Supplementary Notes to *Detecting Derogatory Compounds – An Unsupervised Approach*

March 2, 2019

1 Introduction

This document provides more detailed information regarding certain aspects of our research for which there was not sufficient space in the main paper. We focus on the following aspects:

- creation of the gold standards (§2)
- details regarding the computation of the individual high-precision diagnostics including pseudocode (§3)
- hyperparameter tuning of classifiers (§4)
- backing off compound embeddings based on the compositional model *Wmask* from Dima [2015] (§5)
- details regarding replicating the set of linguistic features from Wiegand et al. [2018a] on German-language data (§6)

2 Creation of the Gold Standards

In our work, we use two different gold standards, a gold standard of compounds (§2.1) and a gold standard of unigrams (§2.2). In both cases, the gold standards represent expressions (i.e. either compounds or unigrams) that have been manually annotated out of context. We employ a binary categorization scheme. Each expression is assigned exactly one label: an expression is either labeled as *derogatory* or *not derogatory*.

2.1 Compound Gold Standard Dataset

The creation of a gold standard for learning derogatory compounds is notably different from the creation of a gold standard for learning derogatory unigrams as proposed by Wiegand et al. [2018a]. The most crucial difference is that for their unigram gold standard Wiegand et al. [2018a] made the assumption that derogatory words form a proper subset of negative polar expressions. As a consequence, their dataset exclusively comprised negative polar expressions,

the rationale being that a sufficiently large sentiment lexicon will contain among negative polar expressions derogatory words as well. This notably simplifies the detection of derogatory words, since the search space is reduced to just negative polar expressions.

While the general assumption of derogatory words being a subset of negative polar expressions also holds for derogatory compounds, we cannot follow the same procedure for our compound gold standard. Since there exist so many compounds and compounds are generally rare in lexical resources, such as sentiment lexicons, we cannot reduce the search space to just negative polar compounds.[1] Instead, we consider in our gold standard a sample of (noun-noun) compounds. Compounds are included no matter whether they are negative polar expressions or not. Our procedure specifically comprised following steps:

1. Identify a set of derogatory compounds from existing lexicons of derogatory language.
2. For each derogatory compound sample from the web other compounds sharing the same head.
3. Manually annotate the set of compounds gained by the previous step.

The websites from which we collected derogatory compounds for Step 1 were:

- `www.hyperhero.com/de/insults.htm`
- `www.seechat.de/warmduscher.htm`
- `www.schimpfwoerter.de`

These websites contain a high proportion of derogatory compounds, however, many of these compounds are noun-noun compounds in which one constituent, i.e. either head or modifier, represent a derogatory unigram, e.g. *Handtuchwichser (towel wanker)*, *Mutterficker (motherfucker)*, *Knasthure (prison whore)*. Such compounds are not part of our dataset, since they are fairly easy to detect in the light of lexicons of derogatory unigrams and methods to induce such resources as proposed by Wiegand et al. [2018a]. There even exist tools on the web that generate compounds by randomly combining derogatory words with each other or some other words, such as `http://sweary.com` or `http://foulomatic.hnldesign.nl`. Our gold standard will only comprise (noun-noun) compounds where both head and modifier do not represent derogatory unigrams.

Since we speculate that one can identify a derogatory compound by contrasting it to other compounds sharing the same head (e.g. *booze hound* vs. *fox hound*, *sight hound*, *stag hound*), we create our dataset in such a way that for each derogatory compound there is a substantial amount of other compounds that share the same head (Step 2). This is actually also our way of gaining negative instances (i.e. non-derogatory compounds). We call the set of compounds sharing the same head *head group*.

[1] We hardly found any derogatory noun-noun compounds in the German PolArt sentiment lexicon [Klenner et al., 2009], the standard sentiment lexicon for German language.

Each head group is set to 20 compounds. Thus, we want to avoid biases towards particular heads as it may otherwise distort the evaluation of classification approaches [Shwartz and Waterson, 2018].[2]

We felt the necessity to also manually annotate the set of compounds gained by Step 2 since among these compounds there could well be further derogatory compounds. Just looking up compounds in the web-based resources listed above would also be insufficient since none of these resources exhaustively lists derogatory compounds.

Step 3 also justifies why the task of detecting derogatory compounds cannot be reduced to looking up compounds in such word lists. New derogatory words constantly enter natural language [Wiegand et al., 2018a] and compounds are no exception to that. As a consequence, the lexicons that are currently available cannot be complete. Moreover, publicly available lexicons of derogatory language, particularly those that have been created by some form of crowdsourcing (such as the sources listed above), contain a large amount of noise. There are many ambiguous entries, that is, words that may have a derogatory connotation in certain contexts but, generally, are not considered as derogatory words, e.g. *Colatrinker (coke drinker), Matratzenschläfer (matress sleeper), Wurstwasser (canning liquid), Zahnstocher (toothpick)*. This issue was already identified by Wiegand et al. [2018a] and shown to have a very detrimental effect when converting such lexical resources to a classifier to identify entire abusive utterances (e.g. abusive forum posts or tweets), since these lexical resources will create huge amounts of false positives. This, in general, challenges the usage of such ambiguous word lists for the detection of abusive language.

2.2 Unigram Gold Standard Dataset

We also created a gold standard for derogatory unigrams, since we wanted to examine in how far derogatory compounds can be detected by a classifier trained on derogatory unigrams. If good classification performance on compounds were obtained by classifiers trained by derogatory unigrams, then this would mean that the task of detecting derogatory compounds would not have to be specifically addressed.

Since there exists only an English gold standard of unigrams [Wiegand et al., 2018a] (i.e. the *base lexicon* of 1650 negative polar expressions that were manually classified as either *derogatory* or *not derogatory*) but our compounds are in German, we had to produce a German gold standard of unigrams and translated the English unigram gold standard to German. We translated this lexicon *manually* since we wanted to make sure that the translated words preserve the same degree of derogation as the original English words. (This cannot be achieved by automatic translation.) Unfortunately, a substantial number of English derogatory words could not be translated into German as, due to cultural differences, some English words simply have no German counterpart. For instance, the word *spic* refers to a Spanish-speaking person from Central or South America or the Caribbean, especially a Mexican. Such persons represent a minority in North

[2]For example, there is a large number of German derogatory compounds ending with *Kopf (head)*, e.g. *Atomkopf*, *Babykopf*, *Dönerkopf*, *Eierkopf*, *Saufkopf*, *Wurstkopf* etc. By not restricting the number of heads a classifier would tend to learn that *Kopf* is a derogatory cue. Due to the predominance of that constituent on a dataset, this feature would cause classifiers to score unrealistically well.

America that are frequently verbally offended but not in Germany. There are other minority groups in German society who fall victim to verbal abuse.

In order to obtain the best possible result with this unigram gold standard (as we consider it a *baseline* and we want to produce the strongest possible baseline), we also adjusted that lexicon to the class distribution of our compound dataset.[3] Our final German gold standard comprises 1128 negative polar expressions.

3 Details regarding High-Precision Diagnostics

Almost all our diagnostics rely on a distributional representation of each of our compounds. We induced word-embedding vectors for our compounds using Word2Vec [Mikolov et al., 2013] on the COW16 corpus [Schäfer, 2015]. We leave the tool in its standard configuration with the exception of the number of dimensions, which we set to 100. We made exploratory experiments and found that this configuration performed best. Note that word embeddings also represent critical components of our baselines and using an optimal number of dimensions also meant that these baselines became similarly stronger.

We want to represent our compounds as individual tokens. As we deal with German compounds, this can be achieved by applying a standard tokenization of the corpus prior to inducing the word embeddings. This is since the set of our German noun-noun compounds are either closed compounds (e.g. *Fadenabschneider*) or hyphened compounds (e.g. *Rollstuhl-Athlet*). Both are considered as single tokens by standard tokenizers of German.

3.1 PERSON-feature (Derogatory Compound Must be Person)

Since compounds are rare, we cannot look them up in lexical resources, such as GermaNet [Hamp and Feldweg, 1997]. Therefore, deciding whether a compound represents a person or not can only be determined in a data-driven manner. We rank the compounds according to their cosine-similarity to a centroid vector representing persons. The more similar a compound is to that vector, the more likely it represents a person.

The centroid-person vector is computed by averaging embeddings of words representing common professions. We also experimented with personal pronouns as a proxy for persons. However, we found them unsuitable since they are also often used as referring expressions to other entities, such as animals. Professions, on the other hand, can only refer to humans. The list of professions we used was created ad-hoc. It is pretty generic and we did not make any attempt to tune the set of professions to our task. (It could well be that a smaller list would perform equally well or even better.) It should be reproducible in any arbitrary language. The specific nouns that we used were:

Anwalt (attorney), *Arzt (medical doctor)*, *Bäcker (baker)*, *Bauer (farmer)*, *Bankkaufmann (banker)*, *Beamter (clerk)*, *Betriebswirt (economist)*, *Doktor (doctor)*, *Dozent (lecturer)*, *Elektrotechniker (electrician)*, *Forscher (researcher)*, *Friseur (hairdresser)*, *Gastwirt (barkeeper)*, *Gärtner (gardener)*, *Handwerker (craftsman)*, *Ingenieur (engineer)*, *Informatiker (computer scientist)*, *Jurist (lawyer)*,

[3] Using the original class distribution produced notably worse results.

Kanzler (chancellor), *Koch (chef)*, *Konditor (confectioner)*, *Maler (painter)*, *Maurer (mason)*, *Landwirt (countryman)*, *Lehrer (teacher)*, *Minister (minister)*, *Manager (manager)*, *Offizier (officer)*, *Politiker (politician)*, *Präsident (president)*, *Professor (professor)*, *Schauspieler (actor)*, *Sekretär(in) (secretary)*, *Therapeut (therapist)*, *Tischler (carpenter)*, *Verkäufer (salesman)*, *Wissenschaftler (scientist)*.

3.2 Compound Occurrence vs. Constituent Occurrence.

Derogatory compounds can be very creative word constructions (e.g. *booze hound*, *oxygen thief*, *keyboard warrior*). As a consequence, their constituents are often not semantically related. For instance, in *booze hound*, *booze* bears no common semantic relation towards *hound*. Therefore, the corpus-frequency of a derogatory compound should be much higher than their constituents (i.e. head and modifier) co-occurring in a sentence with other words occurring in between.

We try to capture this by the following formula (the frequencies are computed on the COW16 corpus):

$$COMCON = \frac{\text{\# compound mentions in corpus}}{\text{\# mentions of head and modifier co-occurring in a sentence}} \tag{1}$$

In prose, COMCON ranks all compounds by the ratio of observed compound occurrences and constituent co-occurrences in a sentence. For derogatory compounds, there should be a high frequency of compound mentions but only a low frequency of the constituents co-occurring in a sentence. Therefore, COMCON will have a high score. For non-derogatory compounds, there should be a notably higher frequency of the constituents co-occurring in a sentence since constituents are usually semantically related (e.g. *Warenhausbesitzer (warehouse owner)*, *Zirkusclown (circus clown)*) and, therefore, the constituents may also co-occur more often within a sentence. This should result in COMCON producing comparably lower scores.

3.3 Outlier Compound in Head Group

Algorithm 1 displays the pseudocode for computing an outlier score among compounds (OUT). We are interested in those compounds that are most dissimilar to all the other compounds within the same head group (e.g. *keyboard warrior* vs. *rajput warrior*, *ninja warrior*, *samurai warrior*) Therefore, we first compute the average pairwise cosine-similarity of a compound to all other compounds within a head group (this corresponds to computeAvgSimToCompoundsInHeadGroup in Algorithm 1). Then, we convert the similarity score to a dissimilarity score (by taking its inverse).

Additionally, we need to consider how similar all compounds are within a particular head group. We can only identify an outlier if most compounds within the same head group are generally homogeneous. (For example, if the compounds within the same head group are generally heterogeneous, as in **legacy hunter**, *job hunter*, *autograph hunter*, then it is less obvious from the viewpoint of distributionally similarity which compound is the derogatory outlier.) Therefore, we compute a homogeneity score of each head group

(computeHomogeneityScore in Algorithm 1) and multiply it with the dissimilarity score. This will ensure that compounds being dissimilar to other compounds of the same head group with that head group being fairly heterogeneous to be have a lower rank than compounds being dissimilar to other compounds of the same head group with that head group being fairly homogeneous.

Algorithm 1 Computation of the outlier scores for compounds (OUT).

```
procedure computeOutlierScore(compound)
                                        ▷ A compound is represented by its embedding-vector
    headGroup ⇐ getHeadGroup(compound)
    simScore ⇐ computeAvgSimToCompoundsInHeadGroup(compound, headGroup)
    dissimScore ⇐ 1.0 / simScore
    homogenScore ⇐ computeHomogeneityScore(headGroup)
    outlierScore ⇐ homogenScore · dissimScore
    return outlierScore
end procedure

procedure computeAvgSimToCompoundsInHeadGroup(compoundA, headGroup)
    simSum ⇐ 0                                              ▷ sums the pairwise similarities
    noOfSims ⇐ 0                                          ▷ counts the number of similarities
    for compoundB : getCompounds(headGroup) do
        if compoundA ≠ compoundB then
            pairSim ⇐ cosineSim(compoundA, compoundB)
            simSum ⇐ simSum + pairSim
            noOfSims ⇐ noOfSims + 1
        end if
    end for
    avgSim ⇐ simSum / noOfSims
    return avgSim
end procedure

procedure computeHomogeneityScore(headGroup)
    simSum ⇐ 0                                              ▷ sums the pairwise similarities
    noOfSims ⇐ 0                                          ▷ counts the number of similarities
    for compoundA : getCompounds(headGroup) do
        for compoundB : getCompounds(headGroup) do
            if compoundA ≠ compoundB then
                pairSim ⇐ cosineSim(compoundA, compoundB)
                simSum ⇐ simSum + pairSim
                noOfSims ⇐ noOfSims + 1
            end if
        end for
    end for
    homogenScore ⇐ simScore / noOfSims
    return homogenScore
end procedure
```

4 Hyperparameter Tuning for Supervised Classifiers

For our supervised-learning experiments that used the gold standard of derogatory compounds as training and test data, we first split the data into two disjoint datasets. The first dataset comprises 3000 instances that were used for 10-fold crossvalidation. The second dataset comprising the remaining 500 compounds was used as a development set. (In the first step, we trained on the 3000 in-

stances and tuned hyperparameters on the development set. In the second step, we ran crossvalidation on the 3000 instances using the values established on the development set.) Prior to partitioning the gold standard, we randomized the order of all compounds.

4.1 Character-based Classifier

As a classifier based on sequential character information, we conducted experiments on LSTM being the most state-of-the-art sequential classification algorithm for deep learning. The input of this classifier is the sequence of characters representing a compound. For the representation of characters, we employed a one-hot encoding. We implemented our LSTM with the help of keras.[4] The parameters we tuned on the development data included:

- batch size
- number of epochs
- activation function
- number of hidden layers
- number of neurons
- class weight

4.2 SVM

In our experiments, we used SVM^{Light} [Joachims, 1999] which is widely used for NLP-related tasks and is particularly effective on the detection of derogatory language [Schmidt and Wiegand, 2017, Wiegand et al., 2018a,b]. Since in our gold standard, the derogatory compounds represent a minority class (with approximately 11% of the compounds), the SVM needs to be adjusted to the given class distribution. For that purpose, SVM^{Light} offers a j-parameter that represents a cost-factor by which training errors on positive examples outweigh errors on negative examples. This parameter was tuned on the development set.

4.3 Hyperparameters of the Supervised Classifier Trained on the Unigram Gold Standard

As a baseline, we also trained an SVM-classifier on the unigram gold standard (§2.2) using the linguistic features from Wiegand et al. [2018a] and word embeddings. Since with this baseline, we want to examine in how far we can predict derogatory compounds with training data that comprise derogatory unigrams but no derogatory compounds, we adjusted the hyperparameters for the SVM (i.e. the j-parameter) on the unigram dataset.

[4] `https://keras.io/`

4.4 Hyperparameters of the Unsupervised Classification Approach

The unsupervised approach that we propose in this work also contains some hyperparameters that need to be set. Since this is an unsupervised approach that employs no manually labeled training data, the parameter values were set ad-hoc, mostly by making common-sense assumptions. The most task-specific information we employ in this respect is knowledge about the class distribution in our dataset. However, we have strong reasons to believe that this class distribution is representative of the task and not just our particular dataset. (When sampling compounds for our dataset, we made sure that the proportion of derogatory seeds is reflecting the natural class distribution.) All parameter values were not specifically tuned on the test data and therefore should only be regarded as a lower bound. We briefly discuss the different parameters:

Combination of Individual Rankings. For the combination of individual rankings (COMB), we took the ranking of compounds ranked by their similarity to the centroid embedding-vector representing negative polarity (i.e. NEG) and then only maintained those compounds which were only included in top ranks of the other rankings (i.e. COMCON, PERSON and OUT). Using NEG as a basis for the combination is quite intuitive since conveying a negative polarity is a prerequisite of being derogatory [Sood et al., 2012, Dinakar et al., 2012, Gitari et al., 2015]. We chose for all other individual rankings the top 350 compounds which vaguely corresponds to the number of derogatory compounds in our gold standard lexicon. In order to avoid overfitting, we used the same amount of top ranks across the different individual rankings.

Personalized PageRank. The only essential parameter in this algorithm is α (uniform re-entrance weight). The value of this parameter was set to the default value recommended by Manning et al. [2008]. This means that no knowledge about our particular dataset was included in deciding about this parameter value.

Label Propagation. For label propagation, we made use of the Adsorption propagation algorithm as implemented in *junto* [Talukdar et al., 2008]. We only consider the ***default configuration*** of that tool. We chose the top 50 ranks from PRANK as derogatory seeds and the bottom 500 ranks as non-derogatory seeds. Again, the ratio 50:500 vaguely reflects the class distribution.

The output of label propagation is a binary categorization and each compound that has been classified also obtains a confidence score for the predicted category label. In order to produce a ranking from the prediction of label propagation, as we used it in our evaluation against the other types of rankings, we took the set of compounds classified by label propagation to be derogatory and ranked those compounds according to their confidence score.

5 Backing off Compound Embeddings based on Compositional Model *Wmask* from Dima [2015]

Noun compounds occur less frequently than unigram nouns. As a consequence, there is no publicly available corpus which would allow embeddings to be induced for our entire set of noun compounds. We examined various corpora and publicly available pre-trained embeddings. By far the best coverage on our compound

gold standard was obtained by inducing embeddings on the COW16 corpus [Schäfer, 2015]. Even with this corpus, which contains more than 20 billion tokens[5], we could only induce an embedding space for just 2101 compounds using Word2Vec [Mikolov et al., 2013]. Since this only amounts to 60% of compounds of our dataset and we found that, in general, those approaches that are based on an embedding representation of the entire compounds (rather than just their constituents, i.e. head and modifier) represent the most effective features for our task, we tried to come up with an approximated embedding representation for the remaining 40% of our compounds.

The toolbox *wordcomp*[6] implements various methods to approximate a compound embedding representation from embedding representations of its constituents. For our work, we considered the most complex composition function *Wmask* which is a sophisticated mask model proposed by Dima [2015]. This model builds upon the idea that when a word w enters a composition process, there is some variation in its meaning depending on whether it is the first or the second element of the composition. For each word in the vocabulary, two masks are learned: one for the case when it is the first word in the composition process (i.e. modifier) and one for when it is the second word (i.e. head). This allows a more accurate embedding representation of compounds.

For our task, we followed Dima [2015] by training the *Wmask*-model on the noun compounds of GermaNet 9.0.[7] With about 34,000 noun compounds, this is, by far, the largest set of German noun compounds available to the research community. The only differences to the setting *Wmask* in Dima [2015] are that we used different embeddings as input for the composition learning. In order to enable a meaningful comparison, we had to use exactly the type of embeddings, we also used for our remaining experiments, that is, embeddings induced on COW16 based on Word2Vec (100 dimensions). Dima [2015] induced embeddings using the GloVe-package Pennington et al. [2014]. Please note that the learning of the composition function, however, is not dependent on a particular embedding induction tool, i.e. Word2Vec or GloVe.

6 Replicating the Feature Set from Wiegand et al. [2018a] on German

In the following, we describe how we replicated the set of linguistic features from Wiegand et al. [2018a] (proposed for English data) on German data.

6.1 Polar Intensity

Wiegand et al. [2018a] proposed 3 different approaches to determine the polar intensity of words. They are:

1. A lexicon look-up using a sentiment lexicon that contains binary intensity information.

2. A derivation of polar intensity scores from the distribution of star ratings of reviews using a standard review corpus.

[5]The largest English corpus used in Wiegand et al. [2018a] is about 10 times smaller!
[6]`https://github.com/corinadima/wordcomp`
[7]`http://www.sfs.uni-tuebingen.de/lsd/compounds.shtml`

3. A derivation of polar intensity scores from the distribution of star ratings of reviews using a special review corpus that exclusively contains reviews that address persons. (The authors propose the usage of a crawl from the rateitall-website.[8])

Unfortunately, for German data, we are not aware of any sentiment lexicon containing polar intensity information which meant that Approach 1 could not be replicated. We refrained from automatically translating this resource since our intuition is that translations of polar expressions not necessarily preserve the same level of polar intensity in the target language.

There are only very few review corpora in German that also contain star rating information. We are not aware of any review corpus which allows a reliable isolation to reviews addressing persons. Therefore, we could only replicate a method to compute the polar intensity according to Approach 2.

6.2 Sentiment Views

There does not exist a complete German sentiment lexicon with sentiment views. However, there exist subsets manually annotated from the German PolArt sentiment lexicon [Klenner et al., 2009]. Wiegand and Ruppenhofer [2015] manually annotated all sentiment verbs from that lexicon while Wiegand et al. [2016] provided a manual annotation of atomic German sentiment nouns from the same resource. We merged these two subsets to one lexicon and extracted the binary sentiment-view feature with the help of this combined resource.

6.3 Emotion Categories

We used the German version of the NRC lexicon [Mohammad and Turney, 2013] (Version 0.92). This version comprises the same emotion categories as the English lexicon and the same vocabulary (but translated from the original English lexicon into German).

6.4 WordNet and Wiktionary

For the features derived from WordNet [Miller et al., 1990], we used the German version of WordNet called *GermaNet* [Hamp and Feldweg, 1997]. The design of GermaNet largely follows the English original resource. Wiktionary also encompasses languages other than English including German. Therefore, the WordNet and Wiktionary features proposed by Wiegand et al. [2018a] can be replicated in a straightforward manner on the corresponding German-language resources.

6.5 Surface Pattern

In principle, the surface pattern for English proposed by Wiegand et al. [2018a] (1) can be replicated to German (3).

(1) *English pattern*: called me a(n) <noun>
(2) *English pattern match example*: He called me a **tosser**.

[8] www.rateitall.com

(3) *German pattern*: hat mich ein(e|en) <noun> genannt
(4) *German pattern match example*: Er hat mich einen **Vollidioten** genannt.

However, the pattern is far sparser than the English one. Like Wiegand et al. [2018a] we ran our German pattern as a query on Twitter and extracted all matching tweets coming in a time period of 14 days. However, only 25 different words were thus extracted and all except one were only observed once. Such infrequent words were not sufficiently reliable. On average, less than every 3rd word of these matches was a derogatory word. Furthermore, the proportion of derogatory compounds was not that large either (2 out 25 words were noun-noun compounds). We assume that the surface pattern could also work in German, however, the period for streaming tweets from Twitter matching the query pattern would need to be significantly extended (i.e. several months or even a year), which was beyond the scope of our research.

As a consequence, we had to exclude this feature from the feature set to extract German derogatory words. It also meant that the weakly-supervised method from Wiegand et al. [2018a] (WSUP) could not be re-implemented either as it requires as input derogatory words that are obtained with the help of the surface pattern.

6.6 FrameNet

Although there exists a German equivalent to the English FrameNet [Baker et al., 1998], called Salsa-corpus [Burchardt et al., 2006], we refrained from using it for this work, since the German version is considerably smaller and mostly focuses on verbs (a few hundred words). We could hardly identify any offensive words among the lexical units that had been annotated for the German FrameNet. This does not come as a surprise since the most frequent part of speech in that resource, i.e. verbs, is known to yield only a very small fraction of derogatory words [Wiegand et al., 2018a].

References

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. The Berkeley FrameNet Project. In *Proceedings of the International Conference on Computational Linguistics and Annual Meeting of the Association for Computational Linguistics (COLING/ACL)*, pages 86–90, Montréal, Quebec, Canada, 1998.

Aljoscha Burchardt, Kathrin Erk Anette Frank, Andrea Kowalski, Sebastian Padó, and Manfred Pinkal. The SALSA Corpus: a German Corpus Resource for Lexical Semantics. In *Proceedings of the Conference on Language Resources and Evaluation (LREC)*, pages 969–974, Genoa, Italy, 2006.

Corina Dima. Reverse-engineering Language: A Study on the Semantic Compositionality of German Compounds. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1637–1642, Lisbon, Portugal, 2015.

Karthik Dinakar, Birago Jones, Catherine Havasi, Henry Lieberman, and Rosalind Picard. Common sense reasoning for detection, prevention, and mitiga-

tion of cyberbullying. *ACM Transactions on Interactive Intelligent Systems*, 2(3):18:1–18:30, 2012.

Njagi Dennis Gitari, Zhang Zuping, Hanyurwimfura Damien, and Jun Long. A Lexicon-based Approach for Hate Speech Detection. *International Journal of Multimedia and Ubiquitous Engineering*, 10(4):2015–230, 2015.

Birgit Hamp and Helmut Feldweg. GermaNet - a Lexical-Semantic Net for German. In *Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pages 9–15, Madrid, Spain, 1997.

Thorsten Joachims. Making Large-Scale SVM Learning Practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, pages 169–184. MIT Press, 1999.

Manfred Klenner, Angela Fahrni, and Stefanos Petrakis. PolArt: A Robust Tool for Sentiment Analysis. In *Proceedings of the Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 235–238, Odense, Denmark, 2009.

Christopher D. Manning, Hinrich Schütze, and Prabhakar Raghavan. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In *Proceedings of Workshop at the International Conference on Learning Representations (ICLR)*, Scottsdale, AZ, USA, 2013.

George Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine Miller. Introduction to WordNet: An On-line Lexical Database. *International Journal of Lexicography*, 3:235–244, 1990.

Saif Mohammad and Peter Turney. Crowdsourcing a Word-Emotion Association Lexicon. *Computational Intelligence*, 39(3):555–590, 2013.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. GloVe: Global Vectors for Word Representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Dohar, Qatar, 2014.

Roland Schäfer. Processing and querying large web corpora with the COW14 architecture. In *Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC)*, pages 28–34, Lancaster, United Kingdom, 2015.

Anna Schmidt and Michael Wiegand. A Survey on Hate Speech Detection using Natural Language Processing. In *Proceedings of the EACL-Workshop on Natural Language Processing for Social Media (SocialNLP)*, pages 1–10, Valencia, Spain, 2017.

Vered Shwartz and Chris Waterson. Olive oil is made of oil, baby oil is made for babies: Interpreting noun compounds using paraphrases as in a neural model. In *Proceedings of the Human Language Technology Conference of the North*

American Chapter of the ACL (HLT/NAACL), pages 218–224, New Orleans, LA, USA, 2018.

Sara Owsley Sood, Elizabeth F. Churchill, and Judd Antin. Automatic identification of personal insults on social news sites. *Journal of the Association for Information Science and Technology*, 63(2):270–285, 2012.

Partha Pratim Talukdar, Joseph Reisinger, Marius Pasca, Deepak Ravichandran, Rahul Bhagat, and Fernando Pereira. Weakly-Supervised Acquisition of Labeled Class Instances using Graph Random Walks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 582–590, Honolulu, HI, USA, 2008.

Michael Wiegand and Josef Ruppenhofer. Opinion Holder and Target Extraction based on the Induction of Verbal Categories. In *Proceedings of the Conference on Computational Natural Language Learning (CoNLL)*, pages 215–225, Beijing, China, 2015.

Michael Wiegand, Christine Bocionek, and Josef Ruppenhofer. Opinion Holder and Target Extraction on Opinion Compounds – A Linguistic Approach. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL (HLT/NAACL)*, pages 800–810, San Diego, CA, USA, 2016.

Michael Wiegand, Josef Ruppenhofer, Anna Schmidt, and Clayton Greenberg. Inducing a Lexicon of Abusive Words – A Feature-Based Approach. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL (HLT/NAACL)*, pages 1046–1056, New Orleans, LA, USA, 2018a.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. Overview of the GermEval 2018 Shared Task on the Identification of Offensive Language. In *Proceedings of the GermEval Workshop*, pages 1–10, Vienna, Austria, 2018b.