Automatic Detection of Idiomatic Language

Submitted as part of the requirements for:

CE902 Professional Practice and Research Methodology

**Name**: Jose Juan Zavala Iglesias

**Supervisor**: Aline Villavicencio

**Date**: 22 March 2019

**Abstract**. TODO: *To be written after everything else is done*

**Keywords**: *Multiword Expressions, Verb-Noun Combinations, Idiom Detection*

Table of Contents

[1 Introduction 3](#_Toc3760125)

[2 Literature Review 3](#_Toc3760126)

[2.1 Background 3](#_Toc3760127)

[2.2 Unsupervised Methods 4](#_Toc3760128)

[2.3 Supervised Methods 6](#_Toc3760129)

[3 Goals 9](#_Toc3760130)

[3.1 Objectives 9](#_Toc3760131)

[3.2 Scope and Limitations 10](#_Toc3760132)

[4 Project Description 10](#_Toc3760133)

[4.1 VNC Tokens Dataset 10](#_Toc3760134)

[4.2 Third-Party Tools 11](#_Toc3760135)

[Gensim 11](#_Toc3760136)

[Siamese CBOW 11](#_Toc3760137)

[Skip-thoughts 11](#_Toc3760138)

[Scikit-Learn 11](#_Toc3760139)

[4.3 Project Development Methodology 11](#_Toc3760140)

[5 Evaluation 11](#_Toc3760141)

[6 Project Plan 12](#_Toc3760142)

[Acknowledgements 12](#_Toc3760143)

[References 12](#_Toc3760144)

[Appendix 13](#_Toc3760145)

# Introduction

*// TODO: Mention relevance of idioms detection in translation and semantic parsing*

*// TODO: Idioms appear over time, so we need ever-growing corpora. This motivates unsupervised approach over supervised.*

*// TODO: Write examples of VNCs found in VNC-Token Dataset*

Multiword Expressions (MWEs) are combinations of multiple words that exhibit some degree of idiomaticity, but not necessarily [1]. Verb-noun combinations (VNCs) are a common type of MWEs that consist of a verb with a noun in its direct object position [1].

# Literature Review

Much research on MWE identification focuses on the task of the identification of specific kinds on MWEs, such as English VNCs [1] [2] [3], while other authors focus on multilingual detection of MWEs [4]. Another great distinction is between the tasks of *idiom type* and *idiom token* classification; while *idiom type* classification is the task of identifying expression with possible idiomatic interpretations, *idiom token* classification focuses on distinguishing between idiomatic and literal usages of potentially idiomatic phrases [5].

This research proposal will focus on the task of detecting idiomatic usage of token-level (*idiom token* classification) English VNCs. What distinguishes idiomatic and literal VNCs is the fact that an idiom has a different meaning than the resulting from the simple composition of the meaning of its component words [3]. To detect if a VNC presents idiomatic usage previous research has made use of supervised and unsupervised methods for learning underlying patters in idiomatic VNC formation, making use of the sentence context, the lexical and syntactic fixedness of the phrase in the corpora, and feature extraction with Word2Vec and Sent2Vec methods.

## Background

Past research focuses on VNC analysis since they have been able to extract lexical and semantic consistencies across different idiomatic phrases in the English language. First is the observation that most idiomatic VNCs exhibit **lexico-syntactic fixedness** [1] [3]; i.e. the phrase *see stars* often presents idiomatic meaning when the verb has active voice, the determiner is null, and the noun is in plural form, as in *see stars* or *seeing stars*; while usages with a determiner (*see the stars*), singular noun form (*see a star)*, or passive voice (*stars where seen*) often have literal interpretation [1].

**Lexical-fixedness** of idiomatic phrases means that the substitution of a near synonym for a constituent does not preserve the idiomatic meaning of the expression [3] (i.e. *see stars* and *observe stars*). Even if some idioms allow lexical variations which generate closely related meanings, these are usually highly unpredictable substitutions that can’t be considered as a rule [3].

**Syntactic-fixedness** means that many idiomatic VNCs cannot undergo syntactic variations while retaining their idiomatic interpretation (i.e. the punch let him *seeing stars* / *seeing the star*); however, it is relevant to note that idiomatic VNCs differ with respect to their degree of tolerance to semantic operations (**syntactic flexibility**) [3].

Following on the concept of lexico-syntactic fixedness, a corpus-based study by [6] demonstrates that idiomatic phrases are not as fixed as literature assumed in the past, since “the corpus data in this chapter show that-in contrast to nonidiomatic combinations of words-idioms have strongly preferred canonical form, but at the same time the occurrence of idiom variation is too common to be ignored”. This sounds redundant, as it says that idiomatic VNC identification must be on the lookout for any form of VNC variation since they can all be idiomatic. However, it also stablishes the **canonical form**, which is the base form of the idiomatic VNC (i.e. *see stars*), as a startup point for their detection. Subsequent research works on the assumption that idiomatic VNCs are more likely to appear on canonical form that non-idiomatic phrases [3].

Phrasal idioms have also been found to involve a certain degree of semantic idiosyncrasy, which means that the idiom is hard to determine without special context or previous exposure even if the meaning of the component words is clear [3] [5]. Also, although it is traditionally believed that idioms are completely non-compositional, linguists and psycholinguists claim that they show some degree of semantic compositionality [3]. This suggests that many idioms have internal semantic structure, without ignoring the fact that they are non-compositional in a traditional sense, which opens the field for the introduction of terms such as **semantic decomposability** and/or **semantic analyzability** [3]. To say that an idiomatic VNC is semantically analysable means that the constituents contribute by their independent meanings to the idiomatic interpretation; so, the more semantically analyzable an idiom is, the easier it is to interpret the idiomatic meaning from its constituents [3].

## Unsupervised Methods

Unsupervised models’ approach to detect idiomatic phrases used the assumptions mentioned in Section 2.1.

An approach by [3] calculates the lexical and syntactic fixedness in numerical values to generate a degree of fixedness, which is useful since idiomatic use of VNCs is believed to be both lexically and semantically more fixed than literal verb+noun combinations. To measure lexical fixedness of a pair, [3] calculates its association strength for the target pair and its variants using Pointwise Mutual Information (Equation 1).

Equation 1 - PMI for Verb+Noun Combinations in [3]

Where , in which , being a parameter for the number of closest verbs to target and the number of closest nouns to target [3]; is the total number of verb-object pairs in the corpus;, , and are the frequency counts of the target verb+noun pair, the target verb with any other noun and the target noun with any other verb respectively.

Equation 2 - Degree of Lexical Fixedness of Verb-Noun Combination in [3]

Equation 2 calculates a degree of lexical-fixedness for a verb-noun combination under the assumption that the target pair is lexically fixed to the extent that its PMI deviates from the average PMI of its variants [3]. The higher the degree, the more lexically fixed the pair is, thus . In Equation 2, and are the mean and standard deviation of the following sample: .

The author then proceeds to explain the process of calculating the Syntactic Fixedness, under the assumption that idiomatic VNCs appear in more restricted syntactic forms [3]. To quantify this value, they first identify relevant syntactic patters to distinguish idiomatic from literal usage, to then translate the frequency distribution of the target pair in the identified patterns to measure syntactic fixedness.

The identified syntactic patterns were:

* Passivization: Idiomatic VNCs often do not undergo passivization due to the non-referential status of the noun constituent in most idiomatic verb-noun pairs.
* Determiner type: There’s a strong correlation between the flexibility of the determiner preceding the noun in a VNC and the overall flexibility of the phrase. Idiomatic VNCs are expected to appear with one type of determiner.
* Pluralization: Even if the verb constituent of idiomatic VNCs is morphologically flexible, the non-referential noun constituent of the pair is expected to mainly appear in just one of the singular or plural forms.

The step of *devising a statistical measure* that quantifies the degree of syntactic fixedness using the proposed set of patterns proposes a measure that compares the syntactic behaviour of the target pair with that of a “typical” verb-noun pair. The syntactic behaviour of a typical pair is defined as the prior probability distribution over the selected patterns (Equation 3), where V is the set of all instances of transitive verbs in the corpus, and N is the set of all instances of nouns as direct objects of the verb.

Equation 3 - Syntactic Behaviour of Typical Verb-Noun Pair in [3]

For the target pairs, the syntactic behaviour is defined as the posterior probability distribution over the patterns given the pair, as shown in Equation 4.

Equation 4 - Syntactic Behaviour of Target Verb-Noun Pair in [3]

Using these two equations, the degree of syntactic fixedness for a target verb-noun pair is estimated the divergence of its syntactic behaviour from the typical syntactic behaviour, which is formulated in Equation 5 using Kullback Leibler (KL-) divergence. Thus .

Equation 5 - Degree of Syntactic Fixedness for a Target Verb-Noun pair in [3]

[3] hypothesizes that idiomatic VNCs are both lexically and syntactically more fixed than literal verb-noun combinations, thus they propose Equation 6 to measure overall fixedness of a given pair, rescaling the syntactic and lexical fixedness degrees under the range [0,1], so the overall fixedness falls in the range .

Equation 6 - Overall Fixedness for Target Verb-Noun pair in [3]

To measure the performance, the median score of the fixedness was determined as the threshold for separating idiomatic from literal VNCs, being pairs with an overall fixedness degree over the threshold classified as idiomatic. The results of accuracy, relative error rate reduction (ERR), the precision-recall curves, and the interpolated three-point average precision (IAP) are shown in Figure 1.

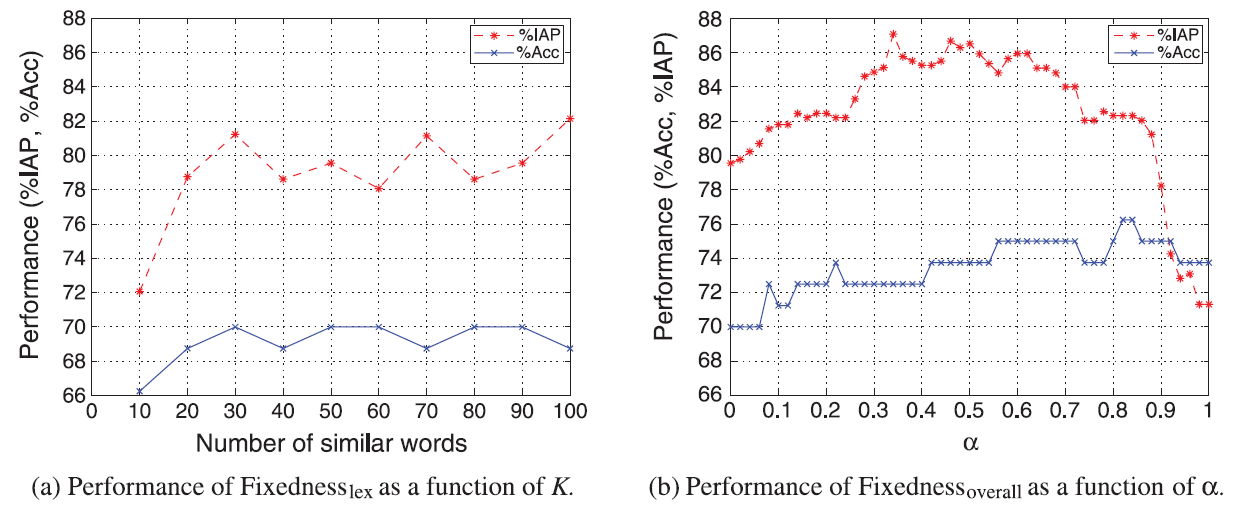


Figure 1 - %IAP and %Acc of Fixedness\_lex and Fixedness\_overall over development data [3]

## Supervised Methods

Supervised model training for idiomatic phrase detection focuses on identifying if a given excerpt of a sentence is of idiomatic or literal meaning. This approach usually tackles token-level identification of VNCs as a supervised binary classification problem, classifying the use of a VNC as idiomatic of literal [1] [5].

Recent research makes use of classifier models such as k-Nearest Neighbours (k-NNs) SVM with linear and polynomial kernels, since these algorithms have been proven to work for binary classification problems [1] [5]. However, the pre-processing of known VNCs and feature creation is the task in which most research focuses on. In an attempt to exploit the knowledge on lexical and syntactic patterns presented in Section 2.1, researchers have made use of unsupervised feature encoders to train the classifiers [1] [2] [5].

One approach tries to train a Linear SVM Classifier with data obtained by using **Word Embeddings** from Word2Vec [7]. This implementation by [2] proposes the creation of two vectors that contain both the representation of the VNC and its context, called and respectively. is created by averaging the word embedding vectors of the lemmatized component words of the VNC, while is the averaging of two other vectors: and , that represent the context of the verb and noun components respectively [2]. Once and are obtained they are subtracted into a feature vector, which is then appended a Boolean feature that determines if the VNC occurs in its Canonical Form (CForm) or not, as described in [3]. The resultant feature vector is then used to train an SVM with linear kernel. The results of this approach are shown on Table 1, comparing them to the models presented in [3].

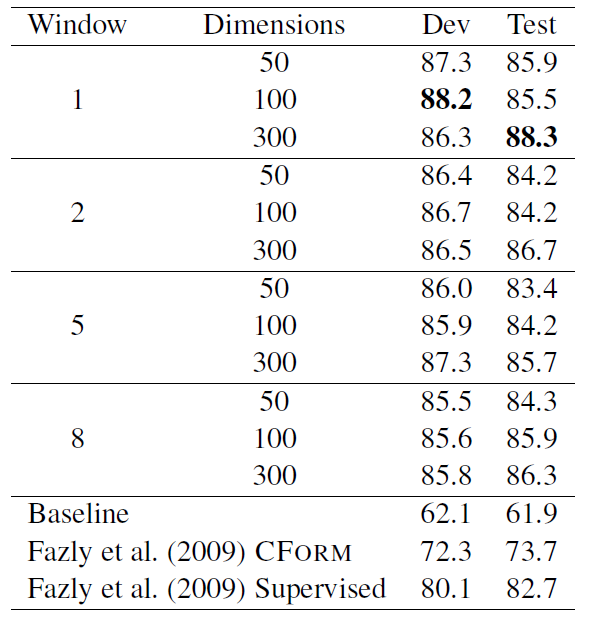


Table 1 - Accuracy Score for supervised word2vec approach by [2]

Other such approach is that of **Skip-Thoughts Vectors** (Sent2Vec) [8], used originally by [5] and then used as a base of comparison by [1]. This model uses the continuity of text from books to train an encoder-decoder model that aims to reconstruct the surrounding sentences of an encoded passage, so sentences with similar semantic and syntactic properties are mapped to similar vector representations [8]. This results in an encoder that can product highly generic sentence representations [8]. Sent2Vec was first used for idiom detection by [5] on the assumption that in a real-world application, target phrases won’t have access to a surrounding context; which motivated the exploration of distributed compositional semantic models to produce reliable estimates of idiom token classification [5]. Utility found in the Sent2Vec model is that it is possible to infer properties of the surrounding context only from the input sentence [5] [8], which allows the classifier to learn lexical and syntactic patterns without complex methods. [5] uses the resulting encodings to train three SVM classifiers with the VNC-Tokens Dataset [9]: Linear Kernel with C=1.0, Polynomial Kernel of degree = 2 and C = 1000, and Linear Kernel trained using Stochastic Gradient Descent with a learning rate of 0.0001. Results on the classifiers (Table 2) show an improvement on the baseline set by the authors, which used entire context extracted from several paragraphs.

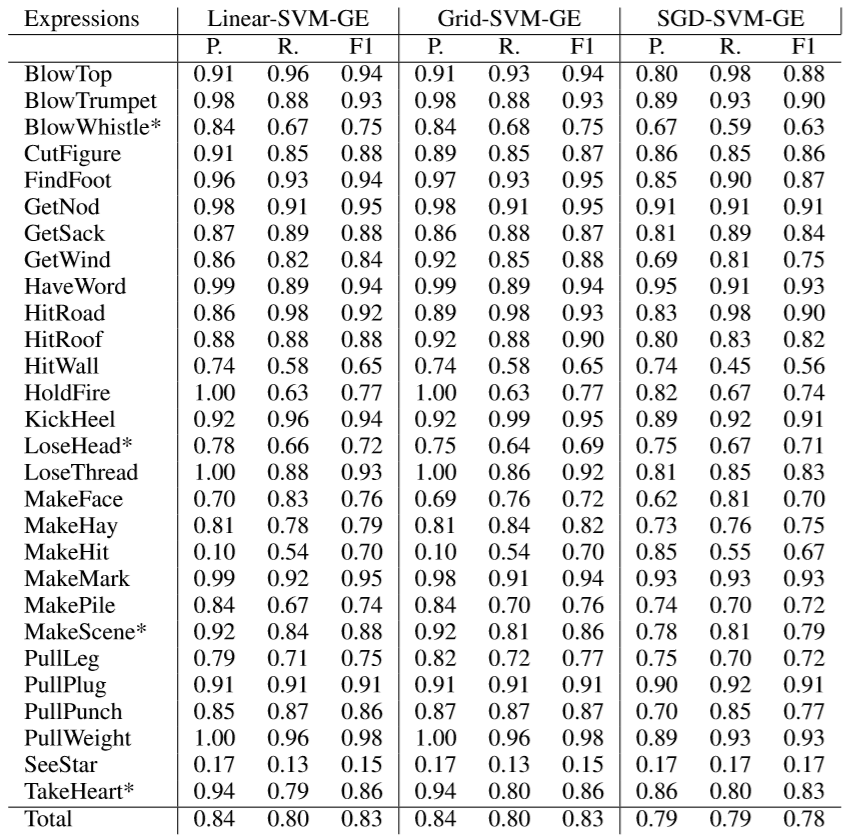


Table 2 - Precision (P.), Recall (R.), and F1-Score (F1) results on Generic Classifiers by [5]

The last studied research for the supervised learning portion of this project is that developed by [1], which also used a Linear SVM kernel but experiments with three different feature encodings for the VNCs and their context. First, they use Word2Vec’s **Skip-Gram** model[7], similarly to the approach taken by [2]; however, instead of encoding the VNC and context in different vectors and then subtracting them, [1]’s implementation averages the normalized word embeddings for each word in the sentences containing a target VNC. Secondly, they use the **Siamese CBOW** model [10] since it “learns word embeddings that are better able to represent a sentence through averaging that conventional word embeddings such as skip-gram or CBOW” [1]; the word embeddings produced for this model for a target sentence are averaged as in the Skip-Gram implementation. Lastly, they replicate the skip-thoughts model approach taken by [5] to use as a strong baseline for comparison. As an extra feature, they append the CForm [3] Boolean feature as described previously with the approach taken by [2]. The accuracy score results for the different word embeddings methods used with and without the added CForm are shown on Table 3.

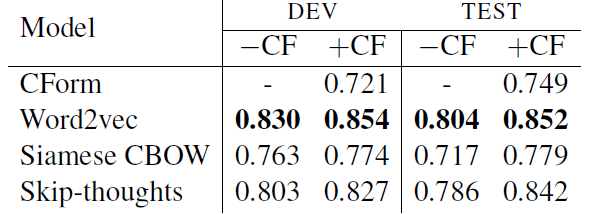


Table 3 - Accuracy Score for supervised word2vec, siamese cbow and skip-thoughts approaches by [1]

# Goals

The goal of this project is to develop a supervised and unsupervised model for idiom detection in text. The supervised model will have the task of detecting idiomatic use of a VNC in a target sentence, while the unsupervised model will detect idiomatic usage of an unknown VNC in target text and potentially annotate the VNC.

## Objectives

The objective of developing the models start with first creating the supervised model, and then expanding the knowledge to train the unsupervised model.

To create the supervised model, we’ll replicate the experiments from [1]:

* Implement **Word2Vec**’s Skip-Gram model for sentence encoding
* Implement **Siamese CBOW** model for sentence encoding
* Implement **Skip-thoughts** model for sentence encoding
* Add **CFORM** feature to encoded feature vectors
* Implement and train **SVM** with available sentence encodings
* Evaluate model performance with Accuracy and F1-Score Metrics

After recreating [1] experiments, a supervised and unsupervised learning models will be implemented using combined knowledge from [1] and [3]:

* Implement **Lexical Fixedness** and **Syntactic Fixedness** formulas on the available corpus
* Implement **Overall Fixedness** formula
* Implement method that extracts the verb-noun pair with highest Overall Fixedness in target sentence. Use this method for unsupervised model.
* Append **Overall Fixedness** metric to the created sentence encodings with the **Word2Vec**, **Siamese CBOW**, and **Skip-thoughts** models.
* Train **Linear SVM** supervised models and **k-Means** unsupervised model with the encodings.
* Evaluate new models’ performance over the previous implementations.

## Scope and Limitations

The **scope** of the project is the development of a supervised and unsupervised classifier that can detect idiomatic usage of VNCs in text. It is limited to only make use of Verb Noun Combinations type of MWE, such as *see stars*, and won’t go into more complex forms of MWEs such as *Time flies when you’re having fun*, that still have idiomatic meaning but are not considered in the background research. Also, the VNCs considered during training and testing will only be those that appear in the available datasets, since the objective of this project is not that of providing manual annotations for idiomatic VNCs.

**Limitations** of the project are the equipment for model training. Hardware is limited by that provided by the University of Essex, so tests may have time and memory constraints regarding the number of tests that can be performed and the number of training examples to use in those states limited by available machine state.

# Project Description

This section explains how the project will be developed, as well as any third-party tools used for its completion.

## VNC Tokens Dataset

The dataset used to finish this project will be the VNC Tokens Dataset developed by Cook et al. 2008 [9]. This dataset was used by previous researches such as [1], [2], and [3] this it will be ideal to replicate their findings and use as a fair baseline for the performance of our classifiers.

The dataset consists of 53 VNC expression, used within roughly 3000 sentences in which they are tagged as expressing “Literal”, “Idiomatic”, and “Unknown” usage. VNC entries are tagged as follows:

ANNOTATION VERB\_NOUN FILENAME SENTENCE-NUM

For example:

I blow\_top A/A7/A7N 1279

Which means that the phrase “Blow Top” has idiomatic usage in sentence 1279 from the corpus A/A7/A7N. The corpus corresponds to an entry in the BNC XML dataset, which is required as well [11].

## Third-Party Tools

### Gensim

Is a free, open source, tool for unsupervised semantic modelling form plain text [12]. This allows for a simple, pretrained, implementation of Word2Vec by [7].

### Siamese CBOW

Public repository containing the code of Siamese CBOW to generate word embeddings based on the paper by the same author [10]. Code can be found here: <https://bitbucket.org/TomKenter/siamese-cbow>.

### Skip-thoughts

Sent2Vec encoder and training code used on the paper by [8]. The source code can be found on the following repository: <https://github.com/ryankiros/skip-thoughts>.

### Scikit-Learn

Simple and efficient tools for data mining and data analysis, build over NumPy, SciPy, and matplotlib [13]. These tools are open source. We will use the SVM implementation found in this library for Python.

## Project Development Methodology

// TODO

# Evaluation

// TODO: Write Evaluation

# Project Plan

// TODO: Add Work Breakdown Structure and Gantt Chart

# Acknowledgements

// Should I include acknowledgements?

# References

|  |  |
| --- | --- |
| [1] | M. King and P. Cook, “Leveraging distributed representations and lexico-syntactic fixedness for token-level prediction of the idiomaticity of English verb-noun combinations,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, Melbourne, 2018. |
| [2] | W. Gharbieh, V. C. Bhavsar and P. Cook, “A Word Embedding Approach to Identifying Verb-Noun Idiomatic Combinations,” in *Proceedings of the 12th Workshop on Multiword Expressions*, Berlin, 2016. |
| [3] | A. Fazly, P. Cook and S. Stevenson, “Unsupervised Type and Token Identification of Idiomatic Expressions,” *Computational Linguistics,* vol. 35, no. 1, pp. 61-103, 2009. |
| [4] | A. Savary, C. Ramish, S. R. Cordeiro, F. Sangati, V. Vincze, B. QasemiZadeh, M. Candito, F. Cap, V. Giouli, I. Stoyanova and A. Doucet, “The PARSEME Shared Task on Automatic Identification of Verbal Multiword Expressions,” in *Proceedings of the 13th Workshop on Multiword Expressions*, Valencia, 2017. |
| [5] | G. D. Salton, R. J. Ross and J. D. Kelleher, “Idiom Token Classification using Sentential Distributed Semantics,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, 2016. |
| [6] | S. Z. Riehemann, “A Constructional Approach to Idioms and Word Formation,” Stanford University, Stanford, 2001. |
| [7] | T. Mikolov, K. Chen, G. Corrado and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” in *Proceedings of Workshop at the International Conference on Learning Representations*, Scottsdale, 2013. |
| [8] | R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba and S. Fidler, “Skip-Thought Vectors,” *Advances in Neural Information Processing Systems,* vol. 28, pp. 3276-3284, 2015. |
| [9] | P. Cook, A. Fazly and S. Stevenson, *The VNC-Tokens Dataset,* Toronto: University of Toronto, 2008. |
| [10] | T. Kenter, A. Borisov and M. de Rijke, “Siamese CBOW: Optimizing Word Embeddings for Sentence Representations,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, 2016. |
| [11] | B. Consortium, *The British National Corpus, version 3 (BNC XML Edition),* Oxford: University of Oxford, 2007. |
| [12] | R. Rehuek and P. Sojka, “Software Framework for Topic Modelling with Large Corpora,” in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, Valletta, 2010. |
| [13] | F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and É. Duchesnay, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research,* no. 12, pp. 2825-2830, 2011. |

# Appendix

// TODO