A Word Embedding Approach to Identifying Verb–Noun Idiomatic Combinations

Waseem Gharbieh and **Virendra C. Bhavsar** and **Paul Cook**
Faculty of Computer Science, University of New Brunswick
Fredericton, NB E3B 5A3 Canada
{waseem.gharbieh,bhavsar,paul.cook}@unb.ca

Abstract

Verb–noun idiomatic combinations (VNICs) are idioms consisting of a verb with a noun in its direct object position. Usages of these expressions can be ambiguous between an idiomatic usage and a literal combination. In this paper we propose supervised and unsupervised approaches, based on word embeddings, to identifying token instances of VNICs. Our proposed supervised and unsupervised approaches perform better than the supervised and unsupervised approaches of Fazly et al. (2009), respectively.

1 Verb–noun Idiomatic Combinations

Much research on multiword expressions (MWEs) in natural language processing (NLP) has focused on various type-level prediction tasks, e.g., MWE extraction (e.g., Church and Hanks, 1990; Smadja, 1993; Lin, 1999) — i.e., determining which MWE types are present in a given corpus (Baldwin and Kim, 2010) — and compositionality prediction (e.g., McCarthy et al., 2003; Reddy et al., 2011; Salehi et al., 2014). However, word combinations can be ambiguous between literal combinations and MWEs. For example, consider the following two usages of the expression *hit the roof*:

1. I think Paula might hit the roof if you start ironing.

2. When the blood hit the roof of the car I realised it was serious.

The first example of *hit the roof* is an idiomatic usage, while the second is a literal combination.[1] MWE identification is the task of determining which token instances in running text are MWEs (Baldwin and Kim, 2010). Although there has been relatively less work on MWE identification than other type-level MWE prediction tasks, it is nevertheless important for NLP applications such as machine translation that must be able to distinguish MWEs from literal combinations in context.

Some recent work has focused on token-level identification of a wide range of types of MWEs and other multiword units (e.g., Newman et al., 2012; Schneider et al., 2014; Brooke et al., 2014). Many studies, however, have taken a word sense disambiguation–inspired approach to MWE identification (e.g., Birke and Sarkar, 2006; Katz and Giesbrecht, 2006; Li et al., 2010), treating literal combinations and MWEs as different word senses, and have exploited linguistic knowledge of MWEs (e.g., Patrick and Fletcher, 2005; Uchiyama et al., 2005; Hashimoto and Kawahara, 2008; Fazly et al., 2009; Fothergill and Baldwin, 2012).

In this study we focus on English verb–noun idiomatic combinations (VNICs). VNICs are formed from a verb with a noun in its direct object position. They are a common and productive type of English idiom, and occur cross-lingually (Fazly et al., 2009).

VNICs tend to be relatively lexico-syntactically fixed, e.g., whereas *hit the roof* is ambiguous between literal and idiomatic meanings, *hit the roofs* and *a roof was hit* are most likely to be literal usages. Fazly et al. (2009) exploit this property in their unsupervised approach, referred to as CFORM. They define lexico-syntactic patterns for VNIC token instances based on the noun's determiner (e.g., *a*, *the*, or possibly no determiner), the number of the noun (singular or plural), and the verb's voice (active or passive). They propose a statistical method for automatically determining a given VNIC type's canonical idiomatic form, based on the frequency of its usage in these

[1] These examples, and idiomaticity judgements, are taken from Cook et al. (2008).

Proceedings of the 12th Workshop on Multiword Expressions, pages 112–118,
Berlin, Germany, August 7-12, 2016.

patterns in a corpus.[2] They then classify a given token instance of a VNIC as idiomatic if it occurs in its canonical form, and as literal otherwise. Fazly et al. also consider a supervised approach that classifies a given VNIC instance based on the similarity of its context to that of idiomatic and literal instances of the same expression seen during training.

Distributed representations of word meaning in the form of word embeddings (Mikolov et al., 2013) have recently been demonstrated to benefit a wide range of NLP tasks including POS tagging (e.g., Ling et al., 2015), question answering (e.g., Dong et al., 2015), and machine translation (e.g., Zou et al., 2013). Moreover, word embeddings have been shown to improve over count-based models of distributional similarity for predicting MWE compositionality (Salehi et al., 2015).

In this work we first propose a supervised approach to identifying VNIC token instances based on word embeddings that outperforms the supervised method of Fazly et al. (2009). We then propose an unsupervised approach to this task, that combines word embeddings with Fazly et al.'s unsupervised CFORM approach, that improves over CFORM.

2 Models for VNIC Identification Based on Word Embeddings

The following subsections propose supervised and unsupervised approaches to VNIC identification based on word embeddings.

2.1 Supervised VNIC Identification

For the proposed supervised approach, we first extract features based on word embeddings from word2vec representing a token instance of a VNIC in context, and then use these representations of VNIC tokens to train a supervised classifier.

We first form a vector $\vec{e}$ representing a given VNIC token at the type level. $\vec{e}$ is formed by averaging the embeddings of the lemmatized component words forming the VNIC.

We then form a vector $\vec{c}$ representing the context of the VNIC token instance. MWEs, including VNICs, can be discontiguous. We therefore form two vectors, $\vec{c}_{\text{verb}}$ and $\vec{c}_{\text{noun}}$, representing the context of the verb and noun components, respectively, of the VNIC instance, and then average

[2]In some cases a VNIC may have a small number of canonical forms, as opposed to just one.

Original text: *You can see the stars, now, in the city*

Context tokens for verb (*see*): *you*, *can*, *the*, *now*

Context tokens for noun (*stars*): *can*, *the*, *now*, *in*

$$\vec{c}_{\text{verb}} = \frac{vec(you)+vec(can)+vec(the)+vec(now)}{4}$$

$$\vec{c}_{\text{noun}} = \frac{vec(can)+vec(the)+vec(now)+vec(in)}{4}$$

$$\vec{c} = \frac{\vec{c}_{\text{verb}}+\vec{c}_{\text{noun}}}{2}$$

Figure 1: An example of computing $\vec{c}$ for a window size (k) of 2, where $vec(w)$ is the vector for word w obtained from word2vec.

these vectors to form $\vec{c}$. More precisely, $\vec{c}_{\text{verb}}$ and $\vec{c}_{\text{noun}}$ are formed as follows:

$$\vec{c_j} = \frac{1}{2k} \sum_{i=-k, i\neq 0}^{k} w_{t-i}^{j} \quad (1)$$

where k is the window size that the word2vec model was trained on, and w_t^j is the embedding of the word in position t of the input sentence relative to the jth component of the MWE (i.e., either the verb or noun). In forming $\vec{c}_{\text{verb}}$ and $\vec{c}_{\text{noun}}$ the other component token of the VNIC is not considered part of the context. The summation is done over the same window size that the word2vec model was trained on so that $\vec{c_j}$ captures the same information that the word2vec model has learned to capture. After computing $\vec{c}_{\text{verb}}$ and $\vec{c}_{\text{noun}}$ these vectors are averaged to form $\vec{c}$. Figure 1 shows the process for forming $\vec{c}$ for an example sentence.

Finally, to form the feature vector representing a VNIC instance, we subtract $\vec{e}$ from $\vec{c}$, and append to this vector a single binary feature representing whether the VNIC instance occurs in its canonical form, as determined by Fazly et al. (2009). The feature vectors are then used to train a supervised classifier; in our experiments we use the linear SVM implementation from Pedregosa et al. (2011). The motivation for the subtraction is to capture the difference between the context in which a VNIC instance occurs ($\vec{c}$) and a type-level representation of that expression ($\vec{e}$), to potentially represent VNIC instances such that the classifier is able to generalize across expressions (i.e., to generalize to MWE types that are unseen during training). The canonical form feature is included because it is known to be highly informative as to

whether an instance is idiomatic or literal.

2.2 Unsupervised VNIC Identification

Our unsupervised approach combines the word embedding–based representation used in the supervised approach (without relying on training a supervised classifier, of course) with the unsupervised CFORM method of Fazly et al. (2009). In this approach, we first represent each token instance of a given VNIC type as a feature vector, using the same representation as in Section 2.1.[3] We then apply k-means clustering to form k clusters of the token instances.[4] All instances in each cluster are then assigned a single class, idiomatic or literal, depending on whether the majority of token instances in a cluster are in that VNIC's canonical form or not, respectively. In the case of ties the method backs off to a most-frequent class (idiomatic) baseline. This method is unsupervised in that it does not rely on any gold standard labels.

3 Materials and Methods

In this section we describe training details for the word embeddings and the dataset used for evaluation.

3.1 Word embeddings

The word embeddings required by our proposed methods were trained using the gensim[5] implementation of the skip gram version of word2vec (Mikolov et al., 2013). The model was trained on a snapshot of English Wikipedia from 1 September 2015. The text was pre-processed using wp2txt[6] to remove markup, and then tokenized with the Stanford tokenizer (Manning et al., 2014). Tokens occurring less than 15 times were removed, and the negative sampling parameter was set to 5.

3.2 VNC-Tokens Dataset

The VNC-Tokens dataset (Cook et al., 2008) contains instances of 53 VNIC types — drawn from the British National Corpus (Burnard, 2007) — that have been manually annotated at the token level for whether they are literal or idiomatic usages. The 53 expressions are divided into three subsets: DEV, TEST, and SKEWED. SKEWED consists of 25 expressions that are used primarily idiomatically, or primarily literally, while DEV and TEST consist of 14 expressions each that are more balanced between their idiomatic and literal usages. Fazly et al. (2009) focus primarily on DEV and TEST; we therefore only consider these subsets here. DEV and TEST consist of a total of 597 and 613 VNIC tokens, respectively, that are annotated as either literal or idiomatic usages.[7]

Window	Dimensions	Dev	Test
1	50	87.3	85.9
	100	**88.2**	85.5
	300	86.3	**88.3**
2	50	86.4	84.2
	100	86.7	84.2
	300	86.5	86.7
5	50	86.0	83.4
	100	85.9	84.2
	300	87.3	85.7
8	50	85.5	84.3
	100	85.6	85.9
	300	85.8	86.3
Baseline		62.1	61.9
Fazly et al. (2009) CFORM		72.3	73.7
Fazly et al. (2009) Supervised		80.1	82.7

Table 1: Percent accuracy using a linear SVM for different word2vec parameters. Results for a most-frequent class baseline, and the CFORM and supervised methods from Fazly et al. (2009), are also shown.

4 Experimental Results

In the following subsections we describe the results of experiments using our supervised approach, the ability of this method to generalize across MWE types, and finally the results of the unsupervised approach.

4.1 Supervised Results

Following Fazly et al. (2009), the supervised approach was evaluated using a leave-one-token-out strategy. That is, for each MWE, a single token instance is held out, and the classifier is trained on the remaining instances. The trained model is then used to classify the held out instance. This is

[3] Based on results in preliminary experiments we found that normalizing the feature vectors led to modest improvements in this case.

[4] In our experiments we use the implementation of k-means clustering from Pedregosa et al. (2011).

[5] https://radimrehurek.com/gensim/

[6] https://github.com/yohasebe/wp2txt

[7] Both DEV and TEST also contain instances that are annotated as "unknown"; following Fazly et al. (2009) we exclude these instances from our study.

	CFORM		Oracle	
k	Dev	Test	Dev	Test
2	67.8 $\pm$3.13	64.2 $\pm$2.57	82.6 $\pm$0.65	81.5 $\pm$2.86
3	68.2 $\pm$4.36	71.1 $\pm$2.99	84.2 $\pm$2.94	83.2 $\pm$2.58
4	69.7 $\pm$5.24	**78.1** $\pm$3.30	86.0 $\pm$3.02	85.9 $\pm$2.82
5	**71.8** $\pm$6.58	76.5 $\pm$4.07	86.9 $\pm$3.54	87.9 $\pm$2.36

Table 2: The percent accuracy, and standard deviation, of our unsupervised approach incorporating CFORM (left), and an oracle (right), for differing values of k.

repeated until all the instances of the MWE type have been classified. The idiomatic and literal classes have roughly comparable frequencies in the dataset, therefore, again following Fazly et al., macro-averaged accuracy is reported.[8] Nevertheless, the idiomatic class is more frequent; therefore, also following Fazly et al., we report a most-frequent class baseline that classifies all instances as idiomatic. Results are shown in Table 1 for a variety of settings of window size and number of dimensions for the word embeddings.

The results reveal the general trend that smaller window sizes, and more dimensions, tend to give higher accuracy, although the overall amount of variation is relatively small. The accuracy on DEV and TEST ranges from 85.5%–88.2% and 83.4%–88.3%, respectively. All of these accuracies are higher than those reported by Fazly et al. (2009) for their supervised approach. They are also substantially higher than the most-frequent class baseline, and the unsupervised CFORM method of Fazly et al.

That a window size of just 1 performs well is interesting. A word2vec model with a smaller window size gives more syntactically-oriented word embeddings, whereas a larger window size gives more semantically-oriented embeddings (Trask et al., 2015). The CFORM method of Fazly et al. (2009) is a strong unsupervised benchmark for this task, and relies on the lexico-syntactic pattern in which an MWE token instance occurs. A smaller window size for the word embedding features might be better able to capture similar information to CFORM, which could explain the good performance of the model using a window size of 1.

4.2 Generalization to Unseen VNICs

We do not expect to have substantial amounts of annotated training data for every VNIC. We therefore further consider whether the supervised approach is able to generalize to MWE types that are unseen during training. Indeed, this scenario motivated the choice of representation of VNIC token instances in Section 2.1. In these experiments we perform a leave-one-type-out evaluation. In this case, all token instances for a single MWE type are held out, and the token instances of the remaining MWE types (limited to those within either DEV or TEST) are used to train a classifier. The classifier is then used to classify the token instances of the held out MWE type. This process is repeated until all instances of all MWE types have been classified.

For these experiments we consider the setup that performed best on average over DEV and TEST in the previous experiments (i.e., a window size of 1 and 300 dimensional vectors). The macro-averaged accuracy on DEV and TEST is 68.9% and 69.4%, respectively. Although this is a substantial improvement over the most-frequent class baseline, it is well-below the accuracy for the previously-considered leave-one-token-out setup. Moreover, the unsupervised CFORM method of Fazly et al. (2009) gives substantially higher accuracies than this supervised approach. The limited ability of this model to generalize to unseen MWE types further motivates exploring unsupervised approaches to this task.

4.3 Unsupervised Results

The k-means clustering for the unsupervised approach is repeated 100 times with randomly-selected initial centroids, for several values of k. The average accuracy and standard deviation of the unsupervised approach over these 100 runs are shown in the left panel of Table 2. For k = 4 and 5 on TEST, this approach surpasses the unsupervised CFORM method of Fazly et al. (2009); however, on DEV this approach does not outperform Fazly et al.'s CFORM approach for any of the val-

[8] This is equivalent to macro-averaged recall.

ues of k considered. Analyzing the results on individual expressions indicates that the unsupervised approach gives especially low accuracy for *hit roof* — which is in DEV— as compared to the CFORM method of Fazly et al., which could contribute to the overall lower accuracy of the unsupervised approach on this dataset.

We now consider the upperbound of an unsupervised approach that selects a single label for each cluster of usages. In the right panel of Table 2 we show results for an oracle approach that always selects the best label for each cluster. In this case, as the number of clusters increases, so too will the accuracy.[9] Nevertheless, these results show that, even for relatively small values of k, there is scope for improving the proposed unsupervised method through improved methods for selecting the label for each cluster, and that the performance of such a method could potentially come close to that of the supervised approach. A word's predominant sense is known to be a powerful baseline in word-sense disambiguation, and prior work has addressed automatically identifying predominant word senses (McCarthy et al., 2007; Lau et al., 2014). The findings here suggest that methods for determining whether a set of usages of a VNIC are predominantly literal or idiomatic could be leveraged to give further improvements in unsupervised VNIC identification.

5 Conclusions

In this paper we proposed supervised and unsupervised approaches, based on word embeddings, to identifying token instances of VNICs that performed better than the supervised approach, and unsupervised CFORM approach, of Fazly et al. (2009), respectively. In future work we intend to consider methods for determining the predominant "sense" (i.e., idiomatic or literal) of a set of usages of a VNIC, in an effort to further improve unsupervised VNIC identification.

Acknowledgments

This work is financially supported by the Natural Sciences and Engineering Research Council of Canada, the New Brunswick Innovation Foundation, and the University of New Brunswick.

[9] When the number of clusters is equal to the number of instances, the accuracy will be 100%.

References

Timothy Baldwin and Su Nam Kim. 2010. Multiword expressions. In Nitin Indurkhya and Fred J. Damerau, editors, *Handbook of Natural Language Processing*, CRC Press, Boca Raton, USA. 2nd edition.

Julia Birke and Anoop Sarkar. 2006. A clustering approach for nearly unsupervised recognition of nonliteral language. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL-2006)*. pages 329–336.

Julian Brooke, Vivian Tsang, Graeme Hirst, and Fraser Shein. 2014. Unsupervised multiword segmentation of large corpora using prediction-driven decomposition of n-grams. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*. Dublin, Ireland, pages 753–761.

Lou Burnard. 2007. Reference guide for the British National Corpus (XML Edition). Oxford University Computing Services.

Kenneth W. Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics* 16(1):22–29.

Paul Cook, Afsaneh Fazly, and Suzanne Stevenson. 2008. The VNC-Tokens Dataset. In *Proceedings of the LREC Workshop on Towards a Shared Task for Multiword Expressions (MWE 2008)*. Marrakech, Morocco, pages 19–22.

Li Dong, Furu Wei, Ming Zhou, and Ke Xu. 2015. Question answering over freebase with multi-column convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*. volume 1, pages 260–269.

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

Richard Fothergill and Timothy Baldwin. 2012. Combining resources for mwe-token classification. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and*

the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012). Montréal, Canada, pages 100–104.

Chikara Hashimoto and Daisuke Kawahara. 2008. Construction of an idiom corpus and its application to idiom identification based on WSD incorporating idiom-specific features. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. Honolulu, Hawaii, pages 992–1001.

Graham Katz and Eugenie Giesbrecht. 2006. Automatic identification of non-compositional multi-word expressions using latent semantic analysis. In *Proceedings of the Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*. Sydney, Australia, pages 12–19.

Jey Han Lau, Paul Cook, Diana McCarthy, Spandana Gella, and Timothy Baldwin. 2014. Learning word sense distributions, detecting unattested senses and identifying novel senses using topic models. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*. Baltimore, USA, pages 259–270.

Linlin Li, Benjamin Roth, and Caroline Sporleder. 2010. Topic models for word sense disambiguation and token-based idiom detection. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Uppsala, Sweden, pages 1138–1147.

Dekang Lin. 1999. Automatic identification of non-compositional phrases. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*. College Park, MD, pages 317–324.

Wang Ling, Yulia Tsvetkov, Silvio Amir, Ramon Fermandez, Chris Dyer, Alan W Black, Isabel Trancoso, and Chu-Cheng Lin. 2015. Not all contexts are created equal: Better word representations with variable attention. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal, pages 1367–1372.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Baltimore, USA, pages 55–60.

Diana McCarthy, Bill Keller, and John Carroll. 2003. Detecting a continuum of compositionality in phrasal verbs. In *Proceedings of the ACL-SIGLEX Workshop on Multiword Expressions: Analysis, Acquisition and Treatment*. Sapporo, Japan, pages 73–80.

Diana McCarthy, Rob Koeling, Julie Weeds, and John Carroll. 2007. Unsupervised acquisition of predominant word senses. *Computational Linguistics* 33(4):553–590.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of Workshop at the International Conference on Learning Representations, 2013*. Scottsdale, USA.

David Newman, Nagendra Koilada, Jey Han Lau, and Timothy Baldwin. 2012. Bayesian text segmentation for index term identification and keyphrase extraction. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*. Mumbai, India, pages 2077–2092.

Jon Patrick and Jeremy Fletcher. 2005. Classifying verb-particle constructions by verb arguments. In *Proceedings of the Second ACL-SIGSEM Workshop on the Linguistic Dimensions of Prepositions and their use in Computational Linguistics Formalisms and Applications*. Colchester, UK, pages 200–209.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research* 12:2825–2830.

Siva Reddy, Diana McCarthy, and Suresh Manandhar. 2011. An empirical study on compositionality in compound nouns. In *Proceedings of the Fifth International Joint Conference on Natural Language Processing (IJCNLP 2011)*. Chiang Mai, Thailand, pages 210–218.

Bahar Salehi, Paul Cook, and Timothy Baldwin. 2014. Using distributional similarity of multi-way translations to predict multiword expression compositionality. In *Proceedings of the*

14th Conference of the EACL (EACL 2014). Gothenburg, Sweden, pages 472–481.

Bahar Salehi, Paul Cook, and Timothy Baldwin. 2015. A word embedding approach to predicting the compositionality of multiword expressions. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado, pages 977–983.

Nathan Schneider, Emily Danchik, Chris Dyer, and Noah A. Smith. 2014. Discriminative lexical semantic segmentation with gaps: Running the mwe gamut. *Transactions of the Association of Computational Linguistics* 2:193–206.

Frank Smadja. 1993. Retrieving collocations from text: Xtract. *Computational Linguistics* 19(1):143–177.

Andrew Trask, David Gilmore, and Matthew Russell. 2015. Modeling order in neural word embeddings at scale. *Journal of Machine Learning Research* 37.

Kiyoko Uchiyama, Timothy Baldwin, and Shun Ishizaki. 2005. Disambiguating Japanese compound verbs. *Computer Speech and Language, Special Issue on Multiword Expressions* 19(4):497–512.

Will Y. Zou, Richard Socher, Daniel Cer, and Christopher D. Manning. 2013. Bilingual word embeddings for phrase-based machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Seattle, Washington, USA, pages 1393–1398.