Why “dark thoughts” aren't really dark: A novel algorithm for metaphor identification

Conference Paper · April 2013

DOI: 10.1109/CCMB.2013.6609166

Why "Dark Thoughts" aren't really Dark: A Novel Algorithm for Metaphor Identification

Dan Assaf
Ben-Gurion University of the Negev
Beersheva, Israel

Newton Howard
Massachusetts Institute of Technology
Boston, MA, United States

Yair Neuman*
Ben-Gurion University of the Negev

Mark Last
Ben-Gurion University of the Negev

Yohai Cohen
Gilasio Coding, Ltd.

Ophir Frieder
Georgetown University
Washington, D.C., United States

Shlomo Argamon
Illinois Institute of Technology
Chicago, IL, United States

Moshe Koppel
Bar-Ilan University
Tel Aviv, Israel

Abstract— **Distinguishing between literal and metaphorical language is a major challenge facing natural language processing. Heuristically, metaphors can be divided into three general types in which type III metaphors are those involving an adjective-noun relationship (e.g. "dark humor"). This paper describes our approach for automatic identification of type III metaphors. We propose a new algorithm, the Concrete-Category Overlap (CCO) algorithm, that distinguishes between literal and metaphorical use of adjective-noun relationships and evaluate it on two data sets of adjective-noun phrases. Our results point to the superiority of the CCO algorithm to past and contemporary approaches in determining the presence and conceptual significance of metaphors, and provide a better understanding of the conditions under which each algorithm should be applied.**

*Keywords—**computational intelligence, natural language processing, metaphor, computational linguistics***

I. Introduction

A conceptual metaphor is a cognitive mechanism whereby one conceptual domain is used to understand another conceptual domain [Lakoff and Johnson, 1980/1999]. Although the notion may appear intuitive to humans who regularly use metaphors in everyday speech, replicating the human ability to detect and perceive metaphors in the computational realm is a significant obstacle for linguists and computer scientists alike. For instance, the metaphor "My lawyer is a shark" explains certain characteristics of the lawyer, a human, as being the same or similar to qualities of a shark, which belongs to another conceptual category. Since sharks have many qualities other than those known to be relevant to the metaphor, as do lawyers, determining not only whether the statement is a metaphor but its significance is a complex but potentially valuable computational undertaking. The ubiquity of metaphors in human thinking [Lakoff and Johnson, 1980/1999] and their importance for guiding human reasoning have turned them into an important object of research in fields ranging from psychology to natural language processing [e.g. Thibodeau, & Boroditsky, 2011; Birke & Sarkar, 2007; Gibbs et al. 2004; Kintsch, 2000; Krishnakumaran & Zhu, 2007; Shutova, 2010; Neuman & Nave, 2009; Turney et al. 2011]. Our goal is to automatically identify metaphorical expression with the ultimate aim of analyzing their meanings and better understanding the people who use them.

Following Krishnakumaran & Zhu's approach, we restrict the scope of this paper to metaphoric usages involving nouns. Those authors differentiate between type I, II, and III metaphors. In type I, two nouns are identified via a form of the verb 'to be' such as in the case of "She is a fox." For type II metaphor the verb is the focus of the metaphorical use and acts on the noun such as in the case of "The tragedy sucked the life out of him." Type III metaphors involve an adjective-noun phrase such as "sweet child". Although the Krishnakumaran & Zhu's categorization of noun-based metaphors does not fully exhaust the wide spectrum of metaphorical language, it is a pragmatic heuristic that guides the development of algorithms for metaphor identification. Our focus herein is on type III metaphors only [Krishnakumaran & Zhu (2007)].

There have been different approaches for the identification of metaphors ranging from Word Sense Disambiguation to the use of words' categorization [Birke and Sarkar, 2007; Krishnakumaran & Zhu, 2007)]. Our starting point is Turney's Concrete-Abstract algorithm, which gives state-of-the-art accuracy for this task. The rationale of this algorithm is as follows [Turney et al., 2011]. Type III metaphors, comprised of adjective-noun phrases as in "dark thoughts", generally involve the use of a concrete concept ("dark") to describe a more abstract concept ("thought"). Therefore, for the task of distinguishing between metaphorical and literal phrases, Turney used only one element: the abstractness rating of the noun in the phrase [ibid]. The basis of Turney's argument was that the fact that the noun is abstract indicates that the adjective is likely used in a concrete sense to explain the meaning of the noun, and therefore the phrase functions as a metaphor. Tested on a list of 100 phrases, the algorithm resulted in an average of 79% accuracy [ibid].

The abstractness of different words was estimated based on its distributional similarity and dissimilarity with prototypical abstract and concrete words. As this procedure relies on a vector space representation of word meaning, it is prone to inherent problems with such methods [Turney & Pantel, 2010] such as contextual influences on the abstractness score of a given word. For instance, although "cat" and "American eagle" are both concrete objects belonging to the same category (Animal), "cat" scores 0.29 on the abstractness scale while "American eagle" scores 0.47. The reason is probably the fact that the "American eagle" is also often used as a symbol, and so is contextually associated with more abstract concepts in texts.

Moreover, Concrete-Abstract assumes that the abstractness level of the noun is indicative of metaphorical relations. However, the noun may be relatively concrete and still constitute a metaphorical phrase. For instance, the metaphor "broken heart" includes a relatively concrete concept "heart" that scores 0.37 in Turney's abstractness scale. As the words rated by the abstractness scale have not been disambiguated, the concrete "heart" is used in the sense of the body organ, while the "heart" in "broken heart" is used in the sense of emotion, kindness and spirit. We give herein a new algorithm for metaphor identification designed to remedy the problems associated with the Concrete-Abstract algorithm.

II. Methodology

We assume, as do Turney et al. [2009], that a metaphor usually involves a mapping from a relatively concrete domain to a relatively abstract domain. However, we consider the importance of considering what those specific conceptual domains are. Literal use of a concrete adjective will tend to be more salient with regards to certain semantic categories of concrete objects and not others. For instance, in its literal use, the word "dark" may be associated with certain word categories such as Physical Object (e.g. "table") or Body Part (e.g. "hair"). This notion leads directly to the Concrete Category Overlap (CCO) we propose here, which assumes that if the noun modified by an adjective, *or head noun*, belongs to one of the concrete categories associated with the literal use of the adjective then it is probably literal and otherwise is probably metaphorical. In a sense, this new method combines the notion of measuring concreteness and that of using selectional preferences, as has been well-explored in previous work on metaphor identification [Wilks, 1975]. This hybrid approach, however, overcomes the well-known issues of the pure selectional preferences approach [Fass, 1991; Shutova, 2010], in particular its tendency to over-generate metaphor hypotheses and be misled by common conventionalized metaphors.

The general CCO algorithm is as follows:
Input: An adjective-noun pair <*A*,*N*>.

1. Let **N**(*A*) be the θ_{num} nouns most frequently modified by *A*, with mutual information of at least θ_{MI}.
2. Let $\mathbf{N^C}$ be the κ most concrete nouns in **N**(*A*).
3. Let **Cat**(*A*) be the set of all semantic noun categories containing at least θ_{cat} nouns in $\mathbf{N^C}$.
4. If *N* belongs to one of the categories in **Cat**(*A*), then return LITERAL, else return METAPHORICAL

The basic idea is that an adjective is assigned to a set of semantic categories based on the most frequent concrete nouns that it modifies. We require these pairs to have a minimum mutual information as well, to ensure that the nouns are closely associated with the adjective. In our experiments, we identified these nouns by simple collocation in the Corpus of Contemporary American English (COCA) [Davies, 2009], setting θ_{num} to 1000 and θ_{MI} to 3. The most concrete nouns are identified based on the abstraction scale developed by Turney et al. [2010], which provides a degree measure from 0 to 1.0 with lower values being more concrete and higher values being more abstract. We have found a κ of 100 to work well, though (as we show below) results are not very sensitive to the exact value.

To classify noun semantics, we use the WordStat noun categorization (see the appendix) based on WordNet, which classifies 69,817 nouns into 25 categories, of which 13 are concrete categories (e.g. artifact). As we selected the most concrete nouns, we expect them to be categorized only in these 13 categories. Based on statistics calculated from the most frequent 10,000 nouns in COCA, we found that on average each noun is assigned to two categories. Therefore, if we randomly assign 100 concrete nouns into the 13 concrete categories we would expect, on average, 15.4 words in each category. We thus set **Cat**(*A*) to be all categories containing at least θ_{cat}=16 nouns from $\mathbf{N^C}$. This helps avoid choosing categories that do not really represent literal use of the adjective *A*.

III. Data Sets

A. Data Set 1

Our first data set is an extended version of the 100 phrases corpus used by Turney et al. [2011]. 500 adjective-noun phrases were drawn from COCA according to the same procedure used by Turney, which involved indexing a corpus of gathered text data, searching for specific vocabulary words, and building a word-context frequency matrix from adjacent words. The set included 38 different adjectives (and 321 nouns) with a clear embodied base (e.g., warm, soft, deep, big, sour). Idioms were removed (based on Wiktionary [www.wiktionary.org], leaving 433 adjective-noun pairs. Ground-truth labels (metaphorical/literal) were based on the judgment of two experts in metaphor analysis, who used the Wiktionary definitions as a point of reference to determine whether the adjective is used in its most salient embodied/concrete sense or in a secondary, extended metaphorical sense. For instance, in the case of "bitter lemon" the first embodied definition of the adjective "bitter" is "Having an acrid taste (usually from a basic substance)". When asked to judge whether the phrase "bitter relations" is literal or metaphorical the judges used the basic denotation of "bitter" to make a decision; as “relations” cannot have an acrid taste, the phrase is judged as metaphorical. Disagreements between the judges were resolved through reconciliation. Inter-annotator agreement was 96%. In the final version of this data set, 215 phrases were judged as literal and 218 as metaphorical.

B. Data Set 2

Since Turney et al.’s Concrete-Abstract algorithm works by examining the abstraction scores of nouns, we would expect it to be considerably less effective for metaphors with less abstract nouns, such as the metaphor “broken heart.” In Data Set 1, nouns in metaphorical phrases scored significantly higher on abstractness than nouns in literal phrases ($\mu = 0.55$, $\sigma = 0.13$ vs. $\mu = 0.32$, $\sigma = 0.09$ respectively). A t-test for independent samples showed a statistically significant difference between the metaphorical and literal conditions ($t(257) = 15.87$, $p = 0.00$). As the fraction of metaphors in the overall type III population with highly abstract nouns is unknown, Concrete-Abstract may have a serious blind spot. To deal with this issue, we also constructed a corpus in which there was no significant difference in abstractness of the nouns comprising the phrases in the metaphorical and literal conditions.

This second data set was constructed based on the sentences and phrases in the Berkeley Master Metaphor List (BMML) [Lakoff 1994]. We used the processed corpus prepared by [Krishnakumaran & Xiaojin, 2007] and parsed each sentence using Stanford Dependency Parser [de Marneffe & Manning, 2008]. Adjective-noun pairs related by the syntactic *amod* relation were extracted, and manual expert analysis identified literal and non-literal adjective-noun phrases using the methodology described above. We then extended the set by retrieving adjective-noun phrases from COCA associated with several head nouns from the first set. The final dataset includes 182 phrases from both the BMML and COCA, out of which 95 are literal and 87 metaphorical. In this corpus there was no statistically significant difference between the abstractness level of the nouns comprising the metaphorical and literal conditions, hence we expect the Concrete-Abstract algorithm to not perform well on it.

IV. Results

We reproduced the Concrete-Abstract algorithm by logistic regression for literal/metaphorical classification based on the noun abstractness score, evaluated by 10-fold cross-validation. The CCO algorithm requires no training, and we also evaluated a hybrid algorithm, using logistic regression including noun abstractness with the output of CCO. Results on Data Set 1 are given in Table 1.

TABLE I.

Results for Data Set 1			
Method	**Accuracy**	**Precision**	**Recall**
Concrete-Abstract	85.0%	87.6%	88.4%
CCO	82.5%	88.6%	**90.2%**
Combined	**88.7%**	**90.4%**	86.7%

These replicate the findings of Turney et al. [2011] and also show that CCO, in combination with Concrete-Abstract, slightly improves metaphor identification. Results for Data Set 2 are given in Table 2.

TABLE II.

Results for Data Set 2			
Method	**Accuracy**	**Precision**	**Recall**
Concrete-Abstract	48.6%	44.2%	26.7%
CCO	**73.6%**	**76%**	**65.5%**
Combined	73.2%	75.6%	65.1%

As expected, the Concrete-Abstract was unable to provide any significant prediction of the criterion. This is an important finding as it exposes a blind spot of the state-of-the-art algorithm. In contrast CCO provided a noticeable improvement over the Concrete-Abstract baseline. Finally, we considered results for a combined data set, including all 615 phrases from both Data Sets 1 and 2. Results are given in Table 3.

TABLE III.

Results for Data Set 1 & 2 together			
Method	**Accuracy**	**Precision**	**Recall**
Concrete-Abstract	74.6%	74.9%	73.4%
CCO	**79.8%**	**84.9%**	72.1%
Combined	78.4%	79.5%	**75.8%**

CCO is clearly preferred to Concrete-Abstract in this case, and in fact, combining the methods produces no noticeable improvement. Finally, we consider the sensitivity of the CCO algorithm to the number of concrete nouns κ used. We tested the algorithm for κ = 50, 100, 150 and 200; on the 615 phrases in the combined dataset. Results are presented in Table 4.

TABLE IV.

Sensitivity of CCO to choice of κ			
κ	**Accuracy**	**Precision**	**Recall**
50	77%	83%	67%
100	**80%**	**85%**	72%
150	**80%**	**85%**	**73%**
200	78%	84%	69%

As can be seen, results are not very sensitive to the exact choice of κ as long as it is around the range of 100-150. In our tests, we preferred using the smaller end of this range to ensure that enough concrete nouns could be found for all adjectives.

V. CONCLUSION

Our aim in this paper was to present a new algorithm for identifying type III metaphors, based on combining selectional preference cues with the concrete/abstract criterion developed by Turney et al. [2011]. This algorithm, CCO, has been shown to work well, even in cases where the Concrete-Abstract algorithm, fails utterly. As it is still unclear what fraction of naturally-occurring metaphorical expressions fall into the "easy" and "hard" cases for any particular approach, we view the CCO algorithm as just the first of an ensemble of methods to be developed for identifying different sorts of metaphorical expressions. As CCO is based on a general idea, it is language-independent in principle; we are currently working on extending it to languages other than English. However, we note that languages evolve based on unique cultural and historical factors and it is highly unlikely that the complexity of metaphors can be fully captured by any one universal form of reasoning. Indeed, some metaphors have a cultural-historical logic that cannot be fully captured by our algorithms [Trim, 2007] and improving the identification of type III metaphors across language/cultural contexts will ultimately need to take into account various irreducible processes that turn literal into metaphorical use.

ACKNOWLEDGMENT

This work is supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense US Army Research Laboratory contract number W911NF-12-C-0021. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U.S. Government. The authors would like to thank Peter Turney for constructive reading of a previous draft of this paper. We would also like to thank Dr. Sai Suresh Krishnakumaran for providing access to his Master Metaphors corpus.

REFERENCES

[1] Birke, J. & Sarkar, A. (2007). Active learning for the identification of nonliteral language. In Proceedings of theWorkshop on Computational Approaches to Figurative Language at HLT/NAACL-07, pages 21–28.

[2] Davies, M. 2009. The 385+ million word Corpus of Contemporary American English (1990–2008+): Design, architecture, and linguistic insights. International Journal of Corpus Linguistics, 14(2):159–190.

[3] de Marneffe M-C & Manning CD: The Stanford typed dependencies representation. In COLING Workshop on Cross-framework and Cross-domain Parser Evaluation, 2008.

[4] Fass. D.1991. met*: A method for discriminating metonymy and metaphor by computer. Computational Linguistics, 17(1):49–90.

[5] Gibbs, R.W., Costa-Lima, P. L., & Francozo, E. (2004). Metaphor is grounded in embodied experience. J. Pragmatics, 36, 1189-1210.

[6] Kintsch, W. (2000) Metaphor comprehension: A computational theory, Psychonomic Bulletin & Review, 7, 257-266.

[7] Krishnakumaran, S., & Xiaojin, Z. 2007. Hunting elusive metaphors using lexical resources. In Proceedings of the Workshop on Computational Approaches to Figurative Language (FigLanguages '07). Association for Computational Linguistics, Stroudsburg, PA, USA, 13-20.

[8] Lakoff, G. (1994). The Master Metaphor List. http://cogsci.berkeley.edu/, University of California, Berkeley.

[9] Lakoff, G., & Johnson, M. (1980). Metaphors we live by. Chicago: University of Chicago Press.

[10] Lakoff, G., & Johnson, M. (1999). Philosophy in the flesh. N.Y.: Basic Books.

[11] Neuman, Y., & Nave, O. (2009). Metaphor-based meaning excavation. Information Sciences, 179, 2719-2728.

[12] Shutova, E. (2010). Models of metaphor in NLP. Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, July 11-16, (pp. 688-697).

[13] Thibodeau, P. H., & Boroditsky, L. (2011). Metaphors we think with: The role of metaphor in reasoning. PLoS ONE 6(2), E16782. Doi: 10.1371/journal.pone.0016782.

[14] Trim. R. (2007). Metaphor networks: The comparative evolution of figurative language. UK: Palgrave MacMillan.

[15] Turney P. D. and Pantel. P. 2010. From frequency to meaning: Vector space models of semantics. Journal of Artificial Intelligence Research, 37:141–188.

[16] Turney, P., Neuman, Y. Assaf, D., & Cohen, Y. (2011). Literal and metaphorical sense identification through concrete and abstract context. Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pages 680–690, Edinburgh, Scotland, UK, July 27–31.

[17] Wilks. Y. 1975. A preferential pattern-seeking semantics for natural language inference. Artificial Intelligence, 6:53–74.

Appendix: WordStat Noun Categories

Table is based on *http://www.provalisresearch.com/wordstat/WordNet.html*

Concrete categories are shown in boldface.

Category	Examples	Total Words
Act	abolition, badminton, diagnosis	3879
Animal	abalone, bacteria, coyote	4826
Artifact	accelerator, aquarium, candlestick	5641
Attribute	adequacy, assertiveness, cadence	2170
Body	abdomen, bronchus, collagen	853
Cognition	activism, amnesia, covariance	1664
Communication	alexandrine, allusion, cantata	3022
Event	bonfire, conjuncture, diving	714
Feeling	ambivalence, appetite, cynicism	366
Food	appetizer, beer, borsch	1203
Group	army, bolshevism, Benelux	829
Location	Afghanistan, Alsace, Babylonia	1833
Motive	agromania, egomania, incentive	30
Object	abyss, electron, granule	876
Person	adjuster, correspondent, creator	7366
Phenomenon	aftermath, blizzard, depolarization	303
Plant	achillea, buxus, cultivar	3896
Possession	benefaction, coinage, fellowship	420
Process	absorption, autoregulation, catalysis	503
Quantity	ampere, baud, carat	647
Relation	causality, fatherhood, relevance	226
Shape	azimuth, cuboid, parabola	216
State	aberrance, affluence, homelessness	2158
Substance	Aldol, alkyd, cellulose	1694
Time	bedtime, days, December	461