Handling Idiomatic Expressions in English

Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science in ***Computer Science and Engineering*** *by Research*

by

Prateek Saxena
200902016
prateek.saxena@research.iiit.ac.in

International Institute of Information Technology
Hyderabad - 500 032, INDIA
MARCH 2023

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "Handing Idiomatic Expressions in English" by Prateek Saxena, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date

Adviser: Prof. Soma Paul

To Vaishali, Kartik and My Parents

Acknowledgments

I want to start by expressing my gratitude to Soma Paul, ma'am, my guide throughout the entire process. Despite my behaviour giving her multiple reasons not to, she continued to trust me when it mattered most. Without her assistance, this thesis could not be completed. I am appreciative of my peers Shastri and Ayushi for supporting me in my research and assisting me in overcoming difficult times. I can't thank my wife Vaishali enough for providing the bulk of the thesis' brainpower and frequently acting as a major brawn. She contributed just as much, if not more, to this work as I did. Finally, without the selfless support, encouragement, and belief of my parents and brother, this would not have been possible.

Abstract

Idiomatic expressions have always been a bottleneck for language comprehension and natural language understanding, specifically for tasks like Machine Translation(MT) and Natural Language Understanding(NLU). MT systems predominantly produce literal translations of idiomatic expressions as they do not exhibit generic and linguistically deterministic patterns which can be utilized for the comprehension of the non-compositional meaning of the expressions. These expressions occur in parallel corpora used for training, but due to the comparatively high occurrences of the constituent words of idiomatic expressions in a literal context, the idiomatic meaning gets overpowered by the compositional meaning of the expression. The absence of data with a large coverage and quantity of idiomatic expressions exacerbates the issue to handling them further. Our work aims to provide a method of handling idiomatic expressions which not only suggests a pipeline for the task but also enables a process of data creation from subsequent steps which can be used in further downstream tasks.

State of the art metaphor detection systems are able to detect non-compositional usage at word level but miss out on idiosyncratic phrasal idiomatic expressions. This creates a dire need for a dataset with a wider coverage and higher occurrence of commonly occurring idiomatic expressions, the spans of which can be used for Metaphor Detection. With this in mind, we present our English Possible Idiomatic Expressions(EPIE) corpus containing 25206 sentences labelled with lexical instances of 717 idiomatic expressions. These spans also cover literal usages for the given set of idiomatic expressions. We also present the utility of our dataset by using it to train a sequence labelling module and testing on three independent datasets with high accuracy, precision and recall scores.

Natural Language Understanding has made recent advancements where context-aware token representation and word disambiguation have become possible to a large extent. In this scenario, comprehension of phrasal semantics particularly in the context of multi word expressions (MWE) and idioms, is the subsequent task to be addressed. Word level metaphor detection is unable to handle phrases or MWE(s) which occur in both literal and idiomatic context. State of the art transformer architectures can be useful in this context, but the absence of a large comprehensive dataset is a bottleneck. In this paper, we present a labelled EPIE dataset containing 3136 occurrences for 358 formal idioms. To prove the efficacy of our dataset, we also train a sequence classification model effectively and perform cross-dataset evaluation on three independent datasets. Our method achieves good results on all datasets with F1 score of 96% on our test data, and 82%, 74% and 76% F1 score on SemEval All Words, SemEval Lex Sample, and PIE Corpus datasets respectively.

We have also utilized our disambiguation model in the shared task of disambiguation of german verbal idioms and our results are comparative to the state of the art results achieved for the task.

Phrasal replacement of idiomatic expressions is a non trivial task as it aims at locally changing the idiomatic expression to another phrase which is semantically similar and more literal in nature, while keeping the rest of the sentence intact. We present a sentence aligned dataset of 3136 sentences of the labelled EPIE dataset with each sentence having a phrasally replaced counterpart containing the replaced phrase. We also present a sequence to sequence model for learning these replacements with good results.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Statement

Idiomatic Expressions are well known phrases, fixed or flexible, the semantics of which is non compositional in nature.

1. We went around town *tripping the light fantastic*.

Example 1 has the idiomatic phrase "trip the light fantastic" which means "to dance nimbly" which is not directly related to the individual words of the phrase. Idiomatic Expressions are a type of multi word expressions(MWEs). Idiomatic expressions have garnered a lot of interest in the recent times. This is primarily because handing idiomatic expressions is essential for a lot of natural language understanding and processing tasks. A system that can automatically paraphrase idioms in context has applications in many NLP tasks.

1.1.1 Non trivial nature of automatic handling of Idiomatic Expressions

Idioms can be challenging for speakers of second languages. In a pilot study by [26], seven non-native speakers were polled on 100 Tweets that contained idioms. According to the results, on average, the participants had problems comprehending 70% of the Tweets because of the use of idioms.

Futhermore, automatic handling of idioms is non trivial in nature as it occurs very frequently in text.[39] states that the main problems consist in recognizing an idiom and in distinguishing idiomatic from literal usage. For example, both sentences below contain the same phrase "black sheep"

2. All the *black sheep* have gone from the barn.

3. He is the *black sheep* of his family.

However, the usage is literal in example 2 and idiomatic in example 3. Since many idioms are structurally flexible and can be unevenly distributed throughout a clause, recognition can be challenging. Example 4 below uses the idiomatic expression "keep the wolf from the door".

4. All my salary goes into *keeping the proverbial wolf far far away from the door*.

But because of the modifier "proverbial" and "far far way" it becomes difficult to recognize the complete phrase. Another issue with handling idiomatic expressions post detection is the difficulty to distinguish between idiomatic and non-idiomatic usage. In some cases, this can be accomplished by using unique phrases that are predominantly used in idioms. The phrases "bite the dust" and "kill time" are some such phrases which predominantly have idiomatic usages. However, this distinction generally falls outside the scope of the present translation technologies because it is a matter of semantics and pragmatics.

[33] also mentions that in natural language processing (NLP), there has been a conflict between symbolic and statistical approaches for some time. While deep processing has already reached an industrial level and is the foundation for continuous product development in a number of application domains, it is commonly acknowledged that deep analysis must address a number of significant issues if linguistically precise NLP is to become a reality. [7] also mention this specific to light-verb constructions that due to their peculiar behaviour at various levels of linguistic description, multiword expressions (MWEs) are widely recognised in the NLP community to be a difficulty for many NLP applications. For example, the word "take" does not contribute fully to the semantics of the phrase "take a bath" but it is still different from the contribution of "make" in "make a bath". Capturing this peculiarity requires capturing this distinction which is difficult for standard NLP systems. Standard MT systems struggle with this because they are unable to discriminate between the literal and idiomatic uses of the verb. [34] states that although idioms that are highly fixed can be represented as words-with-spaces by an NLP system [33] if an idiomatic meaning persists across morphosyntactic variations of an expression, the words-with-spaces approach does not work for these formal example of idioms. Another feature of idioms that make them difficult for NLP system to process is that idiomatic expressions have both idiomatic and literal (non-idiomatic) usages. Machine translation systems are among the most significant NLP applications that are adversely impacted by idioms. [34] demonstrates that a normal MT system only gets around half the BLEU score on sentences including idioms as it does on sentences without idioms. This drop in the score occurs not only due to the comparatively low frequency of the idiomatic phrase with respect to the frequency of the constituent words, but also due to the lack of automatically determinable clear patterns in the wide and varied instances of idioms in data [15]. In this aspect, a regular monolingual training dataset is sparse with respect to idiomatic expressions. The absence of a dataset rich in idiomatic expressions hampers the possibility of modelling the problem into a machine learning task. A dataset rich in idiomatic expressions is necessary to handle these expressions in a specific way in order to improve machine comprehension in NLP due to their intrinsic grammatical complexity and high frequency in data.

1.1.2 Steps for handling idiomatic expressions

Any attempt on handling these idiomatic expressions has to follow certain predefined steps as discussed in [26]. A successful system's whole pipeline must find solutions to several issues. It must first establish that an expression is being utilised in a sentence as an idiom ([14], [22], [37]). Additionally, the system must be able to distinguish between multiple plausible interpretations of the phrase in order to select the proper one. Second, it must produce a suitable idiomatic substitute using literal English. Third, it must guarantee that the substituted term flows naturally into the original sentence.

It is with this motivation that we identify our task of creating a dataset for idiomatic expressions. Our dataset contains multiple layers of data based on the steps described by [26].

1.2 Idiomatic Expression Types

Idiomatic expressions can be classified in a lot of ways. Based on the ease of finding meaning, idiomatic expressions can be *encoding* or *decoding*.

- *Encoding idioms* are idioms whose meaning is either hard to figure out using the constituent words or the conventionality of their usage cannot be assumed by a speaker, for example *answer the door*.
- *Decoding idioms* are idioms whose meaning cannot be figured out using the constituent words, for example *kick the bucket*.

Based on their constituent words' parts of speech, idioms can be of various types[33], some of which are,

- *Verb-Particle Constructions* are made of a verb plus one or more particles. In compositional contexts, the particle(s) function as a construction and change the head verb's spatial, aspectual, etc. qualities. For example, in the phrase "eat up," the particle "up" changes the head verb "eat" from an activity to an accomplishment. In other words, the particle(s) typically take on semantics peculiar to verb-particle formations.
- *Light Verbs* are one that, on their own, provide minimal semantic information and, along with another expression—typically a noun—forms the predicate. Examples include *take bath* and *give a presentation*.
- *Compound Nominals* are created by joining two or more words to create a noun, for example, *company men*. These distinct words don't have to be nouns themselves, for example, *son in law*.

For our task, we have chosen to classify idiomatic expressions based on their lexical flexibility([33]).

- *Static idioms* are idiomatic expressions which are fixed and therefore, do not undergo any lexical modifications during their usage. Examples of *static idioms* include short phrases like *more or less* and full sentences like *rome wasn't built in a day*.

- *Formal idioms* are idiomatic expressions which are syntactically flexible i.e. they can undergo lexical changes during usage. These lexical changes can be a change in order, inflectional changes, tense modifications and adverbial and adjectival qualitative modifiers.*Formal idioms* are idiosyncratic in the lexical variations they allow in their occurrances. There is no set pattern or hint at the word or phrase level that can account for all the changes that can be expected from a particular idiomatic expression. Examples of *formal idioms* are *keep an eye on*, *kick the bucket*, *stitch up* and *man of means*.

In addition to the syntactic flexibility, a lot of formal idiomatic expressions also occur in literal contexts as well. For example, *drag one's feet* is a very common formal idiomatic expression, the meaning of which corresponds to *being slow to act*. However, in multiple instances, *drag one's feet* is used in a literal context, like in the sentence 'Children tend to *drag their feet* while walking'. These occurrences of formal idiomatic expressions in literal context makes any kind of automatic handling based on syntactic behaviour impossible. Therefore, it becomes important to have models trained on disambiguating such instances necessary for any meaningful resolution to the problem. This problem can be modelled as a classification task, with the idiomatic expression sequence getting assigned a label, thus, disambiguating the idiomaticity of the phrase in the given context. However, classification models are like any other machine learning NLP models i.e. they are dependent on good training data which in this context would be data which contains literal and idiomatic usages for a wide variety of idiomatic expressions. [26] also identifies this as step 2 of the predefined steps of handling idiomatic expressions, the first being detection of such lexical occurrences.

1.3 Proposal and Significance of our work

In this thesis, we attempt the task of detection, identification, disambiguation and paraphrasing of English idiomatic expressions into their semantically literal counterparts. Based on the previous works by [26] and [32], we decide on paraphrasing as the best way to handle idiomatic expressions for the following reasons.

- Idiomatic expressions do not have a 1 to 1 cross lingual mapping. Any idiomatic expression of a source language need have a corresponding idiomatic expression in the target language.

- Because of the flexible structure of idiomatic expressions, any conversion to a dictionary or other semantic forms comes with its own set of challenges.

1.4 Key Contributions

The overall research contributions are the following:

- A corpus of 25206 sentences containing 717 idioms with the lexical occurrences labelled. These idiomatic occurrences have also been segregated based on whether the idioms are Static(lexically fixed) or Formal(lexically flexible). We also release an analysis of the distribution of these idioms accross the sentences.
- A corpus of 3136 sentences containing lexical occurrences of 358 formal idioms with each occurrence being disambiguated for idiomatic or literal occurrence. We also release a transformer based model[38] suited for the task and its comparison with other baselines.
- A sentence aligned parallel corpora of 3136 sentences with literal paraphrases of idioms in the sentence. We also discuss the results of a sequence to sequence paraphrasing task with this dataset using the T5 model architecture([30].

1.5 Thesis Organization

The thesis contains following chapters:

- Chapter 2 contains all the background work and related work that this thesis is based on.
- Chapter 3 contains the task of creating a dataset and model for aiding the detection of possible idiomatic expressions in the dataset i.e. phrases that are lexically similar to known idiomatic expressions
- Chapter 4 contains the task of disambiguating the detected phrases in the previous step as idiomatic or literal in order to aid in the phrasal substitution of the idiomatic occurrances of phrases.
- Chapter 5 contains the results and tests of using the phrasal disambiguation model on languages other than English with KONVENS 2021 Shared Task of disambiguating German Verbal Idioms [13].
- Chapter 6 contains the results and evaluations for the task of phrasal substitution of idiomatic phrases with a semantically similar literal phrase.
- Chapter 7 contains the conclusions of the thesis and future work.

Chapter 2

Related Work

2.1 Related work on types of Idiomatic Expressions

[33] created a distinction in idioms i.e. Formal and Static. Static idioms are the kind of idioms which do not exhibit internal or morphosyntactic variation. For example, *As soon as possible*, *no comment*, etc. Formal idioms, on the other hand, undergo inflectional changes, pronominal and determiner modifications, and internal qualitative modifiers(adjectival and adverbial). For example, *keep eye on*, *race against time* etc. The distinction by [33] defines formal idioms as idiomatic expressions which undergo linguistic changes and can have qualitative modifiers in their constructions. In addition to this, the contexts in which these formal idiomatic expressions occur also have a wider coverage in both literal and idiomatic usages. As formal idiomatic expressions are harder to handle automatically, our work, therefore, focuses on labelling a wide range of examples for formal idiomatic expressions.

2.2 Related work on Idiomatic Expression Detection

StringNet[40] identified that mapping base forms of phrases to surface forms is necessary in order to extract their surface realization. StringNet used hybrid ngrams and cross indexing to create a resource to extract idiomatic sentences from the British National Corpus[25]. The British National Corpus (BNC) is a collection of 100 million words that includes samples of spoken and written language from a variety of sources. It was created to reflect a broad cross-section of British English from the latter half of the 20th century, both spoken and written. We use StringNet for the first level extraction of sentences for our work from BNC.

[1] has created the IMIL dataset which maps 2000 of the highly occurring English idioms to their counterparts in different Indian languages. We use their idiom list as a starting point for our sentence extraction.There have been some attempts to extract idiomatic expressions. The VNC-Tokens Dataset[9], IDIX Corpus[37], PIE Corpus[18] and SemEval-2013 Task 5 Dataset[22] all contain around 3000 to 4500 potential idiomatic expressions instances of 53 to 65 candidate idioms. These datasets, though thorough for their respective candidate idioms, are small in size and limited in coverage. By developing

the English Possible Idiomatic Expressions(EPIE) dataset, we attempt to provide a wider coverage for idiomaticity detection over a larger sample set. We cover both lexical occurances of idioms in samples, and also determine whether the occurance is idiomatic in nature.

2.3 Related work on Idiomatic Expression Disambiguation

[16] provides a model for metaphor word detection in context. They use an idiomaticity label which defines the usage of a particular word as metaphorical or literal in the given context. [16] presents two tasks, namely sequence labelling and word classification. For sequence labelling task, each word of the input sequence is labelled with an idiomaticity label whereas for the classification task, one word from the whole input sequence is labelled with the idiomaticity label. For our task, we have chosen to extend the classification task to assign idiomaticity label to an idiomatic expression. We have used the insights provided in the paper as well as a comparative baseline to compare our model. However, the paper detects metaphor words in a given context whereas we classify whole idiomatic phrases usage as idiomatic or literal within the sentence using a single label. [32] presents a recent work on "MWE-aware" metaphor identification systems. They use BERT, Graph Convolution Network(GCN) and multi-headed self attention for metaphor identification. This work provides a good starting point for metaphor identification systems focusing on verbal MWEs. Our work aims to provide a simpler system for idiomaticity detection which works for a wider range of idiomatic expressions than verbal MWEs including other idiomatic expressions like compound nominals and verb particle constructions.

There are many datasets for idiomatic expressions containing both idiomatic and literal usages labelled into the dataset. The PIE Corpus[18] containing around 1100 labelled instances of 200 candidate idioms and the SemEval-2013 Task 5 Dataset[22] contains around 3000 labelled instances of 53 to 65 candidate idioms. These datasets are thorough but do not provide a wide coverage of idiomatic expressions. The EPIE Dataset [35] containing 3136 labelled instances for 358 idiomatic expressions, provides a large dataset with a wider coverage, which is why we labelled the EPIE Dataset [35] for our task.

BERT [11] has been used extensively to capture context aware linguistic information into the neural network model. BERT uses multiple embeddings to capture word level, position level and segment level information, and exploits multi headed attention layers to capture semantic relationship between tokens. Also, BERT enables building and augmenting neural network layers on top of existing models with ease. For our task, we have used BERT tokenizer and augmented the data for our purpose to then initialize and fine tune a BERT for Sequence Classification model for labelling idiomaticity.

2.4 Related work on Idiomatic Expression Paraphrasing

There are two ways for simplify the semantics of idiomatic expression in text.

2.4.1 Text to Text

In text to text conversion we transfer the semantics of the idiomatic phrase to a literal phrase without compromising the semantics. [26] has attempted phrasal substitution of idiomatic expressions to a literal phrase with similar semantics. The study makes the assumption that a current, comprehensive, high-quality dictionary is available. If this presumption is accurate, it does address the issue of providing the correct interpretation for further downstream procedures. Our empirical findings, however, point to a lack of an appropriate resource. Although the source cited by [26] is a good one, it only provides the definitions of 359 of the idioms from the EPIE corpus, or 70% of them. In order to replace the meaning, we collected meanings from many sources and used numerous annotators with interannotator agreement. The substitution method used by [26] also uses an ML-based approach. For generalizing patterns, [26] casts substitute generation as an binary classification problem. They segment the definition to syntactic chunks using off-the-shelf shallow parsers and then apply a trained binary SVM classifier on each chunk to predict whether it should be kept or discarded. This process requires manual feature identification for the SVM task. However, after the prediction of the correct substitute to replace the phrase, incorporating grammatical adjustments, pronominal resolution and smoothening over the replacement boundaries is still required to be done. For the sequence-to-sequence paraphrasing job, we used a modified transformer-based architecture. A deep neural network with the same number of parameters as an SVM always has a higher complexity than the latter, which is the main justification for choosing a transformer-based architecture over the former.

[21] attempts to solve this problem for 24 non compositional idioms by creating rules for their detection and replacement. They use text input and high quality linguistic resources like tokenizer, POS tagger, lemmatizer to write a python code which does string matching to detect idioms are replace them. While the detection works with 88% precision score and 100% recall score for the 24 idioms, the phrasal substitution only improves translation for 18 of the 24 idioms. While string matching works for lexical detection, lack of a disambiguation module enables conversion of idiomatic expressions even if the occurrence is literal. In addition to this, addition of new idioms encompasses complex linguistic analysis overheads for annotators and resources alike.

2.4.2 Text to Semantic Structure

In text to semantic structure conversion, the task is to reduce the idiom to some semantic structure and then transfer the semantics. [24] attempts to handle idiomatic expressions this way for French. Their work bases itself on the assumption that in order to handle idiomatic expressions, the system requires access to the internal syntactic structure of the idioms. They use deep syntactic representation (DSyntR) which is similar to the universal dependency structure, and convert it to a full dependency tree called a surface syntactic representation (SSyntR). This is done as idioms often occur as a single node in DSyntR. As for each idiom the full dependency tree conversion will require separate rules, [24] perform template lexicalization to create generalized full dependency trees from idioms. They create

linguistic structures based on POS and number of nodes for each idiom, and perform similar conversion for idioms sharing the same POS structure and number of nodes. They are able to classify 2919 idioms into 514 patterns, with most idioms being nominal compounds, verbal idioms and prepositional idioms. However, the matched idioms are mapped to a semantic structure, but they do not perform any semantic transfer for the idioms. This work offers high precision and full control over seen idioms but the process of incorporating any new idiom is manual and requires complex linguistic resources and annotators trained on the GenDR system and having knowledge of DSyntR and SSyntR structures.

Chapter 3

EPIE Dataset: A Corpus For Possible Idiomatic Expressions

3.1 Introduction

Idiomatic Expressions are phrasal expressions whose meaning is non compositional i.e. the semantics of which is difficult to figure out based on the meaning of the constituent words. For example, the idiomatic expression *nerves of steel* cannot be understood based on the constituent words *nerves* or *steel*. The meaning of the above expressions is *the ability to stay calm* and has no relation to the syntax of the expression. The meaning of some idiomatic expressions may be figured out based on the context, however assuming the conventionality of usage of such idioms is still dependant on each user and their knowledge. Idiomatic Expressions occur very frequently in text. Light verb constructions like *fall asleep* and *take bath*, compound nouns like *car park* and *part of speech*, and fixed phrases like *by and large* are all examples of idiomatic expressions. Idiomatic expressions are also called Multiword Expressions(MWEs) and idioms.

Because of the high amount of presence of idiomatic expressions in the language, it becomes necessary to have systems which can understand and process the underlying semantics of these phrases. However, the idiosyncratic behaviour that these expressions exhibit makes it a tough problem to handle automatically. Natural language understanding of idiomatic expressions embedded in sentences has been a complex problem to solve for some time. Idiom handling has been a problematic area for a variety of NLP tasks. [33] and [7] have discussed the magnified complexity of this problem with respect to linguistic precision. Idiomatic expressions are hard to handle for machines and humans alike. A pilot study([26]) of seven non-native english speakers using 100 tweets showed that participants could not understand 70% of the tweets that contained idioms. In case of machines, [34] provides empirical evidence that state-of-the-art machine translation systems may achieve only half of the BLEU score on sentences that contain idiomatic expressions as compared to the ones that do not. This drop in the score occurs not only due to the comparatively low frequency of the idiomatic phrase with respect to the frequency of the constituent words, but also due to the lack of automatically determinable clear patterns in the wide and varied instances of idioms in data [15]. These results point towards an integral need for systems to understand and process idiomatic expressions, in order to aid any NLP tasks in the

downstream. A rule based approach for a solution is possible. The system created by the approach will work with precision, but it will lack coverage in terms of the number of idioms it can handle and the number of constructions it can allow for a particular idiomatic expression. A statistical approach for a solution is also possible but the bottleneck of any statistical approach is the availability of good data to develop valid statistical insights on the problem, and a good machine learning algorithm to understand the underlying insights that the data provides. A regular monolingual training dataset is sparse with respect to idiomatic expressions in comparison to its constituent words. The absence of a dataset rich in idiomatic expressions hampers the possibility of modelling the problem into a machine learning task. For a machine learning task, a dataset labelled with idiomatic expressions is necessary, so that the model can understand the difference between the occurrence of the idiom and its constituents. It is with this motivation, that we introduce our task of creating a high volume dataset of a large variety of instances of idiomatic expressions. Our algorithm of choice was to use BiLSTM-CRF as an encoder, as it has shown to capture linguistic information into its models, using word level, position level and segment level information and then utilize everything into a multi headed attention layer to understand the semantic relationships.

The first step of handling idioms according to [26] is to detect lexical occurrences of idiomatic expressions in a given text. The subsequent steps constitute identifying the underlying semantics and learning a simpler representation for any downstream task. In this chapter, we attempt the first step from the aforementioned steps i.e. detection of possible idiomatic expressions in a given text. These lexical variations can have a literal occurrence as our purpose is to capture the span of the phrase in order to identify a metaphorical usage as the next step. We present a dataset of 25206 sentences which contain lexical occurrences of 717 idiomatic expressions from the IMIL dataset [1]. We identify the detection of idiomatic expressions as a sequence labelling task and present a two pronged approach for detection of two different kinds of idioms: Static and Formal. Static idioms do not undergo lexical changes, therefore labelling them can be as simple as a string search in the text. Formal idioms, on the other hand, undergo various lexical modifications, therefore labelling them can be modelled as a supervised task. We test a model trained on our dataset and test on three datasets,"all words" and "lex sample" training datasets of SemEval-2013 Task 5b Dataset[22], and PIE Corpus[18]. All tests give results with high accuracy, precision and recall scores.

The major contributions of this work can be summarized as follows:

- We publically release a dataset of 25206 sentences labelled with lexical occurrences of 717 idioms. These labels are done by automatic systems with high accuracy. Of these, 21891 sentences contain occurrences of Static idioms which are 359 in number and 3135 sentences contain occurrences of Formal idioms which are 358 in number.[1]

- An analysis of the distribution(Mean and Standard Deviation) of idioms over the dataset.

[1]Dataset available at: https://github.com/prateeksaxena2809/EPIE_Corpus

3.2 Data

Our aim is to create a dataset only containing sentences with lexical occurrences of idioms for the IMIL dataset. This requires multiple data filtering steps. These steps are explained in the subsequent subsections.

3.2.1 StringNet Extraction

Variations in Idiomatic Expressions occurs in the following forms:

- Inflectional Modifications (tense, gender, number, etc):

  Bite the dust

  - The visiting team *bit the dust* in the football game yesterday.

- Determiner/Pronominal Replacement:

  Keep up the good work

  - *Keep up your good work* and the promotion will follow.

- Named Entities and Qualitative Modifiers inclusions(Adjectival and Adverbial)

  Keep an eye on

  - *Keep a keen eye on* the child while he plays.

  Behind his back

  - People say a lot *behind James' back*.

In order to extract all instances of an idiomatic expression, it is important to account for all the variation in the expression. We use StringNet for this task. StringNet is made up of a sizable collection of hybrid n-grams. Hybrid n-grams, in contrast to conventional n-grams, can be made up of any co-occurring arrangement of POS tags, lexemes, and particular word forms. These two billion connected hybrid ngrams are cross-indexed with lexeme information, parts of speech information and various word forms. This matches an idiomatic expression like *keep your eye on* to its inflectional modifications like *kept your eye on* and *keeps your eye on*. Due to StringNet's use of crossindexing with lexeme and parts of speech information, every occurrance matches a parent ngrams utilizing that information. Figure 3.1 shows an example of the StringNet interface. We also utilize StringNet's unique feature of vertical pruning and horizontal pruning.

- *Vertical pruning* refers to generalization of lexemes in a given search entry in order to search occurrence of parent ngrams and child ngrams of the entry in the corpus. For example, a parent ngram of the entry *Keep your eye on* is *keep [pron] eye on* as [pron] constitutes all pronouns.

No	Hybrid ngram	Examples	Parents	Children
1.	**keep** a [aj0] eye on the [nn1]			
2.	**keep** a [aj0] eye on			
3.	[cjc] **keep** a [aj0] eye on			
4.	keep a [aj0] eye on			
5.	**keep** a [aj0] eye on the			

Figure 3.1: Example of StringNet search of *keep a [adj] eye on*

No	Hybrid ngram	Examples	Parents	Children
1.	keeping a [aj0] eye on			
2.	**keep** a close eye on			
3.	**keep** a watchful eye on			
4.	kept a [aj0] eye on			
5.	keep a [aj0] eye on			

Figure 3.2: Children of the ngram *keep a [adj] eye on*

Vertical Pruning helps in extraction of pronominal and determiner variation. The children of an example ngram can be seen in Figure 3.2.

- *Horizontal pruning* refers to connecting an ngram with another ngram which differs by one unit or type of ngram. For example, the entry *keep [det] eye on* can be connected to *keep eye on* and *keep [det] keen eye on* using horizontal pruning because it differs from these ngrams by a length of 1. But the entry *keep your eye on* can also be connected to *keep an eye on* using horizontal pruning because both entries differ by 1 ngram type. Horizontal pruning aids in connecting ngrams which share the same parent ngrams. This utility helps in extraction of determiner-pronoun interchangeability and internal qualitative modifiers.

We take the 2000 idioms present in the IMIL dataset and process them automatically in order to be used as search entries into StringNet. The processing involves two features; lemmatization, and

generalization of pronouns and determiners into generic entries *[pron]* and *[det]* respectively. An entry *keep an eye on* becomes *keep [det] eye on*. In addition to searching the term, we also search the idiom in both directions through one level each of vertical and horizontal pruning. This results in the extraction of 81562 sentences containing instances from 758 of the 2000 idioms.

3.2.2 Candidate Idioms Selection

As part of preprocessing before the StringNet extraction, we lemmatize the idioms to search base forms, in order to have a wider coverage of the search. In that process, some of the extracted idioms have the same lemmatized forms. For examples, idioms like *music to my ear* and *music to my ears* are essentially the same idiomatic expression with the same underlying semantics i.e. *pleasant to hear*. The difference in the two expressions is only of the number inflection, which disappears during lemmatization. These multiple entries lead to duplication of search results. In order to avoid this redundancy, we club idioms with same lemmatized form into a single entry for the idiom. Using this method, we filter out redundant idioms from our idioms list, removing duplicate entries of instances from the sentences. In addition to this, we also remove those idioms, which do not have a single idiomatic usage present in the resulting sentences. This step results in filtering 749 idioms and 77894 sentences. The idioms that remain are unique and have idiomatic usages.

3.2.3 Candidate Instances Selection

Idiomatic Expressions are also idiosyncratic in the kind of lexical variations they allow in their occurrences. For example, the idiom *keep an eye on* can occur as *keep your eye on* but *give me a hand* cannot occur as *give me your hand*. These differences are very idiom specific and have no evident paradigmatic reasoning associated with it. It is very local to the idiom and its conventionality of usage. In order to resolve this issue, each instance of the idiom needs to be checked and filtered accordingly. Therefore, in this step, we filter out those lexical variations of idioms, which will never occur idiomatically. In order for an efficient extraction of correct patterns to occur, we manually divide the idioms list into two categories based on [15].

3.2.3.1 Static Idioms

Static idioms are idioms which do not undergo any lexical modification. We identify 388 idioms as Static in our idioms list. These idioms have 45955 instances in the data. Since, these idioms do not undergo any lexical modification, any occurrence which does not contain an exact occurrence of these idioms is not idiomatic. For example, the idiom *by and large* is a static idiom with no other form occurring as idiom. It cannot occur as *by or large* or *by and largest*. Therefore, we filter out sentences which did not have an exact occurrence of the idiom. If no exact occurrence of an idiom is found, we reject the idiom altogether. At the end of this step, 21891 sentences with 359 Static idioms are left.

3.2.3.2 Formal Idioms

Formal Idioms are idioms which occur in sentences with various lexical modifications. We identify 361 idioms from our idioms list as Formal idioms based on their occurrences. These idioms have 31939 instances in the data. As this task requires more flexibility and complexity than Static idioms, an completely automatic approach is not feasible. At the same time, going through the whole dataset sentence by sentence is quite inefficient. Thus, in order to efficiently sift through the data, we extract the unique variations of each idiom and then manually remove the irrelevant occurrence patterns, thus removing all sentences with those occurrences. For this task, we index all phrases of a particular idiom, and create a multi level tree with the idiom as the root and all phrasal occurrences of an idiom are its children. This requires extraction of specific patterns which are relevant exclusively to particular idioms. The children of each phrasal occurrence are the sentences in which that phrase occurs exactly. We remove all the branches of phrasal occurrences of a particular idiom, which will not occur idiomatically. This reduces our load by a scale factor of 1/3 as the unique occurrences are around 10000 in number. This process does not reduce the number of idioms to large extent(358) but we do filter out a considerable number of patterns, resulting in only 3135 remaining sentences.

3.2.4 Final Result

Finally we create a dataset of 717 idioms in 25026 sentences/instances. Each of these 717 idioms has idiomatic and literal usages present in the dataset and each phrasal instances of these idioms can occur idiomatically in a sentence. For the downstream task of phrasal detection, we separate the data into two groups; Static and Formal idioms. We create this distinction in our data because detection of both categories of idioms require separate steps. Static idioms can be treated like words-with-spaces([33]) and the simple heuristic of finding the exact match in the sentence is enough for detecting their lexical occurrence. Formal idioms detection cannot be treated as words-with-spaces as they occur in a lot of forms and variations in sentences. Formal idioms requires a more complex approach which can identify the similarities between instances of the same idiom and their difference from other phrases. Number of sentences and idioms left after each step are given in Table 3.1. The first three rows show the results for the total data extraction while the subsequent rows show extraction results for Formal and Static idioms separately.

We are also interested in finding the spread of each idiom in our idioms list. In this effort, we calculate the total instances of each idiom and calculate the mean and standard deviation on the resultant counts respectively for Formal idioms and Static idioms. Table 3.3 shows the mean and standard deviation of both the Formal idioms dataset and Static idioms dataset with respect to their number of occurrences in data. The mean and standard deviation for Formal idioms are very close which suggests an exponential distribution whereas the Static idioms show a skewed distribution.

Extraction Step	Sentences	Idioms
StringNet Extraction	81562	758
Candidate Idioms Selection(Total)	77894	749
Candidate Instances Selection(Total)	**25206**	**717**
Candidate Idioms Selection(Static Idioms)	45955	388
Candidate Instances Selection(Static Idioms)	**21891**	**359**
Candidate Idioms Selection(Formal Idioms)	31939	361
Candidate Instances Selection(Formal Idioms)	**3135**	**358**

Table 3.1: Number of Sentences and Idioms left after each extraction step

Test Dataset	Accuracy	Precision	Recall
Formal Idioms Test Dataset	**0.98**	**0.95**	**0.91**
SemEval All Words Dataset(all usages)	0.84	0.90	0.85
SemEval All Words Dataset(idiomatic usages)	0.86	0.93	0.86
SemEval Lex Sample Dataset(all usages)	0.89	0.90	0.90
SemEval Lex Sample Dataset(idiomatic usages)	0.92	0.95	0.92
PIE Corpus(all usages)	0.69	0.60	0.69
PIE Corpus(idiomatic usages)	0.88	0.94	0.88

Table 3.2: Test Results from the model trained on Formal Idioms Training Dataset. Formal Idioms Test Dataset is 25% split from the Formal Idioms Dataset. All datasets have been tested separately for *All Usages* and *Only Idiomatic usages* of potentially idiomatic expressions in sentences

3.3 Experiments

We use our Formal idioms dataset containing 3135 sentences to train on a typical sequence labelling neural network. For each sentence, we create a parallel sequence labelling encoding using the BIO convention. We label each space separated token in the sentence as either 0, B-IDIOM, I-IDIOM.

- B-IDIOM: Token is the beginning token for the idiomatic phrase.
- I-IDIOM: Token is part of the idiomatic phrase
- O: Token is not part of the idiomatic phrase

The sequence labelling encoding acts as the target and the original sentence is the source for our task. We do a 75-25 train-eval split on our dataset for our training and evaluation. We use a BiLSTM-CRF [20] module for our task. We use 300 dimensional glove embeddings[29] as our embedding input. We use LSTM hidden representation of dimension 100 and batch size of 20. We train the model for 25 epochs. All the hyperparameters are also present in table 3.4.

In addition to the Formal idioms test dataset, we use three independent datasets for testing in order to test the efficacy of the trained sequence labelling model. These datasets are mentioned as follows:

- ”All words” training dataset from [22] containing 1143 sentences. All sentences contain potentially idiomatic phrases, each usage is labelled with *idiomatic*,*literal* or *both* usage.

Idiom Type	Sentences	Mean	Std Dev
Formal	3135	8.75	8.61
Static	21891	60.9	160

Table 3.3: Mean and Standard Deviations of Final Datasets

Hyperparameter	Value
Embedding dimension	300
Hidden dimension size	100
Batch size	20
Epochs	25
Train Test Split	75-25

Table 3.4: Hyperparameters used for training

- ”Lex sample” training dataset from [22] containing 1423 sentences. All sentences contain potentially idiomatic phrases, each usage is labelled with *idiomatic*,*literal* or *both* usage.
- PIE corpus[18] containing 2239 sentences. All sentences contain potentially idiomatic phrases, each usage labelled with a sense label,”y” meaning idiomatic usage and ”n” meaning literal usage.

We evaluate our models on two versions of each of the three datasets: All samples and samples labelled with idiomatic usages.

3.4 Results

The Results can be seen in Table 5.4. We see that the Formal idioms test dataset gives the best results because of similarity with the training dataset. However, the model also gives good results with other independent datasets. The *idiomatic usages* set for each dataset consistently gives better results than the *all usages* set. This is due to the *Candidate Instance Selection* process where we remove all the instances of idioms which can never occur idiomatically. The *all usages* sets of the independent datasets contain such examples, and the model, rightly does not label these occurrences as possibly idiomatic.

3.5 Conclusion

In this work, we present a semi-automatic approach to create a new dataset of labelled potentially idiomatic expressions in 25206 English Sentences extracted from the BNC corpus[25] with high accuracy. We segregate our dataset into two categories, Formal and Static. The Formal idioms dataset consists of 3135 sentences containing 358 formal idioms and the Static idioms dataet consists of 21891 sentences containing 359 static idioms. This distinction is done because of the difference in the potentially idiomatic span detection mechanisms of these categories. We introduce two different approaches

for detection of these possibly idiomatic phrases. For static idioms, we suggest a direct search approach as these idioms can be treated as words-with-spaces. For formal idioms, we suggest modelling the problem as a sequence labelling task. We create a sentence parallel sequence labelling encoding to be used as the target for the task with good results. We make all data publically available for further research.In the next chapter, we discuss step 2 of idiom handling i.e. disambiguation of idiomatic and literal usage of an idiom.

Chapter 4

Labelled EPIE: A Dataset For Idiomaticity Detection

4.1 Introduction

Polysemy is an inherent part of language. The expressivity of language originates because words can be used in multiple senses in multiple contexts. For Natural Language Understanding (NLU), this behaviour poses a challenge. In order to comprehend a sentence, each word's sense needs to be disambiguated from multiple definitions and senses that exist for that word. Word Sense Disambiguation (WSD) is a domain which has always been a source of interest for the NLP community. Earlier employing knowledge sources like WordNet ([27]) as they provide a great wealth of relational knowledge in structured form (i.e., hypernymy, meronymy, similarity, etc.), WSD is now generally solved using neural networks. With the help of neural networds, the gap between supervision i.e. the data used for training and knowledge i.e. the data that the pretrained language models have seen, is reduced by learning effective vector representations in the same space as contextualized word embeddings. [6] provides a survey stating the current state of the art in WSD. It states that according to the context, ambiguity in WSD is resolved by mapping a target expression to one (or maybe more) of its potential senses. WSD systems use the senses that are listed in a static, predetermined, machine-readable dictionary, or sense inventory. The sense inventory for a language in WSD can be both very vast, with hundreds of thousands of concepts, and very sparse, with only one concept per lexeme is only connected to a small portion of the sense inventory. In today's supervised models, which are built on neural networks, the task is framed as a classification issue, and annotated data is used to learn the associations between words and sensations. As opposed to this, knowledge-based systems frequently use graph algorithms on a semantic network, where senses are connected via semantic relations and are explained with definitions and usage examples.

While WSD is able to disambiguate the sense of a word from a given set of predefined senses, metaphorical usage is a problem that is still outside the purview of WSD. If a word is used in a metaphorical sense, it means that the sense exhibited by the word in a particular sentence, is not among the predefined finite set of senses. The detection of metaphors in natural language demands difficult contextual reasoning about whether particular scenarios can actually occur literally. The metaphorical sense

usually prevails from the unconventionality or impossibility of these scenarios. Our ordinary communication frequently uses metaphors to provide nuanced imagery and help us make sense of our experiences with our conceptual framework. The most significant account Lakoff and Johnson[23] propose a systematic metaphorical association between two different concepts or domains as the best explanation of metaphor to date. For instance, whether we discuss "treating juvenile delinquency" or "government corruption spreading, "rankings", we consider the characteristics of a sickness to be the broad notion of crime (the goal concept) (the source concept). Our usage of metaphor in language is a reflection of how we project knowledge and inferences across areas using such broad generalisations as metaphorical linkages. [16] present an end-to-end neural models for spotting the usage of metaphorical language in sentences. In contrast to earlier work that used more limited forms of linguistic context, they demonstrate that relatively typical BiLSTM models that operate on whole sentences work well in this situation. They also show that even with a small quantity of training data, very simple architectures based on bidirectional LSTMs[19] and enhanced with contextualised word embeddings[17] perform very well on both tasks.

Although metaphor detection detects metaphor usage at a word level, metaphorical usage at phrase level is still an tough problem to solve. Phrasal metaphors, better known as idiomatic expressions, are phrases which are conventionally used as metaphors, rather than literal usage. The issue with idiomaticity detection for idioms is that the literal meaning of the phrase may seem impossible for the remaining part of the sentence, i.e. it is out of context with the rest of the sentence. But the words of the phrase can be in context or aligned with each other. For example, the sentence *By giving the employee the possibility of a bonus, you have given a dog a bone* has the idiom *to give a dog a bone*. In this example, the idiom has words *dog* and *bone* which are closely related to each other and the context has words *employee* and *bonus* which are closely related to each other. But the idiom words and the rest of the context are not related and hence, are out of place for each other, which is a sign of idiomatic usage. Therefore, idiomaticity of lexical occurrences of idiomatic expressions is a more complicated issue to resolve.

Idiomatic Expressions have garnered a lot of interest over the years in the NLP community. They are also sometimes referred to as multi word expressions(MWEs). These expressions occur in a variety of linguistic contexts and therefore, a linguistically precise handling of such expressions has a lot of challenges, some of them highlighted by [39]. [33] and [7] talk about the different complexities associated with handling idiomatic expressions and a lack of clear determinable patterns in different instances of the same idiomatic expressions in data. Their inherent syntactic complexities and high frequency in data has made it unavoidable to handle them in a special manner in order to have better machine comprehension in NLP. Additionally, [33] also makes a distinction among idiomatic expressions, namely static idioms and formal idioms. Static Idioms are syntactically fixed idiomatic expressions i.e. they occur in the same surface form in all the contexts whereas formal idioms are syntactically flexible. Formal idioms can occur with multiple qualitative modifiers and can occur in different surface forms and different pronominal attachments. This linguistic freedom that formal idiomatic expressions exhibit, makes

them more difficult to handle than static idioms. In addition to the syntactic flexibility, a lot of formal idiomatic expressions also occur in literal contexts as well. For example, *drag one's feet* is a very common formal idiomatic expression, the meaning of which corresponds to *being slow to act*. However, on multiple instances, *drag one's feet* is used in a literal context, like in the sentence 'Children tend to *drag their feet* while walking'. These occurrences of formal idiomatic expressions in literal context makes any kind of automatic handling based on syntactic behaviour impossible. Therefore, it becomes important to have models trained on disambiguating such instances necessary for any meaningful resolution to the problem. This problem can be modelled as a classification task, with the idiomatic expression sequence getting assigned a label, thus, disambiguating the idiomaticity of the phrase in the given context. However, classification models are like any other machine learning NLP models i.e. they are dependent on good training data which in this context would be data which contains literal and idiomatic usages for a wide variety of idiomatic expressions. [26] also identifies this as step 2 of the predefined steps of handling idiomatic expressions, the first being detection of such lexical occurrences.
It is with this motivation that we identify our task of creating a labelled dataset, labelling both idiomatic and literal instances of multiple formal idiomatic expressions. We annotate the formal idiom instances of the EPIE dataset [35]. The EPIE dataset contains 25,207 instances of 717 idiomatic expressions. We label a subset of this dataset, containing instances of only the formal idiomatic expressions. We label a total of 3,136 instances, denoting idiomatic or literal usage of 358 formal idiomatic expressions. We also use the labelled dataset to train a sequence classification model inspired from the work of [16] on metaphorical word detection. We adapt a transformer based classification model for our task, with training resulting in high accuracy, precision and recall scores. We also test our trained model on three independent labelled datasets to check the coverage of our model. These datasets are, "all words" and "lex sample" datasets of SemEval-2013 Task 5b Dataset [22], and the PIE Corpus [18]. For all these datasets we have used the whole corpus i.e. training, dev and test datasets together for testing.

The major contributions for our work can be summarized as follows:

- We publicly release a labelled EPIE formal idioms dataset containing 3136 instances of 358 formal idioms. Each instance is labelled with 1 or 0, with 1 denoting an idiomatic usage and 0 denoting a literal usage. These instances have been manually annotated by two annotators with perfect inter annotator agreement.

- We extend the work of [35] by classifying the multi-word expression extracted with [35] and disambiguating whether it is in idiomatic usage or literal usage in a sequence.

- We present a transformer based architecture tweaked for our task which is simpler that recent state-of-the-art models such as [32] without compromising performance. We present the efficacy of our dataset and trained model by performing cross-dataset evaluation on three independent datasets.

4.2 Data

We have used the EPIE dataset [1] [35] for our task. The EPIE dataset consists of $25,207$ samples where the sentences are extracted from the British National corpus (BNC) [8] with lexical occurrences of idioms extracted from the IMIL dataset [1]. The EPIE dataset classifies indiomatic instances into two categories.

- Static Idioms Dataset- The static idioms dataset contains 21891 instances/sentences containing 359 distinct idioms. These idioms do not change their lexical form across usages.
- Formal Idioms Dataset- The formal idioms dataset contains 3136 instances/sentences containing 358 distinct idioms. These idioms occur in different lexical forms, inflecting for number, tense, etc based on the context.

However, the EPIE dataset only contains the instances but it does not disambiguate between idiomatic usage versus literal usage of the potential idiomatic phrase within each sample.

For our task, we have manually annotated the EPIE formal idioms samples, which is a subset of EPIE dataset containing 3136 samples. Formal idioms may undergo lexical modifications in their usage in a sentence. This makes detection and disambiguation of such expression difficult for automatic systems. For example, the formal idiom, *cry over spilled milk*, can be used with lexical modifications in idiomatic usage in 'There is no use *crying over spilled milk*' and 'He *cried over spilled milk* for a whole month' and in literal usage in 'The baby dropped her bottle and started *crying over the spilled milk*'. We label each instance of the formal idioms with 1 for idiomatic usage in the sample and 0 for literal usage. We have used two annotators to label all instances and have seen unanimous cross annotator agreement scores for all instances. This is due to the instruction posed to each annotator for annotating each instance - *Label 1(idiomatic) if it is possible to have an idiomatic usage of the given phrase in the given context and 0(literal) otherwise.* Some examples are :

- All that we earn goes into *keeping the wolf from the door* - labelled 1 (idiomatic)
- It is hard to *keep your head above water* - labelled 1 (idiomatic)
- Try to *keep your head above water* when learning to swim. - labelled 0 (literal)
- He is the *black sheep* of the family. - labelled 1 (idiomatic)

Due to the instruction given to the annotators as mentioned above, the ambiguous instances i.e. the instances which can have both idiomatic and literal sense in a given context have also been labelled as idiomatic by the annotators as our first criteria in the instruction, which is having an idiomatic usage possible, is fulfilled. In example 2 above, since it is possible to have an idiomatic usage, it has been labelled as idiomatic. We did this in order to have more idiomatic instances in the dataset to offset the

[1] https://github.com/prateeksaxena2809/EPIE_Corpus

high frequency of literal usages in a general dataset.
Each entry in the dataset contains the candidate idiom, the sentence, the start and end offsets for the idiomatic expressions in the sentence and the idiomaticity label. We label a total of 3136 instances of 358 formal idioms. Our labelled dataset has a total of 2761 idiomatic usage instances and 375 literal usage instances. We publicly release our dataset.

4.3 Model

The Ge Gao model [16] for metaphor detection detects the target metaphor word by comparing it to the context in which it occurs. The intuition is that when a word is used metaphorically in a context, it is less similar to the words of the context than when the same word is used literally. In the following example,

- The heat *ignited* the matchstick.

the word *ignited* is used literally as there is a close relation between the word *ignite* and *matchstick*. However, in this example,

- His speech *ignited* a revolution.

the same word *ignited* is used metaphorically as there is not a close relation between *ignite* and any of the constituent words in the context. Ge Gao's study is limited for single word metaphors. A more recent work on idiomatic expressions as observed in [32] uses Graph Convolution Network (GCN) to identify verbal multi-word expressions. They train a BERT model augmented with a GCN and a multi-headed self-attention layer. The study focuses only on idiomatic phrases which begin with a verb, such as *keep an eye on* and does not handle idiomatic expressions such as *behind his back* which is a prepositional phrase or *black sheep* which is a compound nominal.

To identify idiomatic usage of any multi-word expression, we propose the following two patterns are required to be learnt by our system:

- the constituent tokens of the idiomatic expression have high similarity with each other.
- the constituent tokens of the idiomatic expression have low similarity to the words in the outer context in which they occur.

Our model uses a BERT-based architecture which captures this relationship between a mutli-word expression in metaphorical usage and its context. Using only a BERT model makes our system simpler than BERT+GCN+self-attention yet taking advantage of self-attention mechanism of transformers. To achieve this, we take inspiration from sentence pair modelling using BERT [3, 2, 31] which studies the semantic relationship between 2 sequences such as textual entailment, paraphrasing and sentence-pair similarity. For our task, however, the input sequence is not 2 consecutive sentences, but a phrase(the idiomatic expression) embedded in a larger context(sentence). To enforce this constraint, we use the

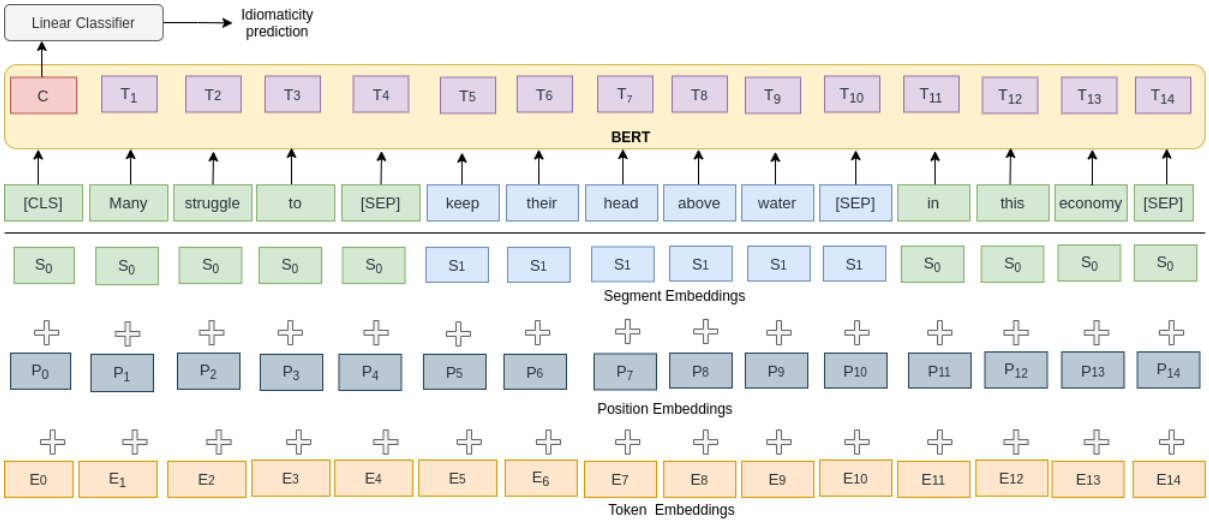

Figure 4.1: Idiom Sense Disambiguation using Bert for Sequence Classification. The target idiomatic expression has segment embeddings for id 1 and the context has segment embeddings for id 0

Hyperparameter	**Value**
Sentence Max Length	110
Number of Epochs	10
Batch Size	8
GPU Size	4GB
Weight Decay	0.01
Warmup Steps	50

Table 4.1: Hyperparameter values used for our model.

token type identifiers of BERT tokenizer. The potential idiomatic expression tokens are annotated with 1 and the context words are annotated by 0. In order to preserve the context information, the idiomatic expression cannot be encoded separately as an independent sequence because then it would appear as an incomplete phrase. Consequently, the context cannot be encoded separately from the idiomatic expression as an independent sequence because then the context would appear as two separate sequences i.e. left context and right context with nothing in the middle. As observed in figure 4.1, the input sequence is encoded into BERT embeddings for each token which encodes the the word embedding, whether the token is idiomatic represented by the token type identifier and the position of each token within the sequence and is transformed by BERT into a sequence of hidden states. The hidden state of the *[CLS]* token is used with a linear classifier to identify the idiomacity of the multi-word expression within the input sequence.

Model	Accuracy	Precision	Recall	F1
Gao et al.(2018)[16]	0.91	0.91	0.98	0.93
BERTBaseline	0.85	0.85	0.99	0.92
BERTModified	**0.95**	**0.96**	**0.98**	**0.97**

Table 4.2: Test results comparison with the baseline model. Train, Dev and Test set were split by 80%, 10% and 10% respectively.

Test Dataset	Accuracy	Precision	Recall	F1
Formal Idioms Test Dataset	**0.94**	**0.95**	**0.97**	**0.96**
SemEval All Words Dataset	0.74	0.70	0.98	0.82
SemEval Lex Sample Dataset	0.66	0.60	0.96	0.74
PIE Corpus	0.66	0.63	0.97	0.76

Table 4.3: Test Results from the model trained on Formal Idioms Training Dataset. Formal Idioms Test Dataset is 20% split from the Formal Idioms Dataset. The model has been tested on the whole(train,test and dev) dataset combined for all the other datasets i.e. *SemEval All Words Dataset*, *SemEval Lex Sample Dataset* and *PIE Corpus*.

4.4 Experiments

We use the 80%:10%:10% training, dev and test split on our dataset for the experiments. For our baseline, we have used the classification model of [16], as that seems closest to our task at hand. We have kept the same hyperparameters for our task with 300 dimensional glove embeddings, 1024 dimensional elmo embeddings and 50 dimensional index embeddings. The only difference is that instead of a single token, we have used the index embedding indicator to identify all constituent tokens of the phrase to be classified. We have used the same pretrained options settings and weights for ELMO as [16].

We train a transformer based model for our task. We modify the 'token type ids' output from the BERT tokenizer to identify the idiom expression tokens within the sequence with id 1 and the context tokens with id 0. This in addition to the 'input ids' and 'attention mask' is fed into the transformer based BERT for sequence classification model. We name this model as BERTModified as we send a modified sequence into the model. We also train a model without explictly encoding the idiom phrase span information in the 'token type ids'. However, we insert 2 *[SEP]* tokens demarking the idiom span boundary in the context. This is done to test whether BERT is able to disambiguate the idiom phrase span implicitly from just the tokenization. We name this model as BERTBaseline. We use the 'bert-base-uncased' model from huggingface [41] for our input sequence tokenization and the initialization of weights of our architecture. The hyperparameter values for both our models can be seen in table 4.1.

For our experiments, we train all settings of our model on our annotated data (Formal Idioms Test Dataset) and evaluate it on our annotated test data and 3 independent datasets. We use the SemEval All Words Dataset, SemEval Lex Sample Dataset and PIE Corpus dataset for the cross-dataset evaluation. The SemEval All Words Dataset and the SemEval Lex Sample Dataset are part of the SemEval2013-Task 5 named 'Evaluating Phrasal Semantics' [22]. These datasets are part of the second task in Task

Sentences	**Usage**
All along, as I reported at the time, Sarah wanted to take the baby with her.	Idiomatic
The chilean black dolphin is caught in surface nets *all along* the chilean coast.	Literal
Choosing plans for the whole group inevitably means making some compromises somewhere *along the line*.	Idiomatic
The third brigade was running into counter attacks all *along the line* and was a risk.	Literal
At the end of the day, the board looks at the person as a whole.	Idiomatic
The restaurant is very inexpensive and worth the car trip *at the end of the day*.	Literal

Table 4.4: Correctly labelled Sentences from the independent test datasets.The second column denotes the correct label for the idiomatic phrase. The first 2 sentences belong to the PIE corpus, the second 2 belong to the SemEval all words corpus and the last two sentences belong to the SemEval lex sample corpus.

Sentences	**Usage**
They played some great stuff and ran us *all over the place*.	Idiomatic
He was laughing as I was still to *get off the ground*.	Literal
It is *cutting it close* at this time and someone must be responsible for the intake of the ER.	Idiomatic
If you visit Darlington or other stations *along the line*, you will find things to see and do.	Literal
At the end of the day, I had retrieved most of the file.	Idiomatic
Those who carried the banner for Winwick were proud to see it fluttering aloft *at the end of the day*.	Literal

Table 4.5: Wrongly labelled Sentences from the independent test datasets. The second column denotes the correct label for the idiomatic phrase. The first 2 sentences belong to the PIE corpus, the second 2 belong to the SemEval all words corpus and the last two sentences belong to the SemEval lex sample corpus.

5 which addressed deciding the compositionality of a phrase in a given context. Each line of these dataset has a sentence containing a phrase and a label of 0 or 1 denoting the compositionality of the phrase within the sentence. The dataset has been created using idioms from English Wiktionary using the JWKTL Wiktionary API [42] and usage contexts from the ukWAC corpus[5]. The PIE corpus is an evaluation corpus for the automatic detection of potential idiomatic expressions (PIEs) with usages from 23 documents of the British National Corpus [8]. Each entry of the PIE corpus is a 5 sentence window, with the idiomatic expression possibly present in the middle sentence. Each entry of the PIE corpus is labelled with the PIE label which denotes whether the middle sentence contains the PIE in question, and a sense label which denotes whether the expressions has been used literally or idiomatically. We merge the train, dev and test sets of individual datasets and use all the samples from the respective datasets for evaluating our models.

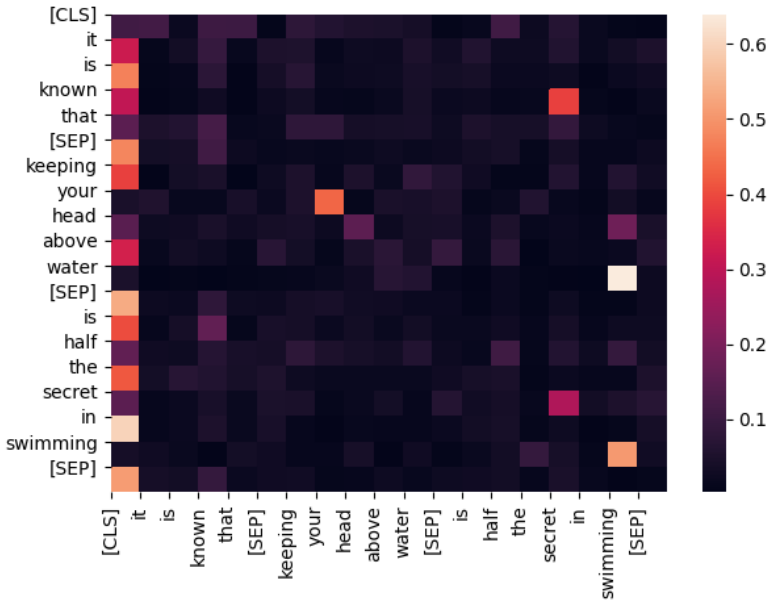

Figure 4.2: Attention Weights of layer 0, attention-head 10, for literal usage in 'it is known that keeping your head above water is half the secret in swimming'. As observed, there is a high correlation between 'water' and context token 'swimming'

4.5 Results

The results comparison with the baseline can be seen in table 4.2. As it is clear from the table, modified bert based transformer architecture outperforms the baseline as well as the unmodified transformer architecture.

The results for tests on the SemEval and PIE corpus can be seen in table 5.4. Clearly, the formal idioms test dataset gives the best results as it is split from the training data and thus, is the most related to the training data. However, the model also gives good results with the SemEval and the PIE corpus. The high recall and low precision test values across the board suggests that, although the model is trained to identify idiomatic usage, it also sometimes identifies literal usages as idiomatic.

Table 4.4 shows some correctly labelled sentences from the independent test datasets. We show one idiomatic and one literal instance from each of the three datasets. As is observed, the model is able to identify correlations like *along* and *coast* in the second example and *trip* and *day* in the sixth example to identify a literal context and a lack of correlations to identify an idiomatic context.

Table 4.5 shows some wrongly labelled sentences from the independents test datasets. For these too, we show one idiomatic and one literal instance from each of the three datasets. As observed in these instances, the model is not able to find correlations between *stations* and *along* or *line* in the fourth example and identify a literal context. It also wrongly associates a correlation between *played* and *place* in the first example to label the instance as literal. In addition to this, some instances in datasets of

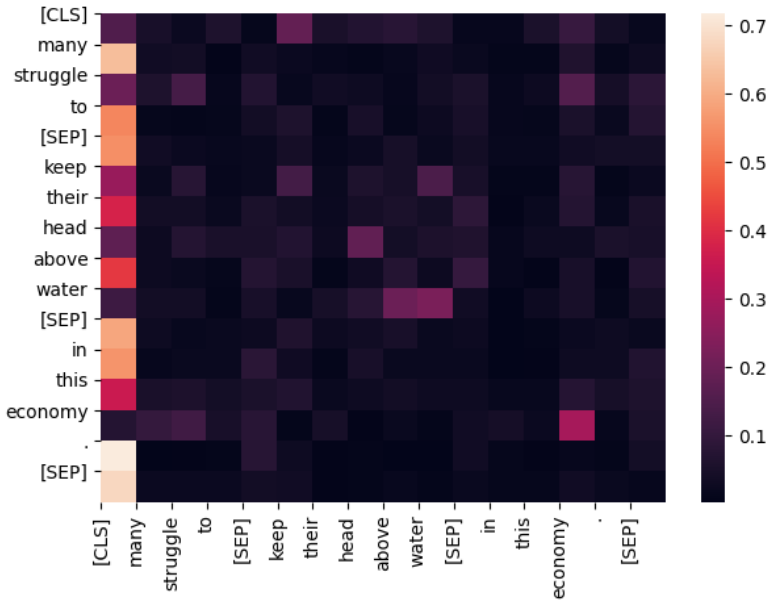

Figure 4.3: Attention Weights of layer 0, attention-head 10 for idiomatic usage 'many struggle to keep their head above water in this economy'. There are no correlations between the tokens of the idiomatic expression and context in the idiomatic usage

idiomatic expressions are ambiguous in general and a label assigned to it by the model, idiomatic or literal, is erroneous or correct only in comparison to the label assigned in the dataset, but not in general.

Figure 4.2 and 4.3 show the correlation between self attention weights for attention head 10 for layer 0 of the trained BERTModified model among words from the idiomatic expression and words in the context for one idiomatic usage example and one literal usage example for the idiomatic expression *keep one's head above water*. The higher the correlation, the brighter the colour of the block. The two sentences used for the correlation are,

- It is clear that *keeping your head above water* is half the trick in swimming. - Literal usage
- Many struggle to *keep their head above water* in this economy. - Idiomatic usage

As observed from the brightest spot in figure 4.2, we see that there is high correlation between the constituent word *water* present in the idiomatic expression and *swimming* present in the outer context of the expression, which results in the model identifying the usage as literal. Compared to this, we see very sparse correlations in figure 4.3, which results in the model identifying the usage as idiomatic.

4.6 Future Work

Our work, with other previous works helps identify idiomatic usages of known multi word expressions in a given context. A possible downstream task to our work can be in machine comprehension of

sentences containing these idiomatic usages. This can be done either by surface realization and phrasal replacement of these phrases into literal translations, or convert the sentences into semantic frames to aid NLP tasks like machine translation, summarization and comprehension.

4.7 Conclusion

In this work, we address the task of idiomaticity classification of potential idiomatic expressions in a given context to aid natural language understanding for downstream tasks like machine comprehension, text summarization and machine translation. To achieve this, we present an annotated EPIE Dataset labelled with idiomatic/literal usage of potential idiomatic phrases. Our dataset contains 3136 labelled occurrences of 358 formal idiomatic expressions. To evaluate the efficacy of our data, we also present a transformer based classification model to label usages of idiomatic expressions in the sentences as idiomatic or literal. We train this model using our labelled EPIE dataset and show that it performs better than the baseline and gives good results on three independent datasets. In the next chapter, we test the utility of our modified model to disambiguate between idiomatic and literal usage of german verbal idioms.

Chapter 5

Disambiguation of German Verbal Idioms

5.1 Introduction

Verbal idioms are idiomatic expressions with a verb and at least one constraint argument slot whose overall meaning cannot be inferred from the meanings that the expression's component parts would convey if they were employed separately in other contexts. The automatic acquisition and identification of compound nouns, adverbs, and other multiword lexical units is particularly problematic but can be dealt with a words-with-spaces approach. Verbal idioms on the other hand are more complex to identify and disambiguate. These idioms fit into sentences naturally because of the verbal word structure, although other metaphorical expressions, such as *a black sheep* or *the tail wags the dog* may be more common in standalone statements. As a result, it is difficult to identify verbal idioms in natural language processing. In the EPIE Dataset, over 70% of the formal idioms have a verbal head. Of these idioms, majority of idioms are verbal idioms, with the minority being the light verb constructions. [4] discusses the issues associated with integrating verbal idioms into an NLP system for the European Portuguese language. [12] explain that in case of verbal idioms, for a lot of cases, the semantics of the phrase can also be understood in a non-figurative literal way, even though the reading maybe implausible. The difference lies in the assumption of the conventionality of usage of the idiom by an English speaker. Although, it can be assumed that some verbal idioms do not and cannot possibly occur in a semantically literal setting, the same cannot be said for other verbal idioms. This can be explained by the following example.

- *Trip the light fantastic* occurs very predominantly in a figurative sense.

  In the sentence 'Tonight, we celebrate by *tripping the light fantastic*', the usage is figurative.

- *Build bridges* can occur in both senses, figurative and literal.

  In the sentence 'After a fallout over the company layoff, the upper management has been *building bridges*', the usage is clearly figurative

  In the sentence 'This company *builds bridges* for train movements throughout the country', the usage in clearly literal

	Literal	Idiomatic	Undecidable	Both	Idiomaticity %
COLF-VID	1511	5386	33	10	77.61%
SemEval 5b data	190	2771	0	0	93.58%
Total	1701	8157	33	10	82.39%

Table 5.1: Dataset spread statistics for the shared task, with contribution from constituent datasets

In the sentence 'Due to the chaos caused by the hurricanes, travel and economy has suffered and the government has been *building brides* since then, to solve the problem', the usage is ambiguous in nature and can be disambiguated with a larger context.

Because of this property, it is imperative to have a system which can disambiguate verbal idioms well. The "Shared Task on the Disambiguation of German Verbal Idioms at KONVENS 2021" by [13] posed this question precisely for German Verbal Idioms. We solve the above shared task using our modified BERT model, and compare it with the solution from the top team on the leaderboard [28].

5.2 Task

The corpus provided by [13] for the shared task are a combination of two dataset; SemEval 2013 task 5b and COLF-VID. Since COLF-VID is a lexical sample corpus, only instances of a predetermined subset of VID types can be found there. Every sentence in COLFVID was taken from the TuPP-D/Z corpus of German newspapers, making it a relatively homogeneous corpus in terms of genre. Three annotators with relatively high Cohen's Kappa scores (0.77, 0.8, and 0.9) labelled the data with the terms LITERAL, FIGURATIVE, UNDECIDABLE, and BOTH.

In many ways, the SemEval 2013 5b dataset for German is quite similar to COLF-VID: it uses a similar label set and includes annotations for the various PIE readings. It is also a lexical sample corpus. Additionally, it was annotated by three individuals, with a very good pairwise agreement of 90% to 95%.

It was fairly simple to integrate the two corpora due to their commonalities. The combined corpus has 9901 occurrences of 67 VID kinds in total. A 70/15/15 split was applied to the data. In order to avoid an imbalance of types in the split dataset, the shared task organisers [13] made sure that the same ratio was applied to each type and not to the dataset as a whole. Additionally, the organisers added examples of three previously undiscovered VID types to the development and test sets, respectively (270 to test, 268 to dev), to evaluate the systems' capacity to generalise. A train set with 6902 instances, a dev set with 1488 instances, and a test set with 1511 instances were the results. The dataset spread is shown in Table 5.1

	Architecture	Pretrained Model
Pannach et. al.	XLM-ROBERTa	xlm-roberta-base
BERT	Bert	dbmdz/bert-base-german-uncased
BERTModified	Bert	dbmdz/bert-base-german-uncased

Table 5.2: Architectures and PreTrained Models used for each experiments

5.3 Model

We model the problem as a multi class classification problem with the four classes from 0 to 3 being LITERAL, IDIOMATIC/FIGURATIVE, UNDECIDABLE and BOTH respectively. We use the Modified BERT architecture from our task of creating the Labelled EPIE dataset in the last chapter as our primary model for this task. We use the token type ids to label the tokens of the possibly idiomatic segment as 1 and the tokens of the rest of the sentence, i.e. the context as 0. This is possible because the segments whose semantics need to be disambiguated have already been detected in the dataset.

For our baseline, we take the unmodified BERT architecture. We also take the [28] model for comparison, as that model gave the best results on the shared task. We only compare with the model without the extended data, as the extended data did not increase the performance of the system. The hyperparameters used for all experiments are given in Table 5.3. The two architectures BERT and XLM-ROBERTa are different in terms of vocab size and number of parameters. BERT has 110 M parameters while XLM-ROBERTa has about 270 M parameters i.e. twice as much as BERT. The vocab size for XLM-ROBERTa i.e. about 250 k, is way bigger than BERT, which is about 30 k.

We use the huggingface([41]) architectures and pretrained models for our task. The architecture and pretrained language model used for each model is shown in Table 5.2. We do not use a language specific model for the XLM-ROBERTa architecture because of two reasons. First, we match the pretrained model used by [28] and secondly, 'xlm-roberta-base' language model has been trained on multilingual data and therefore does not need to be replaced. We use the "dbmdz/bert-base-german-uncased" as our pretrained model for the bert architectures and 'bert-base-uncased' has been trained only on English language data and therefore does not have any understanding for German language. "dbmdz/bert-base-german-uncased" has been trained by the MDZ Digital Library team (dbmdz) at the Bavarian State Library.

For evaluation, we use precision, recall, accuracy and f1 score for comparison. We caluculate all these scores for two classes namely the literal and figurative class. We do not calculate scores for the remaining two classes as the lack of data results in any data getting labelled as the first two classes only. Since most of the data (99%) is labelled either literal or figurative, the task essentially becomes a binary class classification task. However, since such data is present in the test set, each wrong labelling of these classes still results in the reduction in scores for the two classes.

Hyperparameters	Value
Sequence Length	128
Training Batch Size	8
Evaluation Batch Size	8
Optimizer	Adam
Seed	0
Number of Epochs	5
Learning Rate	1e-05

Table 5.3: Hyperparameters for all experiments

Model	R(Lit.)	R(Fig.)	P(Lit.)	P(Fig.)	F1(Lit.)	F1(Fig.)	A
Pannach et. al.(XLM-ROBERTa)	86.4	91.1	66.7	96.5	75.3	93.7	89.8
BERT	83.7	96.3	81.9	96.2	82.8	96.2	93.6
BERTModified	**88.3**	**96.6**	**83.8**	**97.1**	**86.0**	**96.9**	**94.7**

Table 5.4: Results

5.4 Results

The results are shown in the Table 5.4. Because of the space constraint, we shorten Recall score to R, Precision score to P, F1 score to F1 and Accuracy to A. Based on the results, the modified BERT clearly outperforms the other two models. Additionally, the modified BERT also displays closer scores in all metrics for the two classes. The modified BERT score for the literal instances is not as high as the figurative instances but the gap is lesser than the other two models. The highest gap is in the precision score of the two classes; 83.8 and 97.1 i.e. 13.6. The gaps for precision for the unmodified BERT and XLM-ROBERTa are 14.3 and 28.8 respectively.

The unmodified BERT performs better than the XLM-ROBERTa model in most metrics except the recall score for the literal class and precision score of the figurative class. The reasons for the BERT models' better performance than the XLM-ROBERTa model can be multifold. One of the reasons is the specificity of the pretrained model used. As the pretrained model for the BERT models i.e. "dbmdz/bert-base-german-uncased", is trained specifically for German language, they give better results for a monolingual corpus in German. Another reason is the difference in size. The problem may be too simple for a large model like XLM-ROBERTa, which leads to a drop in performance during training.

5.5 Conclusion

This work provides a good evidentiary support for the utility of the modified BERT model for idiom disambiguation to be used for languages other than English as well. The modified BERT model correctly captures the three criteria that need to be checked to compute the idiomaticity of a phrase occurring within a sentence i.e. similarity between context tokens, similarity between phrase tokens and

dissimilarity between context and phrase tokens. It shows that the modified BERT model can be used to detect idiomatic behaviour and can be used as an upstream model for different NLP tasks like machine comprehension, Natural Language Understanding and Machine Translation. In the next chapter, we discuss in place phrasal substitution of idioms as a sequence to sequence task using t5 encoder decoder architecture.

Chapter 6

Phrasal substitution of Idioms

6.1 Introduction

Once an idiom is identified and disambiguated, one of the most important next step is to transfer it into a literal phrase with the same semantics especially in the context of Natural Language Understanding(NLU) and Machine Translation(MT). Since we have earlier observed in Section 1.3, an idiomatic expression need not have a one-to-one correspondence in the target language, one solution to handle the transfer is to reduce the idiom to its semantic structure (through dependency parsing, MRS representation[10], etc). [24] has attempted to handle the task in this way. We discuss their work in greater detail in Chapter 2 Section 2.4.2. They perform manual template lexicalization to create generalized dependency trees for idioms, which can be plugged into dependence trees of sentences containing those idioms. This work offers high precision results and full control over the idioms they handle. However this method has its own challenges. Developing a semantic structural representation for an idiom and then generating a target language sentence from the source involves multiple levels of complexity. The task requires complex linguistic resources like POS tagger, stemmer, lemmatizers and dictionaries in addition to annotators having domain expertise to perform the task, for instance, [24] requires trained annotators on the GenDR system.

The other solution is to consider the semantic transfer as a paraphrasing task, essentially transferring the semantics of an idiom into a literal representation without compromising the semantics. We discuss two such system [21] and [26] in greater detail in Chapter 2 Section 2.4.1. [21] performs this task by a rule based approach by creating symbolic rules for basic string matches for recognition and replacement of 24 idioms as part of the preprocessing before any downstream tasks. This method results in 88% precision and 100% recall for recognition, and successful downstream translation of 18 out of 24 idioms after preprocessing. [26] offers a hybrid solution to solve this problem using two steps, substitution generation and post editing. Substitution Generation is defined by [26] as the process of extracting the core meaning for the full definition of an idiom present in the dictionary. Substitution Generation is done by either the rule based method of identifying boundary words, or the ML based method which identifies proper boundaries from sentence features, definition features and rule-based intervention, based

on surface words and shallow parser outputs. The post editing step is a semi-automatic method of reference resolution like pronouns, grammatical adjustments like inflections and redundancy reduction with boundary smoothing. We attempt to solve this problem as a paraphrasing task, relying on the language modelling capabilities of deep learning models. For such language modeling tasks, deep learning models have been shown to achieve and create state of the art benchmarks. Recently, encoder-decoder attention-based architectures like BERT([11]) and T5([30]) have attained a lot of major improvements in machine translation, paraphrase generation tasks. Our work has two major contributions.

- We release a 3136 sentence aligned corpus from the EPIE Dataset[35] with each idiomatic part of the sentence replaced with its literal counterpart.
- We also release our experiments and results of a paraphrasing task for phrasal substitution of idioms using our dataset. We present a T5 encoder decoder architecture with good results when trained on the dataset.

6.2 Data

We annotate the labelled EPIE[36] dataset which provides idiomatic and literal labels to 3136 sentences of idiomatic occurrences in the EPIE dataset. For the sentences labelled as literal, we treat the same sentence as the target sentence in the parallel corpora. This includes around 400 of the sentences out of 3100 sentences. For the remaining 2700 sentences, we create a csv file with four columns.

- *Source Sentence*: This column contains the source sentence with the idiom. The phrase containing the idiom is distinguished by using all upper case letters for the idiomatic phrase.
- *Meanings*: This column contains all the meanings of the idiom from 'Thefreedictionary.com' with one meaning per line. Consequently, there can be multiple entries aligned for each source sentence.
- *Example Sentences*: This column contains one example sentence for each meaning mentioned in the preceding column. This column only contains an example sentence if 'Thefreedictionary.com' contains an example for a particular meaning.
- *Target Sentence*: This column is for the annotator to fill the target sentence with the literal replacement for the idiom.

A snippet of the annotation can be seen in 6.1.

The columns 'Meanings' and 'Example Sentences' are aligned i.e. each example sentence shows the idiom being used in the sense of the meaning in the preceding column. For our dataset, we add the literal labelled sentences as well in order to experiment with the model's capabilities to distinguish between the instances and perform replacement only for the idiomatic instances.

Source Sentence	Meanings	Example Sentences	Target Sentence
We all do , really , and we 're KEEPING OUR HEADS ABOVE WATER most beautifully .	1. Stay calm, retain self-control, as in When the rowboat capsized, George yelled that everyone should keep their head and hold onto the boat . This usage dates from the early 1600s and is about two centuries older than the antonym, lose one's head, meaning "to become confused and agitated," as in Whenever the stock market goes down sharply, people seem to lose their heads and sell.		We all do , really , and we 're KEEPING OUR HEADS ABOVE WATER most beautifully .
	2. keep one's head above water. See head above water.		
	To be and remain in a calm, stable, sensible, and pragmatic state or condition despite stress.	My father has always been a rock of level-headed judgment and advice. Even during our family's lowest points, he's always kept his head.	
KEEPING YOUR HEAD ABOVE WATER is half the secret in sport and young Karen Rake seems to be made of the right stuff .	1. Stay calm, retain self-control, as in When the rowboat capsized, George yelled that everyone should keep their head and hold onto the boat . This usage dates from the early 1600s and is about two centuries older than the antonym, lose one's head, meaning "to become confused and agitated," as in Whenever the stock market goes down sharply, people seem to lose their heads and sell.		KEEPING YOUR HEAD ABOVE WATER is half the secret in sport and young Karen Rake seems to be made of the right stuff .
	2. keep one's head above water. See head above water.		
	To be and remain in a calm, stable, sensible, and pragmatic state or condition despite stress.	My father has always been a rock of level-headed judgment and advice. Even during our family's lowest points, he's always kept his head.	

Figure 6.1: Annotation File Example

6.3 Model

For training we use the attention based encoder decoder architecture([38] for our task. We use the huggingface implementation of the T5 encoder decoder architecture([30]). T5 is an encoder-decoder model that has already been trained on a variety of tasks that are both supervised and unsupervised and are each translated into a text-to-text format. We warm start both the encoder and decoder with weights from “t5-small”. T5-small contains 60 million parameters.

6.4 Experiments

Our initial set of experiments consist of two questions.

- Can the model learn the literal phrasal replacements of idioms?
- Can the model disambiguate between literal and idiomatic examples of the same idiom and only perform replacement for the idiomatic phrases?

To test the model's capabilities to answer both of these questions, we divide the experiments in two parts, one with all the samples as the training set and the other with only the idiomatic samples as the training set. This way, if the model performs similarly for both means that the model is able to distinguish between instances.

As the number of samples for each idiom is not same throughout the dataset, we pose a third question.

- Is there a minimum number of samples needed by the model for each idiom in order to learn the phrasal replacement ?

To test the model's capabilities for answering this question, we divide the experiment further into multiple parts based on the training data we use. For our first task we only take samples of idiom which have more than 50 instances. We continue doing the same for 40, 30, 20 , 10 and 0 instances. The 0 instance experiment is our initial experiment as 0 signifies no minimum number of samples required. We do this set of experiments for idiomatic samples separately. We create our training and testing data based on even split for each idiom, in order to have even distribution in training and testing data. Total Idioms per sample size is provided in 6.1 for the first set and 6.2 for the second set.

Minimum Samples	**Number of Idioms**
50	1
40	2
30	6
20	26
10	102
0	358

Table 6.1: Number of Idioms whose have at least the given number of minimum samples in the dataset

Minimum Idiomatic Samples	**Number of Idioms**
50	1
40	2
30	5
20	20
10	86
0	340

Table 6.2: Number of Idioms whose have at least the given number of minimum idiomatic samples in the dataset

For both set of experiments, we use a 95-5 train-test split. The rest of the hyperparameters are mentioned in Table 6.3. We use rouge score to evaluate the predictions.

Hyperparameters	Value
Sequence Length	50
Training Batch Size	4
Evaluation Batch Size	4
Optimizer	Adam
Weight Decay	0.01
Number of Epochs	30
Learning Rate	2e-3

Table 6.3: Hyperparameters for all experiments

6.5 Results

Tables with full set of results for the all samples set and only idiomatic samples set are given in Table 6.4 and Table 6.5 respectively. Our initial pair of experiments (with minimum samples 0) yield around 85 % F-score. We notice that since most of the sentence in the source and target is same(except the idiom), the first thing that the model learns is to replicate the source as output. This is the reason that the model converges to 80% score in the first epoch itself as replicating is a very easy task for the model to learn. As the all samples contain samples which are also literal in nature and thus have the same source and target sentences, the model trained on all samples gives a marginally higher score. To circumvent this issue, we train the model with data having a predefined minimum number of samples for each idiom.

The first question posed in our experiment was whether the model is capable of disambiguating idiomatic from literal usage and perform phrasal substitution only for the idiomatic usage. We see that both set of experiments give similar results, proving the model's capabilities of handing the disambiguation well.

We also see that a minimum samples results for both experiments in Graph 6.2 and Graph 6.3. It is clear from the results that the model performs well for idioms when a minimum number of samples are provided in the training data. For the all samples set of experiments results in Graph 6.2, we see peak performance is achieved at the 40 minimum samples benchmark, whereas for the idiomatic samples set of experiments results in Graph 6.3, we see peak performance at the 30 minimum samples benchmark.

However, we also see that the relation between the minimum number of samples and score is not directly proportional. For the all samples set, the model trained on minimum 40 samples per idiom gives better results than 50 samples, even though the former has to learn paraphrasing for 2 idioms as compared to 1 idiom for the later. Similar results are seen for the models trained on only the idiomatic samples. The model trained on minimum 30 samples per idiom gives better results than its 40 sample and 50 sample counterparts even though it has to learn paraphrasing for 5 idioms. This result means that once the model has seen a given amount of variation per idiom, more samples does not necessitate better results.

Minimum Samples	Precision	Recall	F1
0	90.1	84.9	86.7
10	95.0	89.8	91.7
20	96.4	93.0	94.0
30	97.7	93.9	95.1
40	**99.7**	**97.1**	**98.3**
50	99.1	96.7	97.8

Table 6.4: Results for all samples

Minimum Samples	Precision	Recall	F1
0	89.2	83.1	85.1
10	91.7	88.4	89.6
20	96.5	90.9	93.2
30	**99.7**	**97.3**	**98.3**
40	96.5	92.8	94.4
50	99.1	96.7	97.8

Table 6.5: Results for idiomatic samples

6.6 Conclusion

In this chapter, we test our formal idioms sentence aligned corpus for a phrasal substitution task of idiomatic phrases to literal phrases. The trained model gives good results with the caveat that there needs to be a minimum number of samples for each idiom for the model to learn the paraphrased surface form. The model also shows capabilities of disambiguating between idiomatic and literal usage of the idiom and performing phrasal substitution only for the idiomatic usage. In the next chapter, we give the conclusion of the thesis.

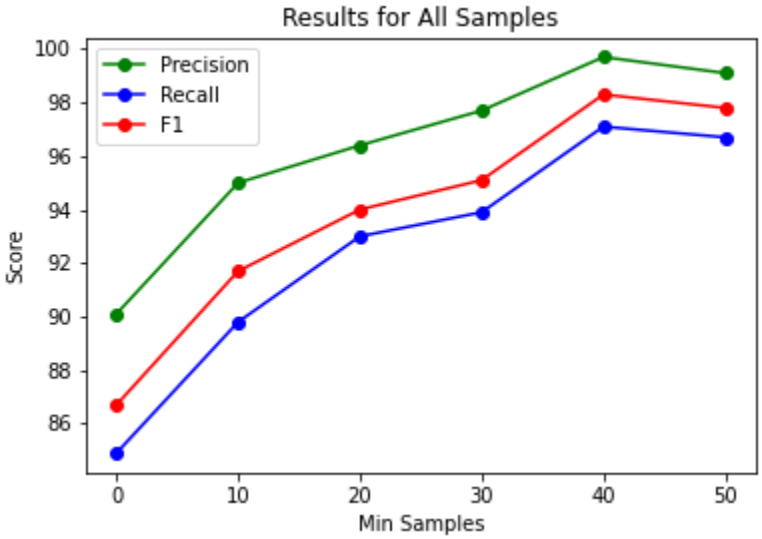

Figure 6.2: Results for experiments done with both idiomatic and literal samples

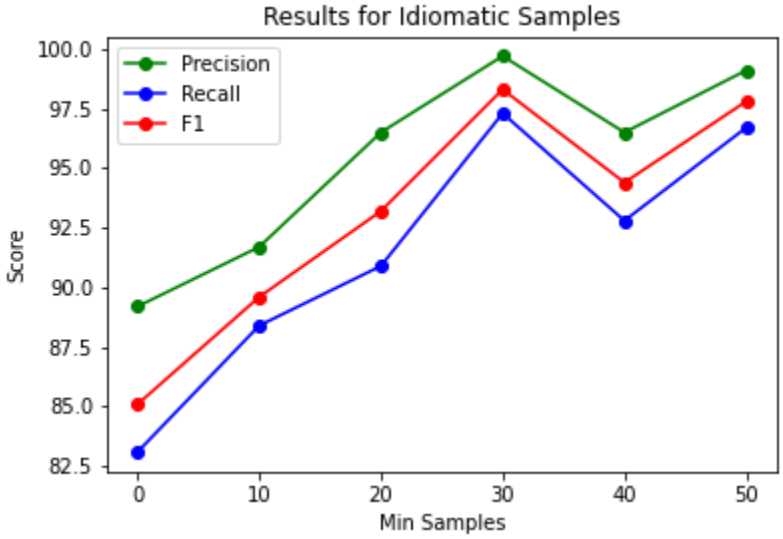

Figure 6.3: Results for experiments done with only idiomatic samples

Chapter 7

Conclusions and Future Work

Through this work, we have tried to aid in handling idiomatic expressions as idiomatic expressions have been difficult to handle for tasks like Natural Language Understanding and Machine Translation. We split the task into three steps. Each following section describes each task and our contribution to the task.

7.1 Idiom Detection

In this task, we detect lexical instances of commonly known idiomatic expressions for other downstream processes. As part of this task, we publically release 25000 sentences with above 700 distinct idioms with lexical instances of idiomatic expressions. We also release a model trained on the released dataset for sequence labelling any idiomatic instance.

7.2 Idiomaticity Detection

In this task, we detect whether the labelled lexical instance of an idiom is used idiomatically in the sentence or literally. As part of this task, we publically release idiomaticity labels for 3136 sentences containing 358 distinct idioms. We also release a modified BERT model training of the released dataset for idiomaticity classification tasks. We also use our model to solve a shared task for German Verbal Detection in which we beat the top model in the leaderboard.

7.3 Phrasal Substitution

In this task, we substitute the idiomatic phrase of the sentence with a literal counterpart. As part of this task, we publically release a sentence aligned corpus of 3136 sentences, with each sentence containing an idiom aligned with another sentence with a literal counterpart. We show that training a T5 model with this data results in good scores for the substitution task. We also show that the model is

dependent on enough samples per idiom to correctly understand the idiosyncrasies associated with each particular idiom.

7.4 Future Work

With this work, we showcase a method of handling idiomatic expressions in English. However, our work needs to be utilized for a lot of downstream tasks to test improvement in performances across the domain. Also, further idiomatic expressions need to be added for better coverage.

Related Publications

- Saxena, P. and Paul, S., 2020, September. EPIE dataset: a corpus for possible idiomatic expressions. In International Conference on Text, Speech, and Dialogue (pp. 87-94). Springer, Cham.

  (https://link.springer.com/chapter/10.1007/978-3-030-58323-1_9 *(Accepted for oral presentation as full paper*)

- Saxena, P. and Paul, S., 2021, September. Labelled EPIE: A Dataset for Idiom Sense Disambiguation. In International Conference on Text, Speech, and Dialogue (pp. 210-221). Springer, Cham.

  (https://link.springer.com/chapter/10.1007/978-3-030-83527-9_18 *(Accepted for oral presentation as full paper*)

Bibliography

[1] R. Agrawal, V. C. Kumar, V. Muralidaran, and D. Sharma. No more beating about the bush: A step towards idiom handling for indian language nlp. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, 2018.

[2] M. Ahmed and R. E. Mercer. Efficient transformer-based sentence encoding for sentence pair modelling. In M.-J. Meurs and F. Rudzicz, editors, *Advances in Artificial Intelligence*, pages 146–159, Cham, 2019. Springer International Publishing.

[3] Y. Arase and J. Tsujii. Transfer fine-tuning of bert with phrasal paraphrases. *Computer Speech Language*, 66:101164, 2021.

[4] J. Baptista, N. Mamede, and I. Markov. Integrating verbal idioms into an nlp system. In *International Conference on Computational Processing of the Portuguese Language*, pages 250–255. Springer, 2014.

[5] M. Baroni, S. Bernardini, A. Ferraresi, and E. Zanchetta. The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, 43(3):209–226, 2009.

[6] M. Bevilacqua, T. Pasini, A. Raganato, R. Navigli, et al. Recent trends in word sense disambiguation: A survey. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*. International Joint Conference on Artificial Intelligence, Inc, 2021.

[7] F. Cap, M. Nirmal, M. Weller, and S. S. Im Walde. How to account for idiomatic german support verb constructions in statistical machine translation. In *Proceedings of the 11th Workshop on Multiword Expressions*, pages 19–28, 2015.

[8] B. Consortium et al. British national corpus. *Oxford Text Archive Core Collection*, 2007.

[9] P. Cook, A. Fazly, and S. Stevenson. The vnc-tokens dataset. In *Proceedings of the LREC Workshop Towards a Shared Task for Multiword Expressions (MWE 2008)*, pages 19–22, 2008.

[10] A. Copestake, D. Flickinger, C. Pollard, and I. A. Sag. Minimal recursion semantics: An introduction. *Research on language and computation*, 3(2):281–332, 2005.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[12] R. Ehren, T. Lichte, L. Kallmeyer, and J. Waszczuk. Supervised disambiguation of german verbal idioms with a bilstm architecture. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 211–220, 2020.

[13] R. Ehren, T. Lichte, J. Waszczuk, and L. Kallmeyer. Shared task on the disambiguation of german verbal idioms at konvens 2021. *Proceedings of the Shared Task on the Disambiguation of German Verbal Idioms at KONVENS*, 2021.

[14] A. Fazly, P. Cook, and S. Stevenson. Unsupervised type and token identification of idiomatic expressions. *Computational Linguistics*, 35(1):61–103, Mar. 2009.

[15] C. J. Fillmore, P. Kay, and M. C. O'connor. Regularity and idiomaticity in grammatical constructions: The case of let alone. *Language*, pages 501–538, 1988.

[16] G. Gao, E. Choi, Y. Choi, and L. Zettlemoyer. Neural metaphor detection in context. *arXiv preprint arXiv:1808.09653*, 2018.

[17] M. Gardner, J. Grus, M. Neumann, O. Tafjord, P. Dasigi, N. Liu, M. Peters, M. Schmitz, and L. Zettlemoyer. Allennlp: A deep semantic natural language processing platform. *arXiv preprint arXiv:1803.07640*, 2018.

[18] H. Haagsma, M. Nissim, and J. Bos. Casting a wide net: Robust extraction of potentially idiomatic expressions. *arXiv preprint arXiv:1911.08829*, 2019.

[19] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[20] Z. Huang, W. Xu, and K. Yu. Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991, 2015.

[21] T. Khanna. *Rule-based pre-processing of idioms and non-compositional constructions to simplify them and improve black-box machine translation*. PhD thesis, International Institute of Information Technology Hyderabad, 2021.

[22] I. Korkontzelos, T. Zesch, F. M. Zanzotto, and C. Biemann. Semeval-2013 task 5: Evaluating phrasal semantics. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 39–47, 2013.

[23] G. Lakoff and M. Johnson. *Metaphors we live by*. University of Chicago press, 2008.

[24] M. D. F. Lareau. Handling idioms in symbolic multilingual natural language generation.

[25] G. N. Leech. 100 million words of english: the british national corpus (bnc). 1992.

[26] C. Liu and R. Hwa. Phrasal substitution of idiomatic expressions. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 363–373, 2016.

[27] G. A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[28] F. Pannach and T. Dönicke. Cracking a walnut with a sledgehammer: Xlm-roberta for german verbal idiom disambiguation tasks. *Proc. of the Shared Task on the Disambiguation of German Verbal Idioms at KONVENS*, 2021.

[29] J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[30] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67, 2020.

[31] N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *CoRR*, abs/1908.10084, 2019.

[32] O. Rohanian, M. Rei, S. Taslimipoor, and L. A. Ha. Verbal multiword expressions for identification of metaphor. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2890–2895, Online, July 2020. Association for Computational Linguistics.

[33] I. A. Sag, T. Baldwin, F. Bond, A. Copestake, and D. Flickinger. Multiword expressions: A pain in the neck for nlp. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 1–15. Springer, 2002.

[34] G. Salton, R. Ross, and J. Kelleher. An empirical study of the impact of idioms on phrase based statistical machine translation of english to brazilian-portuguese. 2014.

[35] P. Saxena and S. Paul. Epie dataset: A corpus for possible idiomatic expressions. In *International Conference on Text, Speech, and Dialogue*, pages 87–94. Springer, 2020.

[36] P. Saxena and S. Paul. Labelled epie: A dataset for idiom sense disambiguation. In *International Conference on Text, Speech, and Dialogue*, pages 210–221. Springer, 2021.

[37] C. Sporleder, L. Li, P. Gorinski, and X. Koch. Idioms in context: The idix corpus. In *LREC*. Citeseer, 2010.

[38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[39] M. Volk and N. Weber. The automatic translation of idioms. machine translation vs. translation memory systems. *Sprachwissenschaft, Computerlinguistik und neue Medien*, (1):167–192, 1998.

[40] D. Wible and N.-L. Tsao. Stringnet as a computational resource for discovering and investigating linguistic constructions. In *Proceedings of the NAACL HLT workshop on extracting and using constructions in computational linguistics*, pages 25–31. Association for Computational Linguistics, 2010.

[41] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

[42] T. Zesch, C. Müller, and I. Gurevych. Extracting lexical semantic knowledge from wikipedia and wiktionary. In *LREC*, volume 8, pages 1646–1652, 2008.