Classifying Idiomatic and Literal Expressions Using Vector Space Representations

Jing Peng
Montclair State University
pengj@mail.montclair.edu

Anna Feldman
Montclair State University
feldmana@mail.montclair.edu

Hamza Jazmati
Montclair State University
jazmatih1@mail.montclair.edu

Abstract

We describe an algorithm for automatic classification of idiomatic and literal expressions. Our starting point is that idioms and literal expressions occur in different contexts. Idioms tend to violate cohesive ties in local contexts, while literals are expected to fit in. Our goal is to capture this intuition using a vector representation of words. We propose two approaches: (1) Compute inner product of context word vectors with the vector representing a target expression. Since literal vectors predict well local contexts, their inner product with contexts should be larger than idiomatic ones, thereby telling apart literals from idioms; and (2) Compute literal and idiomatic scatter (covariance) matrices from local contexts in word vector space. Since the scatter matrices represent context distributions, we can then measure the difference between the distributions using the Frobenius norm. We provide experimental results validating the proposed techniques.

1 Introduction

Despite the common belief that idioms are always idioms, potentially idiomatic expressions, such as *hit the sack* can appear in literal contexts. Fazly et al. (2009)'s analysis of 60 idioms from the British National Corpus (BNC) has shown that close to half of these also have a clear literal meaning; and of those with a literal meaning, on average around 40% of their usages are literal. Therefore, idioms present great challenges for many Natural Language Processing (NLP) applications. Most current translation systems rely on large repositories of idioms. In this paper we describe an algorithm for automatic classification of idiomatic and literal expressions. Similarly to Peng et al. (2014), we treat idioms as semantic outliers. Our assumption is that the context word distribution for a literal expression will be different from the distribution for an idiomatic one. We capture the distribution in terms of covariance matrix in vector space.

2 Previous Work

Previous approaches to idiom detection can be classified into two groups: 1) type-based extraction, i.e., detecting idioms at the type level; 2) token-based detection, i.e., detecting idioms in context. Type-based extraction is based on the idea that idiomatic expressions exhibit certain linguistic properties such as non-compositionality that can distinguish them from literal expressions (Sag et al., 2002; Fazly et al., 2009). While many idioms do have these properties, many idioms fall on the continuum from being compositional to being partly unanalyzable to completely non-compositional (Cook et al., 2007). Katz and Giesbrech (2006), Birke and Sarkar (2006), Fazly et al. (2009), Sporleder and Li (2009), Li and Sporleder (2010), among others, notice that type-based approaches do not work on expressions that can be interpreted idiomatically or literally depending on the context and thus, an approach that considers tokens in context is more appropriate for idiom recognition.To address these problems, Peng et al. (2014) investigate the bag of words *topic* representation and incorporate an additional hypothesis– contexts in which idioms occur are more affective. Still, they treat idioms as semantic outliers.

3 Proposed Techniques

We hypothesize that words in a given text segment that are representatives of a common topic of discussion are likely to associate strongly with a literal expression in the segment, in terms of projection (or inner product) of word vectors onto the

vector representing the literal expression. We also hypothesize that the context word distribution for a literal expression in word vector space will be different from the distribution for an idiomatic one.

3.1 Projection Based On Local Context Representation

The local context of a literal target verb-noun construction (VNC) must be different from that of an idiomatic one. We propose to exploit recent advances in vector space representation to capture the difference between local contexts (Mikolov et al., 2013a; Mikolov et al., 2013b).

A word can be represented by a vector of fixed dimensionality q that best predicts its surrounding words in a sentence or a document (Mikolov et al., 2013a; Mikolov et al., 2013b). Given such a vector representation, our first proposal is the following. Let v and n be the vectors corresponding to the verb and noun in a target verb-noun construction, as in *blow whistle*, where $v, n \in \Re^q$. Let $\sigma_{vn} = v + n \in \Re^q$. Thus, σ_{vn} is the word vector that represents the composition of verb v and noun n, and in our example, the composition of *blow* and *whistle*. As indicated in (Mikolov et al., 2013b), word vectors obtained from deep learning neural net models exhibit linguistic regularities, such as additive compositionality. Therefore, σ_{vn} is justified to predict surrounding words of the composition of, say, *blow* and *whistle*.

For a given vocabulary of m words, represented by matrix $V = [v_1, v_2, \cdots, v_m] \in \Re^{q \times m}$, we calculate the projection of each word v_i in the vocabulary onto σ_{vn}

$$P = V^t \sigma_{vn} \tag{1}$$

where $P \in \Re^m$, and t represents transpose. Here we assume that σ_{vn} is normalized to have unit length. Thus, $P_i = v_i^t \sigma_{vn}$ indicates how strongly word vector v_i is associated with σ_{vn}. This projection forms the basis for our proposed technique.

Let $D = \{d_1, d_2, \cdots, d_l\}$ be a set of l text segments, each containing a target VNC (i.e., σ_{vn}). Instead of generating a term by document matrix, where each term is *tf-idf* (product of term frequency and inverse document frequency), we compute a term by document matrix $M_D \in \Re^{m \times l}$, where each term in the matrix is

$$p \cdot idf, \tag{2}$$

the product of the projection of a word onto a target VNC and inverse document frequency. That is, the term frequency (tf) of a word is replaced by the projection (inner product) of the word onto σ_{vn} (1). Note that if segment d_j does not contain word v_i, $M_D(i, j) = 0$, which is similar to *tf-idf* estimation. The motivation is that topical words are more likely to be well predicted by a literal VNC than by an idiomatic one. The assumption is that a word vector is learned in such a way that it best predicts its surrounding words in a sentence or a document (Mikolov et al., 2013a; Mikolov et al., 2013b). As a result, the words associated with a literal target will have larger projection onto a target σ_{vn}. On the other hand, the projections of words associated with an idiomatic target VNC onto σ_{vn} should have a smaller value.

We also propose a variant of $p \cdot idf$ representation. In this representation, each term is a product of *p* and typical *tf-idf*. That is,

$$p \cdot tf \cdot idf. \tag{3}$$

3.2 Local Context Distributions

Our second hypothesis states that words in a local context of a literal expression will have a different distribution from those in the context of an idiomatic one. We propose to capture local context distributions in terms of scatter matrices in a space spanned by word vectors (Mikolov et al., 2013a; Mikolov et al., 2013b).

Let $d = (w_1, w_2 \cdots, w_k) \in \Re^{q \times k}$ be a segment (document) of k words, where $w_i \in \Re^q$ are represented by a vectors (Mikolov et al., 2013a; Mikolov et al., 2013b). Assuming w_is have been centered, we compute the scatter matrix

$$\Sigma = d^t d, \tag{4}$$

where Σ represents the local context distribution for a given target VNC.

Given two distributions represented by two scatter matrices Σ_1 and Σ_2, a number of measures can be used to compute the distance between Σ_1 and Σ_2, such as Choernoff and Bhattacharyya distances (Fukunaga, 1990). Both measures require the knowledge of matrix determinant. In our case, this can be problematic, because Σ (4) is most likely to be singular, which would result in a determinant to be zero.

We propose to measure the difference between Σ_1 and Σ_2 using matrix norms. We have experimented with the Frobenius norm and the spectral norm. The Frobenius norm evaluates the difference between Σ_1 and Σ_2 when they act on a standard basis. The spectral norm, on the other hand,

evaluates the difference when they act on the direction of maximal variance over the whole space.

4 Experiments

4.1 Methods

We have carried out an empirical study evaluating the performance of the proposed techniques. For comparison, the following methods are evaluated: **1** *tf-idf*: compute term by document matrix from training data with *tf-idf* weighting; **2** *p-idf*: compute term by document matrix from training data with proposed *p-idf* weighting (2); **3** *p*tf-idf*: compute term by document matrix from training data with proposed p*tf-idf weighting (3); **4** $CoVAR_{Fro}$: compute literal and idiomatic scatter matrices from training data (4). For a test example, compute a scatter matrix according to (4). Calculate the distance between the test scatter matrix and training scatter matrices using Frobenius norm; and **5** $CoVAR_{Sp}$: compute literal and idiomatic scatter matrices from training data (4). For a test text segment, compute a scatter matrix according to (4). Calculate the distance between the test scatter matrix and training scatter matrices using the spectral norm.

For methods from **1** to **3**, we compute a latent space from a term by document matrix obtain from the training data that captures 80% variance. To classify a test example, we compute cosine similarity between the test example and the training data in the latent space to make a decision.

4.2 Data Preprocessing

We use BNC (Burnard, 2000)) and a list of verb-noun constructions (VNCs) extracted from BNC by Fazly et al. (2009; Cook et al. (2008) and labeled as L (Literal), I (Idioms), or Q (Unknown). The list contains only those VNCs whose frequency was greater than 20 and that occurred at least in one of two idiom dictionaries (Cowie et al., 1983; Seaton and Macaulay, 2002). The dataset consists of 2,984 VNC tokens. For our experiments we only use VNCs that are annotated as I or L. We only experimented with idioms that can have both literal and idiomatic interpretations.

We use the original SGML annotation to extract paragraphs from BNC. Each document contains three paragraphs: a paragraph with a target VNC, the preceding paragraph and following one.

Since BNC did not contain enough examples, we extracted additional from COCA, COHA and GloWbE (http://corpus.byu.edu/). Two human annotators annotated this new dataset for idioms and literals. The inter-annotator agreement was relatively low (Cohen's kappa = .58); therefore, we merged the results keeping only those entries on which the two annotators agreed.

Table 1: Datasets: Is = idioms; Ls = literals

Expression	Train	Test
BlowWhistle	20 Is, 20 Ls	7 Is, 31 Ls
LoseHead	15 Is, 15 Ls	6 Is, 4 Ls
MakeScene	15 Is, 15 Ls	15 Is, 5 Ls
TakeHeart	15 Is, 15 Ls	46 Is, 5 Ls
BlowTop	20 Is, 20 Ls	8 Is, 13 Ls
BlowTrumpet	50 Is, 50 Ls	61 Is, 186 Ls
GiveSack	20 Is, 20 Ls	26 Is, 36 Ls
HaveWord	30 Is, 30 Ls	37 Is, 40 Ls
HitRoof	50 Is, 50 Ls	42 is, 68 Ls
HitWall	90 Is, 90 Ls	87 is, 154 Ls
HoldFire	20 Is, 20 Ls	98 Is, 6 Ls
HoldHorse	80 Is, 80 Ls	162 Is, 79 Ls

4.3 Word Vectors

For our experiments reported here, we obtained word vectors using the word2vec tool (Mikolov et al., 2013a; Mikolov et al., 2013b) and the text8 corpus. The text8 corpus has more than 17 million words, which can be obtained from `mattmahoney.net/dc/text8.zip`. The resulting vocabulary has 71,290 words, each of which is represented by a $q = 200$ dimension vector. Thus, this 200 dimensional vector space provides a basis for our experiments.

4.4 Datasets

Table 1 describes the datasets we used to evaluate the performance of the proposed technique. All these verb-noun constructions are ambiguous between literal and idiomatic interpretations. The examples below (from the corpora we used) show how these expressions can be used *literally*.

BlowWhistle: *we can immediately turn towards a high-pitched sound such as whistle being blown. The ability to accurately locate a noise* $\cdots$ **LoseHead**: *This looks as eye-like to the predator as the real eye and gives the prey a fifty-fifty chance of losing its head. That was a very nice bull I shot, but I lost his head.* **MakeScene**: $\cdots$ *in which the many episodes of life were originally isolated and there was no relationship between the parts, but at last we must make a unified scene of our whole life.* **TakeHeart**: $\cdots$ *cutting off one of the forelegs at the shoulder so the heart can be taken*

Table 2: Average accuracy of competing methods on 12 datasets

Method	**BlowWhistle**			**LoseHead**			**MakeScene**			**TakeHeart**		
	Pre	Rec	Acc	Pre	Rec	Acc	Pre	Rec	Acc	Pre	Rec	Acc
tf-idf	0.23	0.75	0.42	0.27	0.21	0.49	0.41	0.13	0.33	0.65	0.02	0.11
p-idf	0.29	0.82	0.60	0.49	0.27	0.48	0.82	0.48	0.53	0.90	0.43	0.44
p*tf-idf	0.23	0.99	0.37	0.31	0.30	0.49	0.40	0.11	0.33	0.78	0.11	0.18
$CoVAR_{Fro}$	**0.65**	**0.71**	**0.87**	**0.60**	**0.78**	**0.58**	**0.84**	**0.83**	**0.75**	**0.95**	**0.61**	**0.62**
$CoVAR_{sp}$	0.44	0.77	0.77	0.62	0.81	0.61	0.80	0.82	0.72	0.94	0.55	0.56
	BlowTop			**BlowTrumpet**			**GiveSack**			**HaveWord**		
	Pre	Rec	Acc	Pre	Rec	Acc	Pre	Rec	Acc	Pre	Rec	Acc
tf-idf	0.55	0.93	0.65	0.26	0.85	0.36	0.61	0.63	0.55	0.52	0.33	0.52
p-idf	0.59	0.58	0.68	0.44	0.85	0.69	0.55	0.47	0.62	0.52	0.53	0.54
p*tf-idf	0.54	0.53	0.65	0.33	0.93	0.51	0.54	0.64	0.55	0.53	0.53	0.53
$CoVAR_{Fro}$	**0.81**	**0.87**	**0.86**	**0.45**	**0.94**	**0.70**	**0.63**	**0.88**	**0.72**	0.58	0.49	0.58
$CoVAR_{sp}$	0.71	0.79	0.79	0.39	0.89	0.62	0.66	0.75	0.73	**0.56**	**0.53**	**0.58**
	HitRoof			**HitWall**			**HoldFire**			**HoldHorse**		
	Pre	Rec	Acc	Pre	Rec	Acc	Pre	Rec	Acc	Pre	Rec	Acc
tf-idf	0.42	0.70	0.52	0.37	0.99	0.39	0.91	0.57	0.57	0.79	0.98	0.80
p-idf	0.54	0.84	0.66	0.55	0.92	0.70	0.97	0.83	0.81	0.86	0.81	0.78
p*tf-idf	0.41	0.98	0.45	0.39	0.97	0.43	0.95	0.89	0.85	0.84	0.97	0.86
$CoVAR_{Fro}$	**0.61**	**0.88**	**0.74**	**0.59**	**0.94**	**0.74**	**0.97**	**0.86**	**0.84**	**0.86**	**0.97**	**0.87**
$CoVAR_{sp}$	0.54	0.85	0.66	0.50	0.95	0.64	**0.96**	**0.87**	**0.84**	0.77	0.85	0.73

out still pumping and offered to the god on a plate.
BlowTop: *Yellowstone has no large sources of water to create the amount of steam to blow its top as in previous eruptions.*

5 Results

Table 2 shows the average precision, recall and accuracy of the competing methods on 12 datasets over 20 runs. The best performance is in bold face. The best model is identified by considering precision, recall, and accuracy together for each model. We calculate accuracy by adding true positives and true negatives and normalizing the sum by the number of examples.

As for the individual model performance, the $CoVAR$ model outperforms the rest of the models. Interestingly, the Frobenius norm outperforms the spectral norm. One possible explanation is that the spectral norm evaluates the difference when two matrices act on the maximal variance direction, while the Frobenius norm evaluates on a standard basis. That is, Frobenius measures the difference along all basis vectors. On the other hand, the spectral norm evaluates changes in a particular direction. When the difference is a result of all basis directions, the Frobenius norm potentially provides a better measurement. The projection methods (p-idf and p*tf-idf) outperform tf-idf overall but not as pronounced as $CoVAR$.

Finally, we have noticed that even the best model ($CoVAR_{Fro}$) does not perform as well on certain idiomatic expressions. We hypothesized that the model works the best on highly idiomatic expressions. Idiomaticity is a continuum. Some idioms seem to be more easily interpretable than others. We conducted a small experiment, in which we asked two human annotators to rank VNCs in our dataset as "highly idiomatic" to "easily interpretable/compositional" (in context) on a scale of 5 to 1 (5: highly idiomatic; 1: low idiomaticity). While we cannot make strong claims based on a such small-scale experiment, the results of our pilot study suggest that there is a correlation between the idiomaticity scores and the performance of our model – the highly idiomatic expressions seem to be detected better. We plan to conduct an experiment with more human annotators and on an larger dataset to verify our hypothesis.

6 Conclusions

In our experiments we used a subset of Fazly et al. (2009)'s dataset plus some additional examples extracted from other corpora. Similarly to us, Fazly et al. (2009)'s goal is to determine whether a given VNC is idiomatic or literal in context. Our model is comparable to and often outperforms Fazly et al. (2009)'s unsupervised CForm model. Our method can also be compared with Peng et al. (2014) who also experiment with LDA, use similar data, and frame the problem as classification.

Acknowledgements

This material is based in part upon work supported by the U.S. National Science Foundation under Grant Number 1319846. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Julia Birke and Anoop Sarkar. 2006. A clustering approach to the nearly unsupervised recognition of nonliteral language. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL'06)*, pages 329–226, Trento, Italy.

Lou Burnard, 2000. *The British National Corpus Users Reference Guide*. Oxford University Computing Services.

Paul Cook, Afsaneh Fazly, and Suzanne Stevenson. 2007. Pulling their weight: Exploiting syntactic forms for the automatic identification of idiomatic expressions in context. In *Proceedings of the ACL 07 Workshop on A Broader Perspective on Multiword Expressions*, pages 41–48.

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 (MWE 2008)*, Marrakech, Morocco, June.

Anthony P. Cowie, Ronald Mackin, and Isabel R. McCaig. 1983. *Oxford Dictionary of Current Idiomatic English*, volume 2. Oxford University Press.

Afsaneh Fazly, Paul Cook, and Suzanne Stevenson. 2009. Unsupervised Type and Token Identification of Idiomatic Expressions. *Computational Linguistics*, 35(1):61–103.

K. Fukunaga. 1990. *Introduction to statistical pattern recognition*. Academic Press.

Graham Katz and Eugenie Giesbrech. 2006. Automatic Identification of Non-compositional Multiword Expressions using Latent Semantic Analysis. In *Proceedings of the ACL/COLING-06 Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*, pages 12–19.

Linlin Li and Caroline Sporleder. 2010. Using gaussian mixture models to detect figurative language in context. In *Proceedings of NAACL/HLT 2010*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proceedings of Workshop at ICLR*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*.

Jing Peng, Anna Feldman, and Ekaterina Vylomova. 2014. Classifying idiomatic and literal expressions using topic models and intensity of emotions. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2019–2027, Doha, Qatar, October. Association for Computational Linguistics.

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 the 3rd International Conference on Intelligence Text Processing and Computational Linguistics (CICLing 2002)*, pages 1–15, Mexico City, Mexico.

Maggie Seaton and Alison Macaulay, editors. 2002. *Collins COBUILD Idioms Dictionary*. HarperCollins Publishers, second edition.

Caroline Sporleder and Linlin Li. 2009. Unsupervised Recognition of Literal and Non-literal Use of Idiomatic Expressions. In *EACL '09: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 754–762, Morristown, NJ, USA. Association for Computational Linguistics.