Originally published in: Preliminary proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019), October 9 – 11, 2019 at Friedrich-Alexander-Universität Erlangen-Nürnberg. - München [u.a.]: German Society for Computational Linguistics & Language Technology und Friedrich-Alexander-Universität Erlangen-Nürnberg, 2019. Pp. 149-160

Metaphor detection for German Poetry

Ines Reinig
Computational Linguistics
Heidelberg University
wuerz@cl.uni-heidelberg.de

Ines Rehbein
Leibniz ScienceCampus
IDS Mannheim/Heidelberg University
rehbein@ids-mannheim.de

Abstract

This paper presents first steps towards metaphor detection in German poetry, in particular in expressionist poems. We create a dataset with adjective-noun pairs extracted from expressionist poems, manually annotated for metaphoricity. We discuss the annotation process and present models and experiments for metaphor detection where we investigate the impact of context and the domain dependence of the models.

1 Introduction

Metaphors are commonly used to conceptualise all aspects of our social and intellectual lives, thus helping us to make sense of the world around us (Lakoff, 1987, p.6). Therefore, many studies in NLP have addressed the task of metaphor detection for English and other languages, focussing on everyday language use. But metaphors are also an important stylistic device in literary texts, and recently more and more interest in computational methods for metaphor detection comes from the newly emerging area of Computational Humanities (Kesarwani et al., 2017; Tanasescu et al., 2018).

Our work is situated in the context of Computational Literary Studies. We are interested in the use of metaphors as stilistic devices in poetry, in particular in expressionist poems. Expressionism is an art movement originating in Germany at the beginning of the 20th century. In contrast to earlier periods such as Naturalism, expessionist artists focussed on describing the world not according to its physical properties but from a subjective and highly emotional perspective.

> *"Dem Dichter geht es also nicht um eine Darstellung der empirischen Wirklichkeit, sondern darum, wie er sie, nur er sie sieht und wie er möchte, daß sie auch von anderen gesehen werde. Er erarbeitet deshalb eine Metapher, die fähig ist, seine Gestimmtheit auszusprechen und eine gleiche Gestimmtheit hervorzurufen: eine Art magische Formel."*[1] (Dietz, 1959, p.56)

For illustration, consider the following adjective-noun pairs from *Grodek*, a well-known expressionist war poem by Georg Trakl. In *Grodek*, Trakl creates a nightmarish atmosphere by means of colour symbolism, imagery, personification and neologisms, making extensive use of metaphors to express his inner view of reality (example 1).

(1) a. *rotes Gewölk* (red clouds)
b. *schwarze Verwesung* (black decay)
c. *zerbrochene Münder* (broken mouths)
d. *wilde Klage* (wild lament)
e. *schweigender Hain* (silent forest)
f. *mondne Kühle* (lunar coolness)

To be able to do large-scale investigations of metaphors in expressionist poems and to compare the use of metaphors in different literary genres or in the writings of individual authors, we need to be able to automatically detect metaphors in literary text with high precision and recall. This work presents first steps towards this goal. Our contributions can be summarised as follows:

- We create a new corpus with adjective-noun (A-N) pairs from expressionist poems, annotated for metaphoricity.
- We develop a classifier for automatically predicting A-N metaphors in literary texts.
- We investigate the domain dependence of our model by creating a second dataset for German A-N metaphors, based on the translation of the English A-N dataset of (Tsvetkov et al., 2014), extracted from web corpora.

[1] *Engl. translation: "The poet is not interested in a representation of empirical reality, but only in his subjective view of reality, and how he wants it to be seen by others. He therefore develops a metaphor that is capable of expressing his mood and evoking the same mood in others: a kind of magic formula."*

The paper is structured as follows. We first review related work on metaphor detection for English and German (§2). Then we describe the creation of the two datasets (§3) and present our experiments on metaphor detection for German (§4 and §5). We evaluate and discuss our results and outline avenues for future work (§6).

2 Related Work

Extensive research on metaphor detection has been conducted for English. Early approaches rely on lexical resources such as hyponym relations in WordNet and word co-occurrence information (Krish nakumaran and Zhu, 2007). Others have used abstractness ratings for individual words as features (Turney et al., 2011; Tsvetkov et al., 2014). Turney et al. (2011) show that abstractness scores extracted from a word's context is an effective indicator of its metaphoricity. The system in Tsvetkov et al. (2014), which achieves an F-score of 85% on detecting English adjective-noun metaphors, uses imageability scores in addition to abstractness, in combination with WordNet supersenses and word embeddings.

Shutova et al. (2013) create a statistical model that does not depend on lexical knowledge from external knowledge bases but relies on weakly supervised distributional clustering. The more recent work in Rei et al. (2017) also identifies metaphors without the need for handcrafted features: a supervised similarity network uses the semantic information encoded in word embeddings to detect metaphorical relations. Their system is on a par with the work of Tsvetkov et al. (2014).

Only few studies have investigated metaphor detection for German, due to the lack of freely available annotated resources.[2] Köper and Schulte im Walde (2016a) develop a classifier for the identification of metaphorical uses of German particle verbs. Among other features, they use affective ratings for German lemmas (Köper and Schulte im Walde, 2016a) which we also employ in this work. Köper and Schulte im Walde (2017) model word senses for particle verbs and evaluate their model on metaphor detection, among other tasks.

3 Data & Annotation

In the paper, we focus on metaphorical adjective-noun (A-N) pairs and conduct experiments on two datasets: i) one new dataset with A-N metaphors from German expressionist poems (POEMS) and ii) a second dataset based on a translation of the English A-N data of Tsvetkov et al. (2014) (TSV).

3.1 Annotating Metaphors in Poetry

For the first dataset, we extract A-N pairs from expressionist poems and annotate these pairs for metaphoricity. The process of creating and annotating the POEMS dataset is described below.

Dataset creation The poems have been collected from Project Gutenberg[3], Deutsches Textarchiv[4] and from various poetry websites. We extract the raw text and predict lemmas and POS tags using the TreeTagger (Schmid, 1994; Schmid, 1995). Then we extract lemma pairs that consist of an adjective followed by a noun.

In addition, we extract context for each A-N pair. Since the use of punctuation in poems does not always follow standard German grammar and sentence length in poetic texts can strongly vary in length, we choose to extract context information based on a fixed token window. For each A-N pair, we extract at most 10 tokens on the left and at most 10 tokens on the right. This approach generates context that varies only minimally in length.

We also limit the number of context strings extracted for each A-N pair to avoid that high-frequency A-N pairs are overrepresented in our data. For POEMS, we limit the number of context strings per pair to 20 in the training set and 10 in the test set. For the out-of-domain TSV dataset, we use a limit of 129 in the training set and 47 for the test set. These numbers were determined empirically such that i) no pair is overrepresented and ii) the original distributions between metaphorical and literal instances in the data is maintained.

Annotation procedure In the next step, we annotate each A-N pair with one of three labels (*literal*, *metaphorical*, *ambiguous*). The annotators do not see to the instance's context but assign the label *ambiguous* for instances where context is necessary to disambiguate between literal and metaphorical uses. We found that most of the A-N pairs were unambiguous, making this procedure suitable for annotation and, at the same time, speeding up the annotation process by a large margin.

[2] The Hamburg Metaphor DB Project (Lönneker-Rodman, 2008) created a resource for French and (some) German metaphors. Unfortunately, the data is not publicly available.

[3] https://gutenberg.spiegel.de

[4] http://www.deutschestextarchiv.de/

Example (2) shows an ambiguous instance from our corpus where the adjective *heiß* (hot) can refer to high temperatures in a literal sense or, metaphorically, to a subject of interest (hot topic). In such ambiguous cases, human annotators are usually able to determine the intended sense based on context. This was done in a second pass over the data where we presented the annotators with context for the ambiguous instances.

(2) *heißes Feld* (hot field)

Different approaches have been proposed for metaphor annotation. One of them is the Metaphor Identification Procedure (MIP) (Pragglejaz Group, 2007) which first establishes the contextual meaning of a lexical unit, then determines whether a more basic (concrete, precise, older or more related to bodily action) meaning exists. It then marks the unit as metaphorical if the contextual meaning contrasts with the basic meaning while being understandable in comparison with it.

A similar approach by Shutova (2017) uses the same definition of basic meaning but extends the annotation procedure by additionally identifying source and target domains. Shutova (2017) also highlights problems with the concept of *basic meaning*, i.e. the degree of conventionality of metaphors and the partially unsystematic inclusions of word senses in dictionaries make the use of dictionaries problematic for the identification of basic meanings. We encountered the same difficulties in the early stages of annotation when trying to use a dictionary as a reference. In consequence, we choose not to rely on dictionaries during the annotation process but instead extended our guidelines with a categorisation of adjectives and their interpretation (see A.2 in Appendix).

Do Dinh et al. (2018) address the problem of conventionalised metaphors by augmenting an English metaphor corpus with scores for metaphor novelty. They compare different approaches for annotation and show that best-worst scaling[5], while being more time-consuming, yields the highest IAA. Their annotations, however, assume that the metaphors have already been identified.

Our annotation procedure follows previous work by marking A-N pairs as metaphorical if a more basic meaning of the adjective can be found. For example, in *durstiges Kind* (thirsty child), the adjective's meaning used to describe the noun can be considered as basic. In contrast, the adjective's meaning in *durstige Flamme* (thirsty flame) is considered to be different from the basic meaning.

The annotation is performed in several batches by two annotators. After each batch, the annotators discuss difficulties and annotation disagreements to discover grey areas not yet covered in the annotation guidelines, which were continuously improved during the annotation process.

While Tsvetkov et al. (2014) did not use context information in their experiments, we wanted to test the hypothesis that the context is useful for automatically distinguishing metaphors from literal senses. Therefore, after labeling each instance as either *metaphorical, literal* or *ambiguous*, annotators performed an additional annotation step and further annotated ambiguous A-N pairs as either metaphorical or literal by referring to their context. However, both annotators reported that they found this second step difficult because the context often did not provide enough information for disambiguation. Consider the following example:

(3) Er schleudert die mächtigen [...] Kurven umher in der Welt, sie kehren zu ihm zurück, wie dem **dunklen Krieger**, der den Bumerang schnellt.

In the example above, *dunkler Krieger* (dark warrior) was labeled as ambiguous in the first round of annotation since *dunkel* (dark) could refer to colour (e.g. of the warrior's equipment or skin colour) in a literal sense, or to a gloomy or scary appearance in a metaphorical sense. Such ambiguities are characteristic for expressionist poems and again illustrate the use of metaphors as “a magic formula” (Dietz, 1959) to evoke certain emotions in readers.

As only 49 instances had been annotated as ambiguous, we decided to discard all ambiguous A-N pairs from the POEMS corpus, keeping only metaphorical and literal instances. All experiments described in section 5 are conducted on this two-class dataset.[6]

3.2 IAA and Error Analysis

We measured inter-annotator agreement (IAA) for the different batches during annotation. On average, we observe an IAA of 0.62 (Fleiss' κ) and a percentage agreement of 84,9%.

A particular challenge for annotation are adjectives of measurement. Take, for example, *hohe Kosten* (high costs) or *ein langer Tag* (a long day)

[5] In best-worst scaling, annotators select the most novel and the most conventionalised from a set of four metaphors.

[6] We make all annotated data publicly available in form of lists (see `https://github.com/ireinig/metaphor-german-poetry/tree/master/data`).

where it is not clear whether the adjective's basic sense should only refer to physical objects with spatial extensions (length, width, depth, heigth) or also capture other measures such as monetary values or the length of time. During annotation, we discussed these disagreements and extended the guidelines accordingly. For example, in the case of *groß* (big/large), we decided to mark instances as literal when the adjective refers to a quantifiable or classifiable attribute, such as size, surface or intensity, and label all other uses as metaphorically.

While most disagreements concern adjectives of measurement, we did not observe an annotation bias in terms of one annotator choosing a particular class particularly more often than the other. The probability of choosing the literal class varies between 70-80% for both annotators across all batches.

3.3 Translating an English Metaphor Dataset

To investigate the domain dependence of our metaphor detection model, we create a second metaphor corpus based on the English dataset of Tsvetkov et al. (2014). The dataset was created manually using collections of metaphors from the web (training set) and sentences from the TenTen Web corpus (test set). The domains in this dataset range from economics to politics and sports, and are thus crucially different from our POEMS corpus.

We automatically translated the English A-N pairs to German using DeepL[7]. The set of translated instances was then cleaned up by removing i) instances that are not A-N pairs (e.g. English A-N pairs translated to German N-N compounds) and ii) duplicate instances, resulting from the translation of two distinct English instances to the same German expression (e.g. *little chance* and *slim chance* were both translated to *geringe Chance*).

We then lemmatise the translated A-N pairs using the TreeTagger and extract context for each A-N pair from the sDeWaC German Web corpus (Faaß and Eckart, 2013). Table 1 shows the size of the dataset for the original English data (Tsvetkov et al., 2014) and for the translated TSV dataset.

4 Experimental Setup

Training/test split We divide the data into training and test sets by putting all A-N pairs that appeared at least twice in the corpus in the training set while instances occuring only once constitute the test set.[8] This ensures that none of the test instances have been seen during training. Table 1 illustrates the class imbalance in this dataset: approximately 17% (train) and 26% (test) of the instances are metaphorical while the majority class accounts for 83% and 74% of the data.

Lang	Set	Total	metaphorical	literal
		POEMS dataset		
DE	Training	578	100	478
DE	Test	378	98	280
		TSV dataset		
EN	train	1768	884	884
EN	test	200	100	100
DE	train	1149	546	603
DE	test	142	65	77

Table 1: Number of A-N pairs in the German POEMS and the English and German TSV datasets.

4.1 Features

In our experiments, we use the following features that have been shown to be beneficial for metaphor detection in the literature.

Word embeddings are dense vector representations that capture syntactic and semantic properties of words (Turian et al., 2010). Previous work has used embeddings for metaphor detection and reported high scores for baseline models that rely only on word embeddings as features (Tsvetkov et al., 2014; Bulat et al., 2017; Rei et al., 2017; Shutova et al., 2016).

For each A-N pair, we extract embeddings for the adjective and for the noun from the 100-dimensional SkipGram embeddings of Reimers et al. (2014).[9] We average both vectors and obtain one 100-dimensional compositional embedding vector for each A-N pair.

Supersenses Our next feature uses the GermaNet (Hamp and Feldweg, 1997) supersense taxonomy for adjectives and nouns where word senses (and the associated lemma forms) are sorted into semantic fields (e.g. *Menge* (set), *Gesellschaft* (society) or *Koerperfunktion* (bodily functions)).[10]

[7] `https://www.deepl.com/en/translator`

[8] All parameter tuning was done in a cross-validation setup on the training set.

[9] The embeddings are available from `https://www.informatik.tu-darmstadt.de/ukp/research_6/ukp_in_challenges/germeval_2014`.

[10] For the list of supersenses, please refer to `http://www.sfs.uni-tuebingen.de/GermaNet/germanet_structure.shtml#Tops`

Following Tsvetkov et al. (2014), we construct feature vectors by calculating the degrees of membership for noun and adjective supersenses. GermaNet contains 16 distinct semantic fields for adjectives and 23 for nouns. We extract supersense features for each A-N pair as follows. For a given word, we count the number of synsets s it belongs to. Then, for each semantic field f, we count the number of synsets s_f from the set s that are related to f. Finally, for each f, we compute the resulting value by diving s_f by s. We thus obtain vectors of length 16 for adjectives and vectors of length 23 for nouns. The resulting 39-dimensional vector representation is a concatenation of both vectors.

Affective ratings Tsvetkov et al. (2014) and Turney et al. (2011) show that abstractness and imageability scores are useful features for metaphor detection. We use ratings for abstractness, imageability, arousal and valence published by Köper and Schulte im Walde (2016b). The dataset contains ratings for 351,617 German lemmas, in a range of 0 to 10. According to Köper and Schulte im Walde (2016b), *abstractness* characterises anything that cannot be perceived using our senses, as opposed to concreteness; *imageability* refers to words for which one can easily form a mental image; *arousal* refers to the intensity of the emotion linked to a word and *valence* describes whether positive or negative emotions are linked to the word.

We extract affective ratings for each adjective and each noun, resulting in an 8-dimensional vector representation. Words for which no rating is available are assigned the default value of 5.0.

5 Experiments

We investigate the following three hypotheses:

H1 Supersenses, word embeddings and affective ratings are useful features for A-N metaphor detection in German poetry.

H2 Context features extracted from the A-N pair's surrounding text can further improve classification accuracy for metaphor detection.

H3 Metaphors are not domain-dependent but a general cognitive phenomenon, thus supplementary out-of-domain training data can improve results for metaphor detection in poetry.

Setup We train two SVM models on our datasets, POEMS and TSV. For model selection, we perform the following three steps:

1. Algorithm selection and hyperparameter tuning
2. Feature selection for A-N pairs
3. Feature selection for context features

Model and feature selection was done separately for the POEMS and TSV datasets. We refer to the models trained on each dataset as POEMS and TSV.

5.1 Model Selection

Following previous work (Turney et al., 2011; Tsvetkov et al., 2014; Bulat et al., 2017), we experiment with three ML algorithms, i) a Random Forest classifier (Breiman, 2001), ii) a Support Vector Machine (SVM) (Joachims, 1998) and iii) logistic regression (Le Cessie and Van Houwelingen, 1992).[11] Based on 10-fold cross-validation on the training set, we select the SVM as the best performing model for the POEMS and TSV datasets. We will use this model in all further experiments.

5.2 Class Imbalance in the POEMS Data

As shown in Table 1, the POEMS dataset is highly imbalanced, with far more instances for the non-metaphorical class. A common problem when training classifiers on imbalanced data is the classifier's bias towards the majority class. Several techniques have been proposed to tackle this problem (Chawla, 2010). One example are resampling techniques where, in the case of *oversampling*, the minority class is increased by randomly adding duplicates from this class to the training set. *Undersampling*, on the other hand, reduces the number of instances from the majority class in order to obtain a more balanced distribution, at the cost of decreasing the size of the training data.

Another solution is *cost-sensitive learning* where the model is punished harder when misclassifying instances from the minority class while prediction errors on the majority class do not lead to high costs. We determine the best suited approach to deal with class imbalance using 10-fold cross-validation on the POEMS training set. We select a cost-sensitive SVM[12] with a Radial Basis Function (RBF) kernel for the POEMS model. We use this cost-sensitive model in all further experiments.

[11] We use the Scikit-learn toolkit (Pedregosa et al., 2011) implementations for all models.

[12] In Scikit-learn, this algorithm can be made cost-sensitive by adapting the parameter `class_weight`, which controls the weights attributed to each class.

Features	**F1 (macro)**	stdev	**F1 (M)**	stdev
All features	**72.7**	(8.2)	**54.7**	(14.1)
All - supers.	70.2	(8.8)	51.3	(14.2)
All - embed.	<u>67.8</u>	(7.6)	<u>48.4</u>	(12.5)
All - ratings	71.9	(7.8)	53.6	(13.4)

Table 2: Feature ablation on the POEMS data (Macro F1 and F1(M): F1 for the minority class; stdev: standard deviation for cross-validation).

5.3 Feature Selection

We perform feature selection for POEMS and TSV using feature ablation with 10-fold cross-validation on the respective training sets. We conduct these experiments to test our first hypothesis.

By dropping one feature at a time, we can determine the feature's importance by measuring the decrease in performance in terms of F1-score. Table 2 shows 10-fold cross-validation results for POEMS. The highest F1-score is bolded while the lowest is underlined. Removing word embeddings results in the highest loss in performance, showing their usefulness for metaphor detection. Since we obtain highest performance when using all features, we conclude that all feature types contribute relevant information and keep them for the next set of experiments.

We conduct the same experiment on the TSV data. As for the POEMS, best results are obtained when using all features. Results for the balanced TSV dataset, however, are much higher with an F1-score of 82.8% (10-fold cross-validation on the training set).

We also compare the impact of different embeddings types. For POEMS, we obtain best results for the SkipGram embeddings (Reimers et al., 2014) while for TSV, 100-dimensional FastText embeddings (Bojanowski et al., 2017) trained on the SDeWac corpus (Faaß and Eckart, 2013) give slightly higher results. We use FastText for all subsequent experiments on the TSV dataset.

5.4 Context Features

Turney et al. (2011) state the hypothesis that "the degree of abstractness of the context in which a given word appears is predictive of whether the word is used in a metaphorical or literal sense". They support their claim with experiments showing that i) the abstractness of an adjective's noun, seen as context, can be used to predict the adjective's metaphoricity and ii) averaged abstractness ratings of a verb's context, excluding the verb itself, can be used to predict the verb's metaphoricity. In all experiments, the authors report classification performances significantly higher than the majority class baselines and systems from related work.

While Turney et al. (2011) use only the noun modified by the adjective as context, we extend their hypothesis and test whether using features extracted from the A-N pairs' surrounding context can further improve classification accuracy (**H2**). In addition to affective ratings, we also extract supersenses and word embeddings from the context and add these new features to the feature vectors.

Context feature extraction We extract features for word embeddings, supersenses and affective ratings from a context window of size 20 for each A-N pair and concatenate the additional feature representations with the feature vectors for the A-N pairs. Similar to Turney et al. (2011)'s experiments on verbs, we do not extract features for every word in the context but limit feature extraction to adjectives, nouns and verbs. The final context representation is the average over all individual context features for a specific A-N pair.

We use the same word embedding types that gave us best results in the previous experiments. In other words, we use Reimers et al. (2014)'s word embeddings for POEMS and FastText word embeddings trained on SDeWac for the TSV data.

Context feature selection We now compare the setting without context, using only features extracted for the A-N pairs, to models that are also trained on context features. Again, we perform an ablation study to measure each context feature's importance, doing 5-fold cross-validation on the training set. The results for POEMS in Table 3 show a slight increase in overall F1-score; however, the improvement on the minority class is subtle and the high standard deviations for the different folds shows that the results are not robust. Since removing context supersenses from the set of context features lowers the performance, we test another setting in which we add only context supersense features to the feature set. This setting corresponds to the last row in Table 3. Since the cross-validation results suggest that context supersenses might include relevant information for the metaphor detection model, we add them to the feature vector.

We run the same experiment for TSV. For this model, we achieve highest F1-scores without additional context features. This means that we found no evidence to support hypothesis **H2** that con-

Cont. features	F1 (macro)	stdev	F1 (M)	stdev
None	73.8	6.2	54.3	10.9
All	74.0	4.4	54.2	8.2
All - supers.	72.9	4.3	52.3	8.0
All - embed.	74.4	3.4	55.0	6.3
All - ratings	73.5	4.9	53.1	9.2
Only supers.	**74.5**	3.5	**55.3**	6.3

Table 3: Poems context feature selection (in addition to features extracted from A-N pairs)

text features can provide useful information for metaphor detection for A-N pairs.

This is in contrast to Turney et al. (2011) who did report positive results for employing context features for metaphor detection. There are, however, some crucial differences between their and our setup. For adjectives, Turney et al. (2011) used only the adjectives' nominal heads as context but did not include additional context features extracted from the local context of the A-N pair. Thus, their experiments only show improvements for using context for verbal metaphors where they do extract abstractness features for all nouns, adjectives and verbs in a sentence.

As a result, we do not know whether the use of additional context features might only be relevant for verbs where we would add information for verbal arguments that might be useful for disambiguation. For adjectives, we already include their syntactic heads in our setup and the surrounding context might not be as relevant as for verbal metaphors.

It might also be possible that our setup for extracting context information from a fixed-size window around the target A-N pair is suboptimal and that a different approach might be more successful. We leave this to future work.

Results on the test sets All experiments reported above were run in a cross-validation setup on the training sets and served to determine the best model and feature combinations for each dataset. We now report results on the test sets for the selected models and compare them to two baselines (Table 4). The *majority* baseline simulates a rule-based system that always predicts the majority class (i.e. literal) while *probability matching* corresponds to a classifier that makes random predictions according to the training set class distribution. For Poems, we also report the performance using supersense context features in addition to the features extracted from A-N pairs (*All features + cont.*).

Model	Features	macro F1	F1 (M)
Poems	Majority	42.6	0.0
	Probab. matching	53.2	26.3
	All features	62.9	42.8
	All + supersense cont.	61.7	37.1
Tsv	Majority	35.2	0.0
	Probab. matching	45.0	43.5
	All features	79.7	76.3

Table 4: Baselines and test set results for Poems and Tsv. *All features* corresponds to the best model from the previous experiments.

For the Poems, the features we investigate produce a model that substantially outperforms both baselines and seems to be able to distinguish between metaphorical and literal uses of adjectives. The improvements over the baselines, however, are not statistically significant. In addition, the supersense context features selected using cross-validation on the training set do not generalise to new data. We speculate that this might be related to the dataset's class imbalance in addition to its relatively small size. Thus, increasing the size of the dataset is highly recommendable and should be the next step for future work.

For the balanced Tsv dataset, results for the selected model (last row of Table 4) are much higher with an average F1 of 79.7% and an F1 for the metaphorical class of 76.3%. Here, results are also significantly better than both baselines.[13]

5.5 Domain impact

In our last experiment, we test the usefulness of out-of-domain training data for metaphor detection in German poetry (**H3**).

The datasets used for Poems and for Tsv present several differences: the former is unbalanced, as opposed to the latter, and of smaller size. Moreover, both datasets originate from different domains, namely i) poetry and ii) a range of different genres from the web. To test our hypothesis that out-of-domain data can be used to improve results for metaphor detection in poetry, we conduct the following experiment.

We merge the training sets from the Tsv and Poems datasets and shuffle the resulting dataset with a fixed random state for reproducibility. We obtain a training set consisting of 624 metaphorical and 1,047 literal instances, which form a total of 1,617 training instances; we denote this new training set

[13] Using McNemar's test, both p-values are below 0.000001.

Model	Training set	#	F1 (macro)	F1 (M)
Poems	Original	578	62.9	42.8
	Merged	1,671	58.8	32.9
TSV	Original	1,149	79.7	76.3
	Merged	1,671	81.2	78.3

Table 5: Test set results of POEMS and TSV using *merged* in comparison to original training sets

as *merged*. We then train the best models selected for POEMS and TSV on the *merged* training set and test the models on the original two test sets.

Table 5 shows results for the models trained on the *merged* training set; results using the original training sets are repeated for comparison. Adding out-of-domain training data to the POEMS did not improve results, meaning that we cannot confirm **H3** (repeated below).

H3: Metaphors are not domain-dependent but a general cognitive phenomenon, thus supplementary out-of-domain training data can improve results for metaphor detection in poetry.

The performance for the TSV data, however, did increase when adding the additional training data from the poetry corpus. At first glance, this is somewhat surprising as adding the TSV data to the poems results in a more balanced training set. This, however, might not be the best idea when the distribution in the test set is highly imbalanced. As a result, the classifier might have lost crucial information about the class distribution, which might explain the decrease in results.

To test whether this loss of information is responsible for the results, we run another experiment where we downsample both data sets so that both training and test sets have the same size and class distribution (table 6). Then we retrain and test our models on the resized datasets.[14] Table 7 shows the same trend as before: training on POEMS does not decrease results for TSV while training on TSV yields substantially lower results for the POEMS.

	train (M/L)	test (M/L)
POEMS (orig)	100/478	98/280
TSV (orig)	546/603	65/77
DOWN-SAMPLED	100/217	65/77

Table 6: Train size and distribution of **M**etaphorical / **L**iteral instances in the datasets.

Model	Train-Test	Prec.	Rec.	F1	stdev
Poems	Poems-Poems	63.4	62.2	61.9	3.8
	Poems-TSV	75.0	65.9	64.1	2.0
TSV	TSV-TSV	75.1	65.2	63.0	2.3
	TSV-Poems	59.5	52.9	44.3	3.6

Table 7: Cross-domain results for downsampled datasets (averaged over 10 runs).

These results should be taken with a grain of salt as the datasets are very small. In future work, we would like to validate our findings on larger data.

For now, we cannot confirm that the difference in class distribution in the training and test sets is the underlying reason for our negative results regarding H3. We thus have to assume that the use of metaphors in expressionist poetry is crucially different from the one in every-day life, as described by Dietz (1959).

6 Conclusions & Outlook

In the paper, we presented first steps towards metaphor detection in German literature, in particular, in expressionist poetry. We created two datasets with adjective-noun pairs, manually annotated for metaphoricity, and evaluated models for metaphor detection for German. Our results show that features that have been used for other languages work well for German, too, and that word embeddings in particular are valuable features. We tested whether additional context information can improve classification accuracy, with negative results. We also explored the domain-dependence for metaphor detection by adding supplementary out-of-domain training data. Here, the results were mixed and require future investigation.

One important finding of our work is that results for metaphor detection are highly dependent on the class distribution in the dataset, and that balanced corpora give overly optimistic and thus misleading results.

For future work, the most important step is to increase the size of the datasets and to add annotations for other metaphors types, such as verbal metaphors. Metaphor annotation is a challenging task where agreement between annotators is often low. As has been noted before (Shutova, 2017; Gibbs, 1984), metaphoricity can be seen as a continuum. Thus, it might be recommendable to annotate metaphors on a scale instead of categorising them into binary classes. This would allow annotators to capture shades of gray and give them more flexibility while avoiding arbitrary decisions.

[14] We report averaged results over 10 trials of sampling with replacement.

Acknowledgments

This research has been partly conducted within the Leibniz Science Campus "Empirical Linguistics and Computational Modeling", funded by the Leibniz Association under grant no. SAS-2015-IDS-LWC and by the Ministry of Science, Research, and Art (MWK) of the state of Baden-Württemberg.

References

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.

Leo Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

Luana Bulat, Stephen Clark, and Ekaterina Shutova. 2017. Modelling metaphor with attribute-based semantics. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, volume 2, pages 523–528.

Nitesh V. Chawla. 2010. Data mining for imbalanced datasets: An overview. In *Data Mining and Knowledge Discovery Handbook, 2nd ed.*, pages 875–886.

Ludwig Dietz. 1959. *Die lyrische Form Georg Trakls*. Salzburg. Otto Müller.

Erik-Lân Do Dinh, Hannah Wieland, and Iryna Gurevych. 2018. Weeding out conventionalized metaphors: A corpus of novel metaphor annotations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1424.

Gertrud Faaß and Kerstin Eckart. 2013. SdeWaC – A Corpus of Parsable Sentences from the Web. In *Language processing and knowledge in the Web*, pages 61–68. Springer.

Raymond W. Jr. Gibbs. 1984. Literal Meaning and Psychological Theory. *Cognitive science*, 8(3):275–304.

Birgit Hamp and Helmut Feldweg. 1997. GermaNet – a Lexical-Semantic Net for German. *Automatic information extraction and building of lexical semantic resources for NLP applications*.

Thorsten Joachims. 1998. Text categorization with Support Vector Machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer.

Vaibhav Kesarwani, Diana Inkpen, Stan Szpakowicz, and Chris Tanasescu. 2017. Metaphor detection in a poetry corpus. In *The Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, LaTeCH 2017, pages 1–9.

Maximilian Köper and Sabine Schulte im Walde. 2016a. Automatic semantic classification of german preposition types: Comparing hard and soft clustering approaches across features. In *The 54th Annual Meeting of the Association for Computational Linguistics*, ACL 2016.

Maximilian Köper and Sabine Schulte im Walde. 2016b. Automatically Generated Affective Norms of Abstractness, Arousal, Imageability and Valence for 350 000 German Lemmas. In *LREC*.

Maximilian Köper and Sabine Schulte im Walde. 2017. Applying Multi-Sense Embeddings for German Verbs to Determine Semantic Relatedness and to Detect Non-Literal Language. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, volume 2, pages 535–542.

Saisuresh Krishnakumaran and Xiaojin Zhu. 2007. Hunting elusive metaphors using lexical resources. In *Workshop on Computational Approaches to Figurative Language*, pages 13–20.

George Lakoff. 1987. *Women, Fire, and Dangerous Things*. Chicago University Press.

Saskia Le Cessie and Johannes C. Van Houwelingen. 1992. Ridge Estimators in Logistic Regression. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 41(1):191–201.

Birte Lönneker-Rodman. 2008. The hamburg metaphor database project: issues in resource creation. *Language Resources and Evaluation*, 42(3):293–318.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research*, 12(Oct):2825–2830.

Pragglejaz Group. 2007. MIP: A Method for Identifying Metaphorically Used Words in Discourse. *Metaphor and Symbol*, 22(1):1–39.

Marek Rei, Luana Bulat, Douwe Kiela, and Ekaterina Shutova. 2017. Grasping the finer point: A supervised similarity network for metaphor detection. In *The 2017 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2017, pages 1537–1546.

Nils Reimers, Judith Eckle-Kohler, Carsten Schnober, Jungi Kim, and Iryna Gurevych. 2014. GermEval-2014: Nested Named Entity Recognition with Neural Networks. In Gertrud Faaß and Josef Ruppenhofer, editors, *Workshop Proceedings of the 12th Edition of the KONVENS Conference*, pages

117–120, Hildesheim, October. Universitätsverlag Hildesheim.

Helmut Schmid. 1994. Probabilistic Part-Of-Speech Tagging Using Decision Trees. In *Proceedings of International Conference on New Methods in Language Processing, 1994*.

Helmut Schmid. 1995. Improvements in Part-Of-Speech Tagging with an application to German. *EAL SIGDAT work-shop, 1995*.

Ekaterina Shutova, Simone Teufel, and Anna Korhonen. 2013. Statistical metaphor processing. *Computational Linguistics*, 39(2):301–353.

Ekaterina Shutova, Douwe Kiela, and Jean Maillard. 2016. Black Holes and White Rabbits: Metaphor Identification with Visual Features. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 160–170.

Ekaterina Shutova. 2017. Annotation of Linguistic and Conceptual Metaphor. In *Handbook of Linguistic Annotation*, pages 1073–1100. Springer.

Chris Tanasescu, Vaibhav Kesarwani, and Diana Inkpen. 2018. Metaphor detection by deep learning and the place of poetic metaphor in digital humanities. In *The 31st International Florida Artificial Intelligence Research Society Conference*, FLAIRS 2018, pages 122–127.

Yulia Tsvetkov, Leonid Boytsov, Anatole Gershman, Eric Nyberg, and Chris Dyer. 2014. Metaphor detection with cross-lingual model transfer. In *The 52nd Annual Meeting of the Association for Computational Linguistics*, ACL 2014, pages 248–258.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 384–394, Stroudsburg, PA, USA. Association for Computational Linguistics.

Peter D. Turney, Yair Neuman, Dan Assaf, and Yohai Cohen. 2011. Literal and metaphorical sense identification through concrete and abstract context. In *The 2011 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2011, pages 680–690.

A Appendices

A.1 List of authors for the POEMS dataset

The following table lists the authors of the poems included in the POEMS corpus, with the number of poems for each author.

Author	**# poems**	**Author**	**# poems**
Gottfried Benn	10	Johannes R. Becher	6
Julius Maria Becker	24	Frieda Bettingen	22
Ernst Blass	68	Paul Boldt	8
Theodor Däubler	62	Gerrit Engelke	66
Max Herrmann-Neisse	12	Georg Heym	87
Jakob van Hoddis	8	Oskar Kanehl	4
Georg Kulka	28	Else Lasker-Schueler	151
Heinrich Lersch	47	Alfred Lichtenstein	123
Oskar Loerke	45	Ernst Wilhelm Lotz	19
Ludwig Rubiner	21	Gustav Sack	65
Daniel Schiebeler	4	Ernst Stadler	116
August Stramm	71	Ernst Toller	5
Georg Trakl	91	Franz Werfel	35
Alfred Wolfenstein	2		

Authors of poems in the dataset POEMS

A.2 Categorisation for adjectives and annotation criteria

Category	Instances in training set	Literal ex.	Metaphorical ex.	Ambiguous ex.	Criteria
Temperature	heiß, kalt, kühl, lau, schwül, eisig, mild, vereist, warm	heiße Stirn	heiße Träne, heißes Auge	heißes Feld	Literal meanings: temperature property of an object (‘Teller’), body part (‘Hand’) or a substance (‘Wind’). Metaphorical: unusual uses of such adjectives are those describing a noun that would not commonly have a temperature attribute (like an eye or a tear).
Color	blau, braun, bunt, gelb, vergilbt, grau, grün, purpur, rosig, rot, schwarz, weiß	blauer Abend, blaue Nacht, blaues Licht	blaue Seele, braune Stille, roter Duft	blauer Tag	Literal: the noun refers to an object that can have the property of a color. Abstract nouns such as ‘soul’ or concepts that cannot be touched such as ‘odor’ are metaphorical cases. In difficult cases such as ‘night’, ‘day’, ‘evening’, the noun does not refer to a physical object but since it can have the attribute of a color, it is considered literal.
Material	golden, kristallen, silbern, steinern, seiden	goldenes Auge, goldene Stufe, goldenes Bildnis, goldener Wald, goldene Luft, goldene Wolke, seidener Strumpf, silberne Hand	steinerne Geduld	goldener Tag	Literal: when the noun refers to a physical object such as an item. Difficulties occurred for the adjective ‘golden’; the decision was made to exclude any sense that does not refer to the material gold or its tint/color from literal tags. Metaphorical: abstract concepts (idea, feeling). ‘goldener Tag’ can mean that the sky is tinted in a golden color (literal) or a day (time indication) was excellent in a certain way (metaphorical); thus the pair is ambiguous.
Texture, substance	blank, dicht, dornig, dumpf, hart, kahl, rostig, trüb, weich, zart, fest	blankes Gewehr, blanker Himmel, dichte Wiese, hartes Lager	hartes Licht, harte Luft	weicher Weg	Literal pairs contain an adjective describing a physical object’s texture or substance: items, surfaces, body parts or certain locations (‘Wiese’). Yet light cannot have a hard texture, for example, thus the corresponding pair is metaphorical.
Shape, volume, weight	dick, dünn, dürr, eng, rund, schmal, spitz, gewölbt, gezackt, hohl, leicht, offen, schwer, tief, weit	dünnes Licht, dünner Nebel, dürres Rohr, tiefe Wunde	schweres Lid, schwerer Schlaf, dürre Straße	-	Literal pairs contain an adjective describing a physical object’s shape, volume and weight. However, ‘dünn’ can concretely describe the narrow shape of a ray of light and is tagged as literal. ‘schweres Lid’ describes a person’s eyelid with the tendency to close; thus the adjective is used in a metaphorical way.
Size, length, height	groß, klein, lang, kurz, hoch, nieder	großer Himmel, große Not, kleines Lied, kleine Stimme, langer Tag, lange Stunde	großes Leben, große Nacht, großer Tag, hohe Lust	hohes Gut, hohe Nacht	Literal: quantifiable or classifiable attribute (‘groß’: of high size, surface or intensity, ‘hoch’: placed high in space or high in quantity or intensity, ‘lang’, ‘kurz’, ‘klein’: an object’s length or an event’s duration). Any other sense is metaphorical.
Age, time	alt, erst, früh, jung, letzt, neu, reif, ewig	altes Haus, alter Gott, erster Mensch, neuer Mensch, frühes Gedicht, ewige Nacht	alter Tag, junger Frühling, junge Kraft, ewiger Raum	-	Objects and living beings can be described as old or young/new. A distinction is made between ‘new’ and ‘young’ since the second one is reserved to living beings (thus ‘junger Frühling’ is metaphorical). In the case of ‘ewiger Raum’, the adjective’s sense is modified to describe a surface or extension in space, thus the pair is metaphorical.
Light	dunkel, düster, finster, glühend, hell, schimmernd, strahlend	dunkler Baum, dunkles Wasser, helle Nacht	dunkle Frage, dunkle Stimme, finsterer Zorn, glühendes Blut	dunkler Wald, dunkler Weg, düsteres Bild, finsteres Wasser, glühende Erde	Adjectives such as ‘dunkel’, ‘düster’ and ‘finster’ refer to a low light context in the literal sense. Any pair that makes use of the second sense (uncanny, sinister) is tagged as metaphorical. This category presents a high amount of ambiguous cases since both senses are often possible and context would be necessary to determine the adjective’s sense.

Annotation criteria for different adjective categories