The Other Side of the Coin: Unsupervised Disambiguation of Potentially Idiomatic Expressions by Contrasting Senses

Hessel Haagsma, Malvina Nissim, Johan Bos
Centre for Language and Cognition, University of Groningen
The Netherlands
{hessel.haagsma, m.nissim, johan.bos}@rug.nl

Abstract

Disambiguation of potentially idiomatic expressions involves determining the sense of a potentially idiomatic expression in a given context, e.g. determining that *make hay* in 'Investment banks *made hay* while takeovers shone.' is used in a figurative sense. This enables automatic interpretation of idiomatic expressions, which is important for applications like machine translation and sentiment analysis. In this work, we present an unsupervised approach for English that makes use of literalisations of idiom senses to improve disambiguation, which is based on the lexical cohesion graph-based method by Sporleder and Li (2009). Experimental results show that, while literalisation carries novel information, its performance falls short of that of state-of-the-art unsupervised methods.

1 Introduction

Interpreting potentially idiomatic expressions (PIEs, for short) is the task of determining the meaning of PIEs in context.[1] In its most basic form, it consists of distinguishing between the figurative and literal usage of a given expression, as illustrated by *hit the wall* in Examples (1) and (2), respectively.

(1) Melanie *hit the wall* so familiar to British youth: not successful enough to manage, but too successful for help. (British National Corpus (BNC; Burnard, 2007) - doc. ACP - sent. 1209)

(2) There was still a dark blob, where it might have *hit the wall*. (BNC - doc. B2E - sent. 1531)

Distinguishing literal and figurative uses is a crucial step towards being able to automatically interpret the meaning of a text containing idiomatic expressions. It has been shown that idiomatic expressions pose a challenge for various NLP applications (Sag et al., 2002), including sentiment analysis (Williams et al., 2015) and machine translation (Salton et al., 2014a; Isabelle et al., 2017). For the latter, it has also been shown that being able to interpret idioms indeed improves performance (Salton et al., 2014b).

In this work, we use a method for unsupervised disambiguation that exploits semantic cohesion between the PIE and its context, based on the lexical cohesion approach pioneered by Sporleder and Li (2009). We extend this method and evaluate it on English data in a comprehensive evaluation framework, in order to answer the following research question: Do contexts enriched with literalisations of idioms provide a useful new signal for disambiguation?

2 Approach

The disambiguation systems presented here[2] are based on the original lexical cohesion graph classifier developed by Sporleder and Li (2009). Their classifier is based on the idea that the words in a PIE will be more cohesive with the words in the surrounding context when used in a literal sense than when used

[1] The task is also known as *token-based idiom detection*.

[2] The code and refined definitions used for implementing these systems are available at https://github.com/hslh/pie-detection.

Proceedings of the Joint Workshop on
Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018), pages 178–184
Santa Fe, New Mexico, USA, August 25-26, 2018.

in a figurative sense. This classifier builds cohesion graphs, i.e. graphs of content word tokens in the PIE and its context, where each pair of words is connected by an edge weighted by the semantic similarity between the two words. If the average similarity of the complete graph is higher than within the context, the PIE component words add to overall cohesiveness and thus imply a literal sense for the PIE. If it is lower, the PIE component words decrease cohesiveness and thus imply a figurative sense. An example of these graphs is shown in Figure 1.

In the original approach, though, it is only tested whether the literal sense fits or not, by comparing the full and pruned graph. However, this does not measure whether the figurative sense fits. Ideally, we would like to compare the fit of the literal and figurative senses directly. We do this by introducing and using *idiom literalisations* (Section 2.2).

2.1 Basic Lexical Cohesion Graph

We reimplement the original lexical cohesion graph method with one major modification: instead of Normalized Google Distance we use cosine similarity between 300-dimensional GloVe word embeddings (Pennington et al., 2014). Furthermore, we adapt specifics of the classifier to optimise performance on the development set. We use only nouns to build the contexts, where the part-of-speech of words is determined automatically using the spaCy PoS-tagger[3], instead of both nouns and verbs. As a context window, we use two sentences of additional context on either side of the sentence containing the PIE. We also remove edges between two PIE component words, since those are the same for all instances of the same type and thus uninformative. Finally, PIEs are only classified as *literal* if average similarity of the pruned graph is 0.0005 higher than that of the whole graph, in order to compensate for overprediction of the *literal* class.

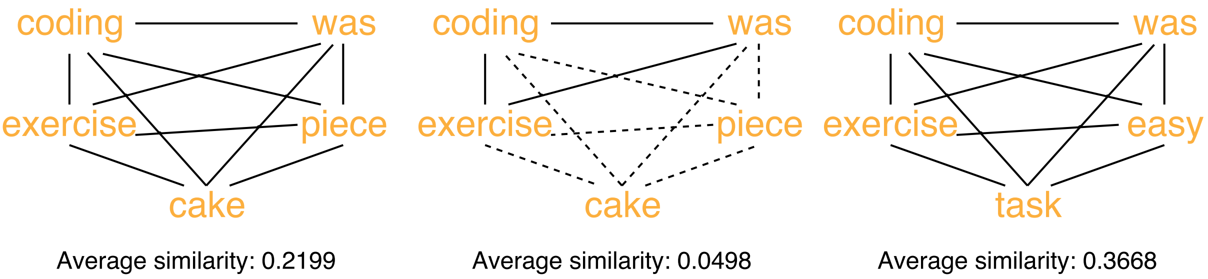
coding
was
exercise
piece
cake
Average similarity: 0.2199
coding
was
exercise
piece
cake
Average similarity: 0.0498
coding
was
exercise
easy
task
Average similarity: 0.3668

Figure 1: Three lexical cohesion graphs for the sentence ‘That coding exercise was a piece of cake’, with their average similarity score. The leftmost figure represents the full graph for the original method, the middle figure the pruned graph, and the right figure the graph containing the idiom literalisation.

2.2 Idiom Literalisation

Idiom literalisations are literal representations of the PIE’s figurative sense, similar to dictionary definitions of an idiom’s meaning. For example, a possible literalisation of *a piece of cake* is ‘a very easy task’. This provides the possibility of building two graphs: one with the original PIE component words, and one with the original PIE replaced with the literalisation of its idiomatic sense. In this way, we can contrast lexical cohesion with a representation of the literal sense to lexical cohesion with a representation of the figurative sense. If the latter is more cohesive, the classifier will label the PIE as idiomatic, and vice versa. Figure 1 illustrates this process; the rightmost graph containing the literalisation has higher cohesion than the original graph, leading to the correct classification of *idiomatic*. Generally, the change in average similarity will be small, since the context words (which stay the same) greatly outnumber the changed PIE component words. However, since we compare the original and the literalised graph directly, only the direction of the similarity change matters and the size of the change is irrelevant.

In this work, we rely on definitions extracted from idiom dictionaries which were manually refined in order to make them more concise. For example, the definition ‘Permanently fixed or firmly established;

[3] https://spacy.io

not subject to any amendment or alteration.' for the idiom *etched in stone* is refined to 'permanently fixed or established', in order to represent the figurative meaning of the idiom more concisely.

3 Experiments

Our research question asks whether literalisations of figurative senses are a useful source of information for improved disambiguation of PIEs. To provide an answer, we test our lexical cohesion graph with and without literalisation on a collection of existing datasets (Section 3.1), and evaluate performance using both micro- and macro-accuracy (Section 3.2).

3.1 Data

In order to provide a comprehensive evaluation dataset, we make use of four sizeable corpora containing sense-annotated PIEs:[4] the VNC-Tokens Dataset (Cook et al., 2008), the IDIX Corpus (Sporleder et al., 2010), the SemEval-2013 Task 5b dataset (Korkontzelos et al., 2013), and the PIE Corpus.[5] An overview of these datasets is provided in Table 1.

	# Types	# Instances	# Sense labels	Source Corpus
VNC-Tokens	53	2,984	3	BNC
IDIX	52	4,022	6	BNC
SemEval-2013 Task 5b	65	4,350	4	ukWaC
PIE Corpus	278	1,050	3	BNC
Combined (development)	299	8,235	2	BNC & ukWaC
Combined (test)	146	3,073	2	BNC & ukWaC

Table 1: Overview of existing corpora of sense-annotated PIEs. The source corpus indicates the corpora from which the PIE instances were selected, either the British National Corpus (Burnard, 2007) or ukWaC (Ferraresi et al., 2008).

Each corpus has slightly different benefits and downsides: VNC-Tokens only contains verb-noun combinations (e.g. *hit the road*) and contains some types which we would not consider idioms (e.g. *have a future*); the IDIX corpus covers various syntactic types and has a large number of instances per PIE type, but is partly singly-annotated; the SemEval dataset is large and varied, but the base corpus, ukWaC (Ferraresi et al., 2008), is noisy; the PIE Corpus covers a very wide range of PIE types, but has only few instances per type and is partly singly-annotated. We combine these four datasets in order to create a more well-rounded dataset. All labels are normalised to a binary sense label. For PIEs with senses which do not fit the binary split, such as *meta-linguistic*, no binary sense label is defined, and we discard those instances. The same goes for false extractions, i.e. sentences included in the corpus not containing any PIEs at all. The combined dataset is split into development and test sets using existing splits of the original datasets. We use the test sets of the original corpora to build the combined test set, which thus consists of: VNC-Test, IDIX-Double, SemEval-*-Test, and PIE-Test. The remaining subsets, including both training and development sets, make up the development set.

3.2 Evaluation

Performance is judged by three evaluation measures: macro-averaged accuracy ('macro-accuracy'), micro-averaged accuracy ('micro-accuracy'), and the harmonic mean of the two. Micro-accuracy reflects how good the disambiguation system is doing overall. Macro-accuracy serves to ensure that we do not just optimise on the most frequent types, since some PIE types are much more frequent than others. By using the harmonic mean of the two, we can rely on a single value to indicate balanced performance.

[4] We do not include the Gigaword-based corpus of Sporleder and Li (2009), since we currently do not have a license for Gigaword.

[5] Corpus available at `https://github.com/hslh/pie-annotation`

3.3 Results & Discussion

The results on the development set are displayed in Table 2. Note that the optimal settings for the literalisation method differ somewhat from those for the original method, as the optimal context window is 5 words on each side, instead of 2 sentences, and the optimal threshold value is $+0.005$, rather than $+0.0005$. These values were optimised by evaluating a range of settings: 0–3 sentences and 1–8 words of context and threshold values from the set $\{-0.05, -0.01, -0.005, +0.0001, +0.0005, +0.001, +0.005, +0.01, +0.05\}$. We also compare to a most frequent sense baseline.

	Macro-Accuracy (%)	Micro-Accuracy (%)	Harmonic Mean (%)
most frequent sense	73.25	57.89	64.67
original	72.74	63.78	67.97
literalisation	71.21	64.94	67.93

Table 2: Results of the original and literalisation-extended cohesion graph classifiers on the combined development set. Accuracy scores are micro- and macro-averages over PIE types, in addition to the harmonic mean of the two.

The graph including literalisations achieves an almost identical score to the original method. It has higher micro-accuracy, but lower macro-accuracy, indicating that it performs better on frequent types than the original classifier. It underpredicts the idiomatic sense quite strongly, which we compensate for by using a higher threshold value.

Although both classifiers show similar performance scores, they make the same judgement in only 5,737 ($\approx 70\%$) of 8,235 instances in the dataset. Additionally, in only $\approx 49\%$ of cases both classifiers are correct, while in $\approx 15\%$ of cases only the original classifier gets it right, and in $\approx 16\%$ of cases, the literalisation classifier predicts the right label. This indicates that the classifiers have at least some partially complementary performances, as they use different information sources, yielding potential for combination; in $\approx 79\%$ of PIE instances, at least one of the classifiers is right.

We combine the classifiers by using the difference in similarity between the two graphs as a confidence value and selecting the classification with the highest confidence value. Rather than improving over the scores of the individual classifiers, this yields a macro-accuracy score of 71.88% and a micro-accuracy of 64.07%, which is squarely in between the scores of the two classifiers. Although the potential for combination is still there, we can conclude that similarity differences do not make reliable confidence values, since larger similarity differences do not correlate with more accurate classifications. As such, the current combination setup yields an average of the two systems, rather than a selection of the best classifications from each system.

By looking at examples with different classifications, we get additional insight into the differences between the two classifiers. In some cases, the differences between the average similarity of the pruned graph, original graph, and the literalised graph are tiny, and any resulting differences are close to random. In other cases, however, both the advantages and disadvantages of using the literalisations come through much more clearly and strongly. Example (3) shows a sentence for which the literalisation graph (3-b) yields the correct classification (idiomatic) and the original graph (3-a) does not. Here, the literalisation is much more cohesive with the context (*feeling-confidence*, 0.69; *feeling-situation*, 0.67) than the idiom (*feeling-feet*, 0.37). Conversely, Example (4) is a case for which only the original gets the correct classification (idiomatic), because of the high similarity between *waste* and *disaster* (0.57), which is lost in the literalisation. The problem here is both related to this specific literalisation, as well as to the more general issue of idioms which are relatively semantically transparent, since those have higher similarity between the context and the component words (in this case, *waste*).

(3) a. In just a couple of days you'll *find your feet* and get that special feeling that you belong in your Club.

b. In just a couple of days you'll *grow in confidence in a new situation* and get that special feeling that you belong in your Club.

(4) a. These figures move slowly around a terrain apparently *laid waste* by some great disaster.
b. These figures move slowly around a terrain apparently *ravage* (sic) by some great disaster.

Splitting out performance by subcorpus shows that the original classifier does better on the VNC-Skewed dataset with a 13% higher score. Conversely, the classifier using literalisations performs about 10 percentage points better on PIE-Train and SemEval-Unknown-Phrases-Dev. The difference on the VNC-Skewed dataset is likely caused by the fact that it contains several frequent types which we would not consider PIEs, such as *catch (someone's) attention* and *have (a) future*.[6] For these items, there is no clear idiomatic sense, so adding literalisations hurts, rather than helps, performance.

Since the graph-based classifiers are optimised on the development set and we report results on that same set, the risk of overfitting exists. Therefore, we evaluate on the unseen combined test set as well. Results in Table 3 indicate that our models generalise well. In absolute terms, performance is very similar to that on the development set. Relative to the most frequent sense baseline, the models do better on the test set than on the development set. In contrast to the identical performance on the development set, the literalisation classifier does better than the original on the test set.

	Macro-Accuracy (%)	Micro-Accuracy (%)	Harmonic Mean (%)
most frequent sense	70.22	55.47	61.98
original	69.68	65.66	67.61
literalisation	69.80	69.18	69.49

Table 3: Results of the original and literalisation classifiers on the combined test set, compared to a most frequent sense baseline. Accuracy scores are micro- and macro-averages over PIE types, in addition to the harmonic mean of the two.

4 Comparison to Related Work

Ideally, we would be able to compare approaches from different papers directly, but this is often impossible. The lack of an established evaluation framework means that reported results for PIE disambiguation are often on different (splits of) datasets, obtained in different ways (cross-validation, leave-one-out) using a range of different metrics (micro- and macro-averaged accuracy and F1-score). For example, Sporleder and Li (2009) report micro-accuracy and micro-F1 scores on the Gigaword corpus, whereas Fazly et al. (2009) report macro-accuracy scores on the VNC-Tokens dataset. A potential solution to this problem was provided by SemEval-2013 Task 5b on PIE disambiguation (Korkontzelos et al., 2013), as results from different participants could be directly compared. However, this dataset does not seem to have been used by the community since.

Nevertheless, we compare to two other unsupervised approaches, the canonical form classifier (CForm) by Fazly et al. (2009), and the k-means clustering approach (KMeans) of Gharbieh et al. (2016). The CForm classifier is based on the assumption that idiomatic PIEs show less variability than literal ones. It uses a set of canonical forms for each idiom (e.g. *make a mark*, *make one's mark*), and labels all PIEs occurring in a canonical form as idiomatic, and literal otherwise. The KMeans classifier builds vector representations of both the PIE and its immediate context based on word embeddings, and clusters those using the k-means algorithm. It then uses the CForm classifier to label the clusters, and propagates the majority label for each cluster to all PIEs in that cluster. Both are evaluated on the test set of the VNC dataset using macro-accuracy by Gharbieh et al. (2016), so we do the same with our system to facilitate comparison.

[6] Both items are not in the Oxford Dictionary of English Idioms (Ayto, 2009), nor in Wiktionary (`https://en.wiktionary.org/wiki/Category:English_idioms`), which is a lot less stringent.

On VNC-Test, the original classifier scores 66.11% macro-accuracy and the literalisation classifier scores 64.67%, compared to 73.7% for CForm and 78.1% for KMeans. As such, our classifiers are outperformed by existing systems. One possible explanation is that our classifiers are optimised on a combination of both macro- and micro-accuracy, whereas previous work focuses only on the macro-averaged score. In addition, Gharbieh et al. (2016) optimise their classifier on the test set, whereas it was completely unseen in our case. Nevertheless, even using development set parameter values, their method achieves a macro-accuracy of 76.5%, which is still clearly higher.

5 Conclusion

In this work, we reimplemented and optimised the original lexical cohesion graph classifier for disambiguation of potentially idiomatic expressions and extended it to make use of literalisations of PIE's figurative senses. By evaluating the systems in a comprehensive evaluation setup, we aimed to answer questions about the contribution of literalisations as an information source.

We have found that the current approach comparing the cohesion of PIEs and their literalisations by itself is not enough to outperform the original lexical cohesion graph classifier. However, both classifiers do well on different subsets of the data, meaning that literalisations are a complementary novel information source and that there is potential for combining the two types of classification to achieve better performance. Using average similarity differences as confidence values to pick one classifier over the other for a particular instance proved ineffective, but a more advanced setup combining the features of the two classifiers could yield a more effective combination.

Moreover, literalisations are cheap to acquire and are available for many PIE types and for any language for which an idiom dictionary exists. Although we use manually created definitions, these can also be acquired and refined automatically, as done by Liu and Hwa (2016). A side benefit of considering both figurative and literal senses is that separate scores can be assigned for both senses. This could be used for detecting difficult cases like dual meanings or puns, since those cases would get high scores for both senses.

In future work, our aim is to further improve these cohesion graph-based classifiers by exploring different similarity measures, such as those tested by Ehren (2017) for German. Another promising avenue is to use more compositional representations of contexts and literalisations (see also Gharbieh et al., 2016). This would also allow us to use the information from verbs and modifiers more effectively, as in its current form our method relies on word-to-word comparisons and only nouns contribute to performance. Finally, we find that, for evaluation, using both micro- and macro-averaged metrics is an important way of ensuring balanced performance on both infrequent and frequent PIE types, in addition to using a wide range of corpora.

Acknowledgements

This work was funded by the NWO-VICI grant "Lost in Translation – Found in Meaning" (288-89-003). We would also like to thank the anonymous reviewers for their useful comments, which helped to significantly improve this paper.

References

John Ayto, editor. 2009. *From the horse's mouth: Oxford dictionary of English Idioms*. Oxford University Press, Oxford; New York, 3rd edition.

Lou Burnard. 2007. Reference guide for the British National Corpus (XML edition).

Paul Cook, Afsaneh Fazly, and Suzanne Stevenson. 2008. The VNC-Tokens dataset. In *Proceedings of the LREC workshop towards a shared task for Multiword Expressions*, pages 19–22.

Rafael Ehren. 2017. Literal or idiomatic? Identifying the reading of single occurrences of German multiword expressions using word embeddings. In *Proceedings of the Student Research Workshop at the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 103–112.

Afsaneh Fazly, Paul Cook, and Suzanne Stevenson. 2009. Unsupervised type and token identification of idiomatic expressions. *Computational Linguistics*, 35(1):61–103.

Adriano Ferraresi, Eros Zanchetta, Marco Baroni, and Silvia Bernardini. 2008. Introducing and evaluating ukWaC, a very large web-derived corpus of English. In *Proceedings of LREC*, pages 47–54.

Waseem Gharbieh, Virendra C. Bhavsar, and Paul Cook. 2016. A word embedding approach to identifying verb-noun idiomatic combinations. In *Proceedings of the 12th Workshop on Multiword Expressions*, pages 112–118.

Pierre Isabelle, Colin Cherry, and George Foster. 2017. A challenge set approach to evaluating machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2476–2486. Association for Computational Linguistics.

Ioannis Korkontzelos, Torsten Zesch, Fabio Massimo Zanzotto, and Chris Biemann. 2013. SemEval-2013 task 5: Evaluating phrasal semantics. In *Proceedings of SemEval*, pages 39–47.

Changsheng Liu and Rebecca Hwa. 2016. Phrasal substitution of idiomatic expressions. In *Proceedings of NAACL*, pages 363–373.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of EMNLP 2014*, pages 1532–1543.

Ivan A. Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. 2002. Multiword expressions: A pain in the neck for NLP. In *Proceedings of CICLING*, pages 1–15.

Giancarlo D. Salton, Robert J. Ross, and John D. Kelleher. 2014a. An empirical study of the impact of idioms on phrase based statistical machine translation of English to Brazilian-Portuguese. In *Proceedings of the 3rd Workshop on Hybrid Approaches to Translation (HyTra)*, pages 36–41.

Giancarlo D. Salton, Robert J. Ross, and John D. Kelleher. 2014b. Evaluation of a substitution method for idiom transformation in statistical machine translation. In *Proceedings of the 10th Workshop on Multiword Expressions*, pages 38–42.

Caroline Sporleder and Linlin Li. 2009. Unsupervised recognition of literal and non-literal use of idiomatic expressions. In *Proceedings of EACL*, pages 754–762.

Caroline Sporleder, Linlin Li, Philip John Gorinski, and Xaver Koch. 2010. Idioms in context: The IDIX corpus. In *Proceedings of LREC*, pages 639–646.

Lowri Williams, Christian Bannister, Michael Arribas-Ayllon, Alun Preece, and Irena Spasić. 2015. The role of idioms in sentiment analysis. *Expert Systems with Applications*, 42(21):7375–7385.