Most studies fall into the medical, biomedical or health domain. They rely on classical readability features such as frequency, term length, syllable count, the Dale-Chall readability formula or affixes (Zeng et al., 2005; Zeng-Treitler et al., 2008; Vydiswaran et al., 2014; Grabar et al., 2014). Some features are tailored to the medical domain, for example relying on neo-classical word components, since medical terminology is considered to be highly influenced by Greek and Latin (Deléger and Zweigenbaum, 2009; Bouamor et al., 2016).

As to our knowledge, there is no previous work that investigated term difficulty prediction for complex phrases. Regarding the more general task of automatic term extraction, a few studies included complex phrases and their constituents. For example, the *C-value* (Frantzi et al., 1998) combines linguistic and statistical information and takes nested terms into account for evaluating termhood. The *FGM score* (Nakagawa and Mori, 2003) relies on the geometric mean of the number of distinct left and right neighboring words for each constituent in a complex term. *Contrastive Selection via Heads (CSvH)* (Basili et al., 2001) is a corpora-comparing measure that computes termhood for a complex term by biasing the termhood score with the general-language frequency of the head. Hätty et al. (2017) combine termhood measures within a random forest classifier to extract single and multiword terms and apply the measures recursively to the components. Hätty and Schulte im Walde (2018) demonstrate that propagating constituent information through neural networks improves the prediction of compound termhood.

3 Data

3.1 German Closed Noun Compounds

Closed compounds are complex expressions that consist of several lexemes and that are written in a single string of characters. The lexemes are called constituents. The constituents of a two-part compound can be divided into *modifier* and *head*, where the latter is word-final in German.

An important empirical compound attribute is the morphological family size (De Jong et al., 2000) of a lexeme, which we refer to as *productivity* henceforth. Morphological family size is defined as the type count of morphological family members, which comprise compounds and derived words that contain the given lexeme as a constituent. We distinguish between two kinds of productivity as a compound attribute: The productivity of a modifier refers to the number of compound types where a certain word type occupies the position of the modifier, and the productivity of a head refers to the number of compound types where a certain word type occupies the position of the head.

[1] Termhood refers to the degree to which a lexical unit can be considered a domain-specific concept (Kageura and Umino, 1996).

3.2 Corpora

As corpus for general language, we rely on the *SdeWaC* (Faaß and Eckart, 2013), a cleaned version of the web-crawled corpus *deWaC* (Baroni et al., 2009), containing ≈ 880 million words.

As domain-specific corpora, we use the three domain corpora that are described by Bettinger et al. (2020). The corpora were crawled for the domains of *Cooking, DIY* and *Automotive*. They were selected to include a variety of different domains; for example, the *Automotive* domain was chosen because it was expected to be more technical than the *Cooking* domain. The domain corpora consist of both user-generated and expert content. User-generated content was extracted from Wikipedia, `wikihow.de` and `wikibooks.de`, filtered by domain-related categories. Further, domain-specific homepages such as `kochwiki.org` were crawled. Expert texts include tool manuals and books (e.g. on *Automotive* and on *Handicraft*), as well as redacted text crawled from homepages such as `1-2-do.com`. Finally, all corpora were reduced to the size of the smallest corpus and are equally-sized with 5.6 million tokens. The texts are tokenized, lemmatized and POS-tagged with spaCy[2].

3.3 Gold Standard

We rely on the domain-specific compound difficulty gold standard developed on the basis of the just-described domain-specific corpora (Bettinger et al., 2020). The gold standard contains 1,030 closed compounds from the domains of *Cooking, DIY* and *Automotive*. Compounds were automatically identified in text by applying the Simple Compound Splitter (Weller-Di Marco, 2017). All compounds with a frequency smaller than three were excluded, which resulted in a pool of 12,400 *Cooking* compounds, 16,935 *DIY* compounds and 20,468 *Automotive* compounds, A subset was selected which was balanced for the following features: frequency of the compound, productivity of the head, productivity of the modifier and frequency of the head. The final dataset was rated by 26 annotators on a Likert-like difficulty scale (Likert, 1932) from 1 (easy; the term does not require specialized knowledge to be understood) to 4 (difficult; the term requires specialized knowledge). After the annotation process, the 20 annotations were selected where annotators agreed most. The average pairwise Spearman's ρ correlations of the 20 annotators is 0.61.

We base our models on two specifications of the gold standard:

four-class: For each compound, we calculate the median.[3] In case of being between values, we decide for the upper median (i.e. if the value is .5, it is rounded up).

binary: We simplify the annotation and break down the four graded classes into two broader classes: *easy* and *difficult*. We decide to cluster classes 2, 3 and 4 into a new class 'difficult' and keep class 1 as 'easy' for the following reasons: Annotators agreed most for class 1, so this is by far the biggest class. Our binary grouping balances the class sizes more equally and we believe that annotators can easily recognize when they find a compound to be easy (because they fully understand it, which is why we get such a good agreement), but when it comes to specifying difficulty they have more problems to express to which degree they do not understand the compound (due to the fact that they cannot know how much they do not understand).

Figure 1 presents the binary and four-class distributions across the three gold standards. The graphs show that there are more difficult compounds in *Automotive* than in *Cooking* and *DIY*.

4 Experiments on Predicting Difficulty

Our prediction experiments investigate and complement insights from decision tree classification using manually designed features to characterise termhood and compound formation (section 4.1), and logistic regression (LR) and multilayer perceptron (MLP) classification using compound and constituent word embeddings (section 4.2).

For evaluation, we use 5-fold cross-validation and Micro- and Macro-F1 score. As a comparison to the model results, we apply a majority-class baseline. When testing for significance, we use the McNemar's significance test (McNemar, 1947), a paired non-parametric statistical hypothesis test.

4.1 Classification with Term and Compound Features

A core research question for the classification experiments is to which degree attributes that are

[2] `https://spacy.io/`

[3] Alternatively, one could calculate the mean compound difficulty values, but the means are more sensitive to outliers, and in our dataset therefore less appropriate.

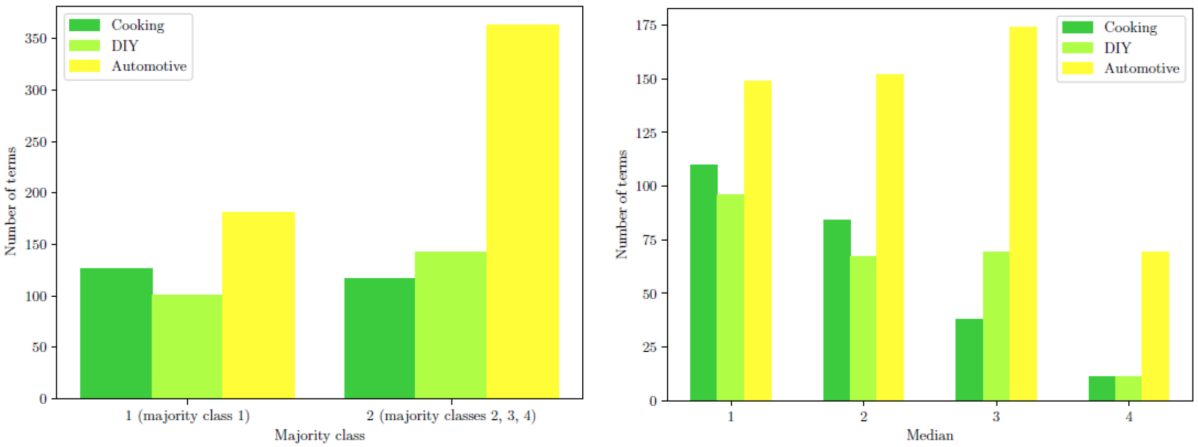

Figure 1: Gold standard: binary and four-class distributions across gold standards (figures taken from Bettinger et al. (2020)).

related to compoundhood influence the prediction, in contrast and in combination with attributes that are related to termhood. The feature types tailored to represent these attributes are the following:

- COMPOUNDHOOD (C) FEATURES[4]: frequencies and productivities of compounds, heads and modifiers in the general-language and the domain-specific corpora; cosine distances between compound modifier and compound head embeddings
- TERMHOOD (T) FEATURES: contrastive measures *Weirdness Ratio* (Ahmad et al., 1994), *TFITF – Term Frequency Inverse Term Frequency* (Bonin et al., 2010), and *CSvH – Contrastive Selection via Heads* (Basili et al., 2001)
- COMBINED C+T FEATURE: *FGM-Score*, a termhood measure that combines compound and termhood attributes (Nakagawa and Mori, 2003)

Note that we decided against a direct computation of compound–constituent compositionality (Reddy et al., 2011; Schulte im Walde et al., 2013, 2016) as a feature, because the compound dataset was balanced for frequency. It includes infrequent compounds for which word embeddings and compositionality measures would be imprecise.

Method: Decision Trees. Decision tree classifiers (DTs) are supervised machine learning methods that are represented as tree structures. DTs were chosen for this task because they are easy to interpret. We identify the optimal tree depth of our decision trees by constantly growing the trees until results decrease, with relying on Gini impurity as the branch splitting criterium. In this way we found an optimal depth of three for the decision tree in the binary task, and an optimal depth of five for the decision tree in the four-class task.

Overall results. Table 1 shows the results for the decision tree classification using all features. The classification models significantly outperform the respective baselines in the binary classification tasks, but in the four-class distinctions this only applies to the *Automotive* domain and across all domains (non-significant results are in italics). For the binary task, the results for *Automotive* are better than for *Cooking* and *DIY*. We assume that this divergence is due to a higher imbalance of class sizes across the domains, cf. figure 1.

Results by feature group. Having looked at the results when using all features at the same time, we now use coherent groups of features:

1. ***Domain-specific corpus-related features:*** frequencies of compounds, heads and modifiers; productivities of heads and modifiers; FGM-Score
2. ***General-language corpus-related features:*** frequencies of compounds, heads and modifiers; productivity of heads and modifiers; FGM-Score
3. ***Contrastive features:*** weirdness scores and TFITFs of compounds, heads and modifiers; CSvH
4. ***Cosine distance features:*** cosine scores of word2vec and fastText constituent vectors

[4]Note that for all but one of these features we have a balanced set of compounds in the gold standard, see section 3.3.

Baselines and Gold Standards	Binary		Four-class	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1
Baseline *Cooking*	0.519	0.342	0.498	0.166
Baseline *DIY*	0.584	0.369	0.407	0.145
Baseline *Automotive*	0.667	0.400	0.325	0.123
Baseline All	0.604	0.377	0.376	0.137
Cooking	0.646	0.631	*0.543*	*0.312*
DIY	0.712	0.684	*0.519*	*0.406*
Automotive	0.750	0.720	0.471	0.286
All	**0.732**	**0.707**	**0.492**	**0.405**

Table 1: Results for classification using all features. All results but those in italics are significant.

Feature Group	Micro-F1	Macro-F1
Baseline	0.604	0.377
Cosine	0.594*	0.391*
Head	0.608*	0.568*
Domain	0.635*	0.593*
Modifier	0.656	0.627
Constituent	0.661	0.648
Contrastive	0.713	0.690
All	**0.732**	**0.707**
General	0.735	0.703
Compound	0.736	0.713

Table 2: Binary: results by feature groups.

Feature Group	Micro-F1	Macro-F1
Baseline	0.376	0.137
Cosine	0.400*	0.258*
Domain	0.405*	0.300*
Head	0.418	0.287
Constituent	0.455	0.364
Modifier	0.457	0.370
General	0.458	0.359
Compound	0.480	0.342
All	**0.492**	**0.405**
Contrastive	0.510	0.408

Table 3: Four-class: results by feature group.

Feature	Micro-F1	Macro-F1
Baseline	0.604	0.377
comp_TFITF	0.637	0.566
FREQ_head_gen	0.642	0.571
FREQ_mod_gen	0.645	0.619
PROD_mod_gen	0.653	0.616
comp_WEIRD	0.709	0.690
FGM_gen	0.713	0.696
FREQ_gen	0.732	0.706

Table 4: Binary: individual features which significantly outperform the baseline.

Feature	Micro-F1	Macro-F1
Baseline	0.376	0.137
comp_TFITF	0.412	0.238
FREQ_mod_dom	0.415	0.280
Num_comp	0.417	0.248
PROD_head_gen	0.426	0.306
FREQ_head_gen	0.435	0.290
FREQ_mod_gen	0.454	0.322
PROD_mod_gen	0.455	0.298
comp_WEIRD	0.462	0.330
FREQ_gen	0.464	0.343
FGM_gen	0.467	0.339

Table 5: Four-class: individual features which significantly outperform the baseline.

5. ***Compound features:***
   compound frequencies in general-language and domain-specific corpora; numbers of compound constituents; weirdness scores and TFITFs of compounds
6. ***Modifier features:***
   frequencies and productivities of modifiers in general-language and domain-specific corpora; weirdness scores and TFITFs of modifiers; CSvH
7. ***Head features:***
   frequencies and productivities of heads in general-language and domain-specific corpora; weirdness scores and TFITFs of heads; CSvH
8. ***Constituent features:***
   union of modifier and head features

Tables 2 and 3 show the results obtained by feature group, sorted by increase in Micro-F1. We

Chosen Feature	Micro-F1	Macro-F1
+FREQ_gen	0.732	0.706
+PROD_mod_dom	0.739	0.720
+PROD_mod_gen	0.744	0.725
+mod_WEIRD	0.746	0.727
+FREQ_dom	0.746	0.727

Table 6: Binary: feature selection.

Chosen Feature	Micro-F1	Macro-F1
+FGM_gen	0.467	0.339
+head_TFITF	0.487	0.350
+PROD_mod_gen	0.493	0.362
+PROD_head_gen	0.511	0.370
+NUM_comp	0.511	0.370

Table 7: Four-class: feature selection.

can see that most feature groups achieve lower results in comparison to using all features (in bold font), but at the same time 'All' does not achieve the best results. The categories Cosine, Domain and Head perform worst and do in most cases not even significantly improve over the baseline. The modifier features are better than the head features, which is in line with the results in (Hätty et al., 2017) where the modifier features are more important for detecting termhood than head features. For both the binary and the four-class tasks, the groups General, Compound and Contrastive perform best, with Compound as the winner for the binary task and Contrastive as the winner for the four-class task. The arrows in the result tables indicate which group results are significantly different to the winner group result.

Individual features. Tables 4 and 5 show the results for those individual features which perform significantly better than the respective baseline, sorted by increase in F1. For the four-class task, three more features perform significantly better than the baseline in comparison to the binary task; these features are marked in bold. The best individual features are the same for both tasks, with almost the same rankings. The best three individual features address distinct attributes of a compound term: a compound's general-language frequency (FREQ_gen), a termhood measure involving constituents (FGM_gen), and a contrastive termhood measure (comp_WEIRD).

Best feature combination. Tables 6 and 7 analyze how features interact: We perform feature selection by repeatedly adding the best-performing individual feature for each task, based on Micro-F1, until the scores stagnate or decrease. The resulting best feature combinations provide us with the best results for each task, while only comprising five individual feature types in both tables. The optimal combinations address attributes of the whole compounds and attributes of constituents.

Analyzing frequency and productivity. For investigating the influence of frequency and productivity properties of compounds and constituents, we created subsets of the gold standard where we distinguished between tertiles regarding compound frequency and constituent productivity: 'low', 'mid' and 'high'. Each property type is assessed once for the general-language and once for the domain-specific language. The 6×3 tertiles are determined by sorting all elements regarding one property and cutting the data into three equally-sized portions. The resulting ranges are shown in table 8.

We then compare the classifier results for the two extreme tertiles, 'low' and 'high', using all features on these subsets. The results are shown in the right-hand part of table 8. It is obvious that across all properties better results are achieved for the 'low'-category, as indicated by the bold font. The gap between the results for 'low' and 'high' is especially large for the productivities of modifiers and heads. Thus low productivity represents a rather clear indicator for a compound to be either easy or difficult (given that the model achieves better results in the prediction), while high productivity is an attribute of harder to distinguish easy and difficult terms. In order to investigate this effect further, we inspect the gold label distribution in the 'low' and 'high'-categories. We find a dominance of difficult compounds in the 'low'-categories, while there is a higher balance between easy and difficult compounds in the 'high'-categories. This shows that low productivity and frequency are indicators of difficulty, while high productivity and frequency are less distinctive.

4.2 Classification with Word Embeddings

For our second kind of classification experiments, we do not use hand-crafted features anymore but semantic representations of compounds and components for general-language and domain. Two kinds of word embeddings are used in the follow-

Compound and Constituent Properties	Tertiles and Ranges			Micro-F1	
	low	mid	high	low	high
compound frequency (domain)	3–4	4–8	8–444	**0.773**	0.722
compound frequency (general)	0	0–17	17–53,569	**0.779**	0.722
modifier productivity (domain)	1–14	14–55	55–665	**0.863**	0.658
modifier productivity (general)	0–101	103–588	590–4,976	**0.884**	0.661
head productivity (domain)	1–14	14–61	62–1,157	**0.802**	0.652
head productivity (general)	0–119	119–786	786–8,293	**0.812**	0.693

Table 8: Ranges of selected properties across tertiles, and results on binary classification for extreme 'low' and 'high' tertiles when using all features (cf. **All** in Table 2 with Micro-F1=**0.732**).

ing: *word2vec* (Mikolov et al., 2013) and *fastText* (Bojanowski et al., 2017).[5]

We use the word2vec model, because it is a standard model for natural language processing applications. The fastText model works on character n-grams and not on words, and Bojanowski et al. (2017) argues that it performs well on closed compounds. This model is particularly interesting for us because a compound embedding is learned partially from the same n-grams as the embeddings of its constituents. Thus, we implicitly have a representation of the constituents in the compound embedding, which we expect to be beneficial for our classification task. Inspecting some words and their nearest neighbors for the two models confirms our intuition. For the verb *kochen* ("cook") the following six words are the most similar according to word2vec: *sieden* ("to boil"), *garen* ("to refine"), *brutzeln* ("to sizzle"), *braten* ("to fry"), *grillen* ("to barbecue") and *zubereiten* ("to prepare"). According to fastText we find the nearest neighbors *erkochen* ("to reach by cooking"), *garkochen* ("to cook sth. well"), *teekochen*[6] ("to make tea"), *reiskochen* ("to cook rice"), *eierkochen* ("to cook eggs") and *bekochen* ("to cook for someone"). The similarity in word2vec neighbors is more on the semantic level in contrast to fastText, where the words are highly similar on a surface morphological level. The embeddings are trained for each domain individually, by concatenating SdeWaC and the respective domain data as input.

Methods: LR and MLP We use our pre-trained word embeddings for compounds and constituents as features and apply two kinds of classifiers:

- *logistic regression*: simple neural network with only input and output layers but no hidden layer,
- *multilayer perceptron*: neural network with each one input, hidden and output layer.

For the binary classification task, the classifiers use a sigmoid activation in the output layer, for the four-class task the classifiers use softmax activation. For the multilayer perceptron, we also use a sigmoid activation for the hidden layer. Concerning the parameters, the batch size is set to 32, there are 50 epochs and the hidden layer has a dimension of 64.

Results. We compare three different input settings for the classification tasks: The first model only takes the compound word embeddings as input (see 'compound' in table 9). For all settings, we distinguish between two differently trained word embeddings: the word-based word2vec and the character-based fastText word embedding models. The second model ('comp+const') takes the concatenated embeddings of the compound and of its constituents (binary split, i.e. two constituents) as input, to evaluate the impact of the constituents. The third model ('only const') only uses the concatenated constituent vectors, to evaluate if this information is competitive.

The results for the classifications are shown in table 9. For the binary task we reach the best results (marked in bold) with word2vec when using a combination of compound and constituent information, and with fastText when only using the compound embeddings. This tendency was expected: Since fastText embeddings are character-based, the constituents are implicitly encoded as well. Using only constituent information provides lower result scores in comparison to using compound information, which is in line with the results of the previous section.

[5] We do not use state-of-the-art contextualized word embeddings such as BERT (Devlin et al., 2019), because we predict difficulty on a type-based, not context-dependent level.

[6] We cite words in their original lowercased version as used in the model.

The distribution of the results of the four-class task in table 9 is similar to the binary task, except for now also for fastText the combination of compound and constituent information works best. This might be caused by the more difficult task and is also indicated by the fact that for the four-class task MLP with the additional hidden layer produces the best results, while for the binary task the simpler model LR obtains the best results.

Interestingly, word2vec models mostly perform better than fastText models, although fastText implicitly contains constituent information. We argue that because 171 infrequent compound vectors are missing for word2vec (with a minimum frequency threshold for word vectors to be trained), these 171 compounds are assigned to the same random vector. Given that low frequency is a reasonable indicator for difficulty, the model might learn from the missing vectors which compounds are infrequent.

Although models using both compound and constituent information seem to be superior to models using only compound information, these results can only be treated as a tendency. For word2vec and both the binary and the four-class tasks, models using both compound and constituent embeddings are not significantly better than models using only compound embeddings. However, although models using compound embeddings perform significantly better than models using only constituent embeddings (which is intuitive), the latter still perform significantly better than the baseline. This shows that constituent embeddings carry informative characteristics for classifying compounds for difficulty.

4.3 Discussion

Our experiments investigated how compound formation and termhood and domain attributes influence the prediction of compound difficulty.

Compounds and constituents. The binary task, as the presumably simpler task, reached better results with simpler means: General-language frequency of the compound is a good indicator (2% better than the second-best feature for Micro-F1); in addition, there is a 5% gap between compound and constituent features (table 4), which shows that compound features are sufficient for this task. For the four-class task, features differ less; the best results include compound and constituent information (table 5). However for both tasks we can see: a combination of compound and constituent features leads to best results (tables 6 and 7).

The experiments with using neural networks show the same tendency (table 9): While for half of the cases in the binary task the compound vector is sufficient, the improvement over 'comp+const' is not significant, and overall using both compound and constituent vectors ('comp+const') provides the best results. We conclude that constituents influence the degree of difficulty of the compounds.

Termhood. Contrastive features (i.e. termhood features) are more important for the four-class task than for the binary task (tables 2 and 3): For the four-class task, they perform significantly better than the general-language features, while for the binary task 'FREQ_gen' is the best individual feature (table 4). In sum, for a broad difficulty distinction as for the binary task, general-language information might be sufficient, but for the more fine-grained four-class task contrastive termhood features are supportive.

Domains. There are no striking differences in the predictive power of the models across domains (table 1). For all three gold standards, the binary classification models outperform the respective baselines. In the four-class distinction, this is only the case for *Automotive*, which includes more difficult compounds than *Cooking* and *DIY*. Presumably, prediction differences are due to the differently (im)balanced sizes of the classes.

Low versus high productivity and frequency. When contrasting the lower and upper tertile value ranges for compound frequency and constituent productivity, we found that low productivity and low frequency are very salient indicators for the level of difficulty. This seems counterintuitive: e.g. high frequency could be a reliable indicator for simplicity of a compound, while low frequency could indicate difficulty, but low frequency could also indicate that concepts are newly coined (which does not mean that they are difficult), or because of spelling or inflection errors. The dataset was cleaned for the latter, but the former case was not paid attention to. Concerning productivity, the gap between 'high' and 'low' is even more extreme. We hypothesize that this could be due to a compound being judged as difficult because of one difficult constituent, but an easy compound requires all constituents to be easy. This is why single easy constituents might be no good indicators – difficulty depends on the other constituent for the compound to be easy or difficult.

	network	word2vec		fastText		word2vec		fastText	
		Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1
compound	LR	0.760	0.722	**0.746**	**0.724**	0.514	0.385	0.459	0.338
	MLP	0.761	0.729	0.738	0.720	0.518	0.383	0.469	0.341
comp+const	LR	**0.771**	**0.758**	0.734	0.715	0.515	0.429	0.465	0.355
	MLP	0.749	0.735	0.732	0.716	**0.525**	**0.431**	**0.477**	**0.369**
only const	LR	0.701	0.685	0.703	0.679	0.460	0.362	0.447	0.355
	MLP	0.714	0.697	0.713	0.696	0.493	0.389	0.469	0.365

Table 9: LR/MLP Classifiers: Mi(cro)-F1 and Ma(cro)-F1 results for the Binary (left) and Four-Class (right) task.

5 Conclusion

This study investigated the automatic prediction of difficulty for domain-specific German compounds across three domains. We asked to what extent compound formation attributes and domain-specific termhood attributes influence and interact in the prediction. We found that plain general-language compound frequency is a reliable indicator for difficulty in our dataset, which shows that effects of domain-specialization and compound formation are reflected to a large extent by general corpus frequency. However, for a more fine-grained four-class distinction of difficulty going beyond a broad binary distinction into 'easy' and 'difficult', contrastive termhood features and compound and constituent information are crucial.

References

Emil Abrahamsson, Timothy Forni, Maria Skeppstedt, and Maria Kvist. 2014. Medical text simplification using synonym replacement: Adapting assessment of word difficulty to a compounding language. In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations*, pages 57–65, Gothenburg, Sweden.

Khurshid Ahmad, Andrea Davies, Heather Fulford, and Margaret Rogers. 1994. What is a term? The semi-automatic extraction of terms from text. *Translation Studies: An Interdiscipline. Selected papers from the Translation Studies Congress, Vienna, 1992*, 2:267––278.

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The WaCky wide web: A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.

Marco Baroni, Johannes Matiasek, and Harald Trost. 2002. Predicting the components of German nominal compounds. In *Proceedings of the 15th European Conference on Artificial Intelligence*, pages 470–474, Lyon, France.

Roberto Basili, Maria T. Pazienza, Alessandro Moschitti, and Fabio M. Zanzotto. 2001. A contrastive approach to term extraction. In *Proceedings of the 4th Terminology and Artificial Intelligence Conference*, Nancy, France.

Julia Bettinger, Anna Hätty, Michael Dorna, and Sabine Schulte im Walde. 2020. A domain-specific dataset of difficulty ratings for German noun compounds in the domains DIY, Cooking and Automotive. In *Proceedings of the 12th International Conference on Language Resources and Evaluation*, pages 4352–4360, Marseille, France.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Francesca Bonin, Felice Dell'Orletta, Giulia Venturi, and Simonetta Montemagni. 2010. A contrastive approach to multi-word term extraction from domain corpora. In *Proceedings of the 7th International Conference on Language Resources and Evaluation*, pages 19—21, Valletta, Malta.

Dhouha Bouamor, Leonardo Campillos Llanos, Anne-Laure Ligozat, Sophie Rosset, and Pierre Zweigenbaum. 2016. Transfer-based learning-to-rank assessment of medical term technicality. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, Portorož, Slovenia.

Elizaveta Loginova Clouet and Béatrice Daille. 2014. Splitting of compound terms in non-prototypical compounding languages. In *Proceedings of the First Workshop on Computational Approaches to Compound Analysis*, pages 11–19, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Nivja H. De Jong, Robert Schreuder, and Harald R. Baayen. 2000. The morphological family size effect and morphology. *Language and Cognitive Processes*, 15(4–5):329–365.

Louise Deléger and Pierre Zweigenbaum. 2009. Extracting lay paraphrases of specialized expressions from monolingual comparable medical corpora. In *Proceedings of the 2nd Workshop on Building and Using Comparable Corpora: from Parallel to Non-parallel Corpora*, pages 2–10.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of

Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, Minneapolis, Minnesota, USA.

Patrick Drouin. 2003. Term extraction using non-technical corpora as a point of leverage. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 9(1):99–115.

Gertrud Faaß and Kerstin Eckart. 2013. SdeWaC – A corpus of parsable sentences from the web. In Iryna Gurevych, Chris Biemann, and Torsten Zesch, editors, *Language Processing and Knowledge in the Web*, volume 8105 of *Lecture Notes in Computer Science*, pages 61–68. Springer, Berlin Heidelberg.

Katerina T. Frantzi, Sophia Ananiadou, and Jun-ichi Tsujii. 1998. The c-value/nc-value method of automatic recognition for multi-word terms. In *Proceedings of the 2nd European Conference on Research and Advanced Technology for Digital Libraries*, pages 585–604, London, UK.

Natalia Grabar and Thierry Hamon. 2014. Unsupervised method for the acquisition of general language paraphrases for medical compounds. In *Proceedings of the 4th International Workshop on Computational Terminology*, pages 94–103, Dublin, Ireland.

Natalia Grabar, Thierry Hamon, and Dany Amiot. 2014. Automatic diagnosis of understanding of medical words. In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations*, pages 11–20, Gothenburg, Sweden.

Anna Hätty, Michael Dorna, and Sabine Schulte im Walde. 2017. Evaluating the reliability and interaction of recursively used feature classes for terminology extraction. In *Proceedings of the Student Research Workshop at the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 113–121, Valencia, Spain.

Anna Hätty and Sabine Schulte im Walde. 2018. Fine-grained Termhood Prediction for German Compound Terms using Neural Networks. In *Proceedings of the COLING Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions*, pages 62–73, Santa Fe, NM, USA.

Kyo Kageura and Bin Umino. 1996. Methods of automatic term recognition: A review. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 3(2):259–289.

Chunyu Kit and Xiaoyue Liu. 2008. Measuring mono-word termhood by rank difference via corpus comparison. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 14(2):204–229.

Natalia A. Kochetkova. 2015. A method for extracting technical terms using the modified weirdness measure. *Automatic Documentation and Mathematical Linguistics*, 49(3):89–95.

Rensis Likert. 1932. A technique for the measurement of attitudes. *Archives of psychology*.

Lucelene Lopes, Paulo Fernandes, and Renata Vieira. 2016. Estimating term domain relevance through term frequency, disjoint corpora frequency-tf-dcf. *Knowledge-Based Systems*, 97:237–249.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Agnieszka Mykowiecka, Małgorzata Marciniak, and Piotr Rychlik. 2018. Recognition of irrelevant phrases in automatically extracted lists of domain terms. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 24(1):66–90.

Hirosi Nakagawa and Tatsunori Mori. 2003. Automatic term recognition based on statistics of compound nouns and their components. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 9(2):201–219.

Paul Rayson and Roger Garside. 2000. Comparing corpora using frequency profiling. In *Proceedings of the Workshop on Comparing Corpora*, pages 1–6, Hong Kong.

Siva Reddy, Diana McCarthy, and Suresh Manandhar. 2011. An empirical study on compositionality in compound nouns. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 210–218, Chiang Mai, Thailand.

Sabine Schulte im Walde, Anna Hätty, and Stefan Bott. 2016. The role of modifier and head properties in predicting the compositionality of English and German noun-noun compounds: A vector-space perspective. In *Proceedings of the 5th Joint Conference on Lexical and Computational Semantics*, pages 148–158, Berlin, Germany.

Sabine Schulte im Walde, Stefan Müller, and Stefan Roller. 2013. Exploring vector space models to predict the compositionality of German noun-noun compounds. In *Proceedings of the 2nd Joint Conference on Lexical and Computational Semantics*, pages 255–265.

V.G-Vinod Vydiswaran, Qiaozhu Mei, David A Hanauer, and Kai Zheng. 2014. Mining consumer health vocabulary from community-generated text. In *AMIA Annual Symposium Proceedings*, pages 1150–1159.

Ornella Wandji Tchami and Natalia Grabar. 2014. Towards automatic distinction between specialized and non-specialized occurrences of verbs in medical corpora. In *Proceedings of the 4th International Workshop on Computational Terminology*, pages 114–124, Dublin, Ireland.

Marion Weller-Di Marco. 2017. Simple compound splitting for German. In *Proceedings of the 13th Workshop on Multiword Expressions*, pages 161–166, Valencia, Spain.

Qing Zeng, Eunjung Kim, Jon Crowell, and Tony Tse. 2005. A text corpora-based estimation of the familiarity of health terminology. *International Symposium on Biological and Medical Data Analysis*, pages 184–192.

Qing Zeng-Treitler, Sergey Goryachev, Tony Tse, Alla Keselman, and Aziz Boxwala. 2008. Estimating consumer familiarity with health terminology: A context-based approach. *Journal of the American Medical Informatics Association*, 15(3):349–356.