MIss RoBERTa WiLDe: Metaphor Identification Using Masked Language Model with Wiktionary Lexical Definitions

Mateusz Babieno 1,*, Masashi Takeshita 2, Dusan Radisavljevic 2, Rafal Rzepka 2 and Kenji Araki 2

1 Graduate School of Information Science and Technology, Hokkaido University, Sapporo 060-0814, Japan
2 Faculty of Information Science and Technology, Hokkaido University, Sapporo 060-0814, Japan; takeshita.masashi@ist.hokudai.ac.jp (M.T.); radisavljevic@ist.hokudai.ac.jp (D.R.); rzepka@ist.hokudai.ac.jp (R.R.); araki@ist.hokudai.ac.jp (K.A.)
* Correspondence: mbabieno@sigmail.media.eng.hokudai.ac.jp; Tel.: +81-11-706-7389

Abstract: Recent years have brought an unprecedented and rapid development in the field of Natural Language Processing. To a large degree this is due to the emergence of modern language models like GPT-3 (Generative Pre-trained Transformer 3), XLNet, and BERT (Bidirectional Encoder Representations from Transformers), which are pre-trained on a large amount of unlabeled data. These powerful models can be further used in the tasks that have traditionally been suffering from a lack of material that could be used for training. Metaphor identification task, which is aimed at automatic recognition of figurative language, is one of such tasks. The metaphorical use of words can be detected by comparing their contextual and basic meanings. In this work, we deliver the evidence that fully automatically collected dictionary definitions can be used as the optimal medium for retrieving the non-figurative word senses, which consequently may help improve the performance of the algorithms used in metaphor detection task. As the source of the lexical information, we use the openly available Wiktionary. Our method can be applied without changes to any other dataset designed for token-level metaphor detection given it is binary labeled. In the set of experiments, our proposed method (MIss RoBERTa WiLDe) outperforms or performs similarly well as the competing models on several datasets commonly chosen in the research on metaphor processing.

Keywords: metaphor detection; figurative language; lexical definitions; Wiktionary; language models; RoBERTa; Sentence-BERT

1. Introduction
1.1. Motivation

Figures of speech in general and metaphor in particular allow us to speak more concisely, amusingly, and evocatively than with only literal language. Besides, such an endeavor would be destined to fail in the first place. This is because the figurative use of words is so ubiquitous in our language that it is very likely to be encountered in any randomly selected passage of the newspaper [1]. Metaphors are widely used in politics [2–4], psychotherapy [5,6], marketing [7], journalism [8,9], and other domains deeming persuasion highly valuable. Unfortunately, metaphorical language remains difficult to process for computers despite the significant progress that has taken place in the field of NLP (Natural Language Processing) over the last few years. This fact alone is a considerable reason to work on improving already existing algorithms as well as creating new ones designed to overcome this issue.

Machine translation is one of the NLP’s subfields that still struggles with handling metaphorical expressions. Consider the following example of English-to-Japanese translation:

1. English: Yuki is so sweet;
2. Japanese: Yuki wa totemo amai.
While the input phrase sounds perfectly fine in English, its output translation is seen as awkward by Japanese native speakers. The adjective *sweet* can be, and often is, used figuratively in English in the sense of ‘kind, gentle, or nice to other people’, but that is not the case in Japanese. The adjective *amai* is often used metaphorically as a noun modifier, but if used to describe a person’s personality traits, it conveys a very different meaning, specifically ‘lenient, forgiving’. By analyzing the above example, we can notice that the input sentence is translated in its literal sense. The algorithm is seemingly not aware of the fact it has encountered a metaphor. As a result of this, it is taking *sweet* as a word belonging to the semantic field of *TASTE* rather than that of *PERSONALITY FEATURES*. As this is the output of the currently available version of Google Translate’s engine (https://translate.google.com/?sl=en&tl=ja&text=Yuki%20is%20so%20sweet&op=translate; last accessed on 21 December 2021), it should be clear that there is still much to do when it comes to improving the performance of the algorithms related to natural language understanding.

1.2. Task Description

A word-level metaphor detection task can be defined as a supervised binary classification problem, where a computational model predicts if the target word comprised by the input sentence is used figuratively or not. The model is trained on datasets where each sample is composed of a number of features $X$ and label $y$. Input features include a target-word, sentence, and—in our case—the definition of the target word. The label is determined by the human annotator and takes values of either 0 or 1 (non-metaphor or metaphor, respectively). The model’s goal is to learn the correlations between features and labels in the training set in order to correctly predict whether the given target word is used figuratively or not in the unseen input sentence from the test set.

1.3. Contribution

In this work, we present three variants of the model built upon MelBERT (Metaphor-aware late interaction over BERT), the model presented by Choi et al. [10]. We found their work inspiring, not only because MelBERT outperforms current state-of-the-art in many cases, but because their approach is well grounded in linguistics. While acknowledging the high quality of their work, we anticipated that applying some changes to their ideas would be appropriate from a theoretical perspective and could be beneficial to the algorithm’s overall performance, making it more consistent conceptually. Specifically, we argued that introducing the lexical definition of the target word in place of the target word itself is better-suited for finding the word’s refined literal sense. We also estimated that using a different kind of sentence embedding representation should allow for achieving even better scores. The results we present in this paper mostly confirm our intuitions.

1.4. Terminology

In what follows, we use the terms *figurative* and *metaphorical* interchangeably. Strictly speaking, what is being used *figuratively* does not necessarily need to be used *metaphorically*, as the term *figurative* seems to point to a much broader concept covering also different figures of speech and tropes such as personification, allegory, and so on. Nonetheless, the terms *metaphorical* and *figurative* are often used to convey the same meaning. It can be witnessed by examining the definition of *figurative* found in the Oxford Dictionary of English [11] described therein as ‘departing from a literal use of words; metaphorical’. On the other hand, *literal* is defined as ‘taking words in their usual or most basic sense without metaphor or exaggeration’. Although we share the lexicographer’s intuition that it is the word’s usage rather than the word itself that should be called *metaphorical or literal*, in this paper we consider a *metaphor* any linguistic unit whose meaning as identified in a given utterance diverges from its most basic sense. As such, we remain in harmony with the Metaphor Identification Procedure [12] described in more details in Section 2.2.1.
2. Related Work

2.1. Research in Natural Language Processing

Metaphor detection using deep learning has gained much popularity over the last few years, as shown in the related survey [13]. To some degree, this growth in interest can be attributed to the emergence of language models like BERT (Bidirectional Encoder Representations from Transformers) [14] achieving state-of-the-art performance irrespective of the NLP task they are being used for. This tendency can be noticed by looking at the list of models participating in the second Metaphor Detection Shared Task [15], where almost all of them use the implementations of ELMo (Embeddings from Language Model) [16], BERT, or some of its derivatives, such as RoBERTa (Robustly Optimized BERT Pretraining Approach) [17] or ALBERT (A Lite BERT) [18].

Presumably, this constatation led Neidlein et al. to publish their analysis [19] of recent metaphor recognition systems based on the language models. The authors argue that although the new models yield very satisfactory results, their design often shows considerable gaps from a linguistic perspective, indicated by the fact that they perform substantially worse on unconventional metaphors than on conventional ones. Subsequently, they present another finding that should be of great value to the whole community. First, the reader should know that VUAMC (Vrije Universiteit Amsterdam Metaphor Corpus) [20] is the corpus underlying the two most frequently used datasets in metaphor detection research, specifically VUA-ALL-POS (All Parts of Speech) and VUA-SEQ (Sequential). Neidlein et al. reveal in their paper [19] that in recent research, some authors compare their results achieved using VUA-SEQ to the results gained on VUA-ALL-POS. As they point out, the underlying corpus remains the same, but the VUA-SEQ is substantially easier to do well on than VUA-ALL-POS, and thus such comparisons are inherently unfair. Later in the paper, the authors present results for both VUA-SEQ and VUA-ALL-POS using a number of models published by other researchers and their own model as well. Implementation of their method achieves an F1-score of 77.5% on VUA-SEQ and only 69.7% on VUA-ALL-POS, which indeed proves that conflating these two cannot be considered good practice.

DeepMet [21] is the winner of the second Metaphor Detection Shared Task mentioned before. It managed to outperform all the other models on any given data subset, often by a large margin. DeepMet uses RoBERTa as its structural foundation, and siamese architecture with two Transformer [22] encoder layers to process different features. The authors reformulate a metaphor detection task from a classification or sequence labeling problem to a reading comprehension task. DeepMet utilizes 5 categories of input features: global text context, local text context, query word, general POS (Parts of Speech), and fine-grained POS generated using SpaCy (https://spacy.io/; last accessed on 21 December 2021). The overall performance is boosted by using ensemble learning and metaphor preference parameter $\alpha$, helping the model to achieve a better recall score. This parameter is introduced due to the fact that the metaphor datasets are highly unbalanced, meaning they comprise many more target words belonging to the non-metaphorical class.

Similar to our approach, in the work of Wan et al. [23], dictionary definitions are also used to improve the performance of the proposed BERT-based model. It is noteworthy that the authors perform not only metaphor detection but metaphor interpretation as well. In order to do so, they utilize every definition of the given word that is available in the Merriam Webster dictionary (https://www.merriam-webster.com/dictionary/; last accessed on 21 December 2021). Using attention, they try to select the one that is semantically closest to the target word’s contextual meaning. Afterwards, they concatenate all of the definitions’ representations with the contextualized representation of the target word. Authors test their method on 3 datasets: VUA-SEQ, TroFi (Trope Finder) [24], and PSU CMC (PSU Chinese Metaphor Corpus) [25]. They use BERT for the experiments on TroFi and VUA-SEQ, and the Chinese BERT for PSU CMC.

Although both ours and Wan et al.’s models use lexical information as one of the input features, there are several dissimilarities that make our approaches fundamentally different. First, while we collect all of the definitions in a simple and completely automatic manner,
Wan et al. recruit a number of annotators to improve a part of their datasets. Even though both the authors and ourselves use dictionaries as the source of the word descriptions, the reason we do it and the goals we are trying to achieve are very different. While we are trying to extract the target word’s meaning in its most basic literal sense regardless of the surrounding context, Wan et al. search specifically for the definition semantically closest to the contextual meaning. Another dissimilarity lays in the choice of dictionary. While Wan et al. choose Merriam-Webster, we are using the openly available Wiktionary (https://en.wiktionary.org/wiki; last accessed on 21 December 2021), which also grants us with a larger number of definitions to utilize. As of November 2021, Wiktionary included 809,475 gloss definitions and 1,300,634 definitions in total just for English (the latter includes the number of definitions for inflections, variants, and alternative spellings as well, cf. https://en.wiktionary.org/wiki/Wiktionary:Statistics; accessed on 21 December 2021). We could not find the precise information on how many definitions there are in the version of Merriam-Webster used by Wan et al., but the explanation found on the dictionary’s website could hint at being somewhere over 275,000 (this is the approximate number of word choices available in Merriam-Webster’s Collegiate Thesaurus API; Merriam-Webster’s Collegiate Dictionary API features more than 225,000 definitions, cf. https://dictionaryapi.com/products/index; last accessed on 21 December 2021).

The work that has become the main inspiration for our project is that from Choi et al. [10]. The authors present MelBERT (Metaphor-aware late interaction over BERT), the model for metaphor detection using RoBERTa as its architectural foundation. MelBERT’s design allows for using the principles of MIP (Metaphor Identification Procedure) [12] simultaneously with the concept of SPV (Selectional Preference Violation) [26], both of which we describe in detail in the following subsection. MIP was proposed by the Pragglejaz Group and provides the reader with instructions on how to establish whether a word is being used figuratively or not in a given context. The concept of Selectional Preference Violation in relation to metaphor was brought to the attention of computational linguistics by Wilks and it can be said to focus on the degree of semantic compatibility between senses of given lexical units. Utilizing both strategies together, while using additional features such as POS tags, as well as a local and global context, Choi et al. conduct a set of experiments on multiple datasets recognized in the field of metaphor detection: MOH-X (Mohammad et al. [2016] dataset) [27], TroFi, and two datasets based on VUAMC (for details on the datasets cf. Section 3.2). Subsequently, their results are compared to those achieved by some of the strongest benchmarks, including Su et al.’s DeepMet [21], which was briefly introduced above. While MelBERT is not the first model following the guidelines of MIP or SPV in metaphor detection, we found the fact of complementing linguistic theory with the power of recently published bidirectional (or non-directional, as this would arguably be a more appropriate description, cf. [28]) language models appealing and decided to further build on MelBERT’s authors’ ideas. This sort of holistic approach seems to also address the issue of having too much focus on technical innovations while disregarding related linguistic theories by the authors of recent models designed for metaphor identification, as signalized by Neidlein et al. [19].

2.2. Metaphor Detection Procedures

In this section we present an overview of the two linguistic procedures for metaphor identification, which provide theoretical justification for the algorithmic architecture we adopt in our models as described in Section 3. Both of them have been successfully used in metaphor detection tasks over the years, becoming a popular choice among researchers in the field (cf. [10,29–36]).

2.2.1. MIP

MIP (Metaphor Identification Procedure) was introduced by Pragglejaz Group [12] and was designed as a method for identifying words used figuratively in discourse. As pointed out by the authors, the primary difficulty with metaphor detection is that researchers often
differ in their opinions on whether a given word is used figuratively or not because of a lack of objective and universal criteria that could be applied to the task. Before MIP was proposed in 2007, the need for undertaking this problem had been signalized by other authors as well. For example, Heywood et al. ([37], p. 36) suggested that:

The fuzzy boundary between literal and metaphorical language can only be properly tackled by being maximally explicit as to the criteria for classifying individual expressions in one way or another.

Similarly, in their work dedicated to the analysis of metaphor in a specialized corpus, Semino et al. ([38], p. 1272) admitted that:

It seems to us that we still lack explicit and rigorous procedures for its identification and analysis, especially when one looks at authentic conversational data rather than decontextualized sentences or made-up examples.

Such concerns have become a motivation for creating a few-step procedure for metaphor identification that would be precise enough not to allow for much idiosyncrasy in the related decision making. In MIP, these steps include reading the whole text to understand its meaning, establishing the contextual meaning of each lexical unit in the text, determining whether a given lexical unit has more basic sense in other contexts, and, finally, deciding if the contextual meaning contrasts with the basic meaning and can be understood by comparison with it. Since the explanation regarding the way of determining a word’s basic meaning is of great importance to the current task, and paraphrasing inevitably leads to some information loss, we allow ourselves to directly cite the authors on this issue ([12], p. 4):

For each lexical unit, determine if it has a more basic contemporary meaning in other contexts than the one in the given context. For our purposes, basic meanings tend to be:

- More concrete; what they evoke is easier to imagine, see, hear, feel, smell, and taste.
- Related to bodily action.
- More precise (as opposed to vague).
- Historically older.

Basic meanings are not necessarily the most frequent meanings of the lexical unit.

Although this list might seem self-explanatory at first sight, in reality, establishing whether given meaning is basic often comes as no easy task. For example, there are cases in which a polysemous word has multiple senses from which one is historically older and the other is more concrete. In such a case, it is not clear which of the meanings should be considered the basic one. Consider the example sentence provided by the authors ([12], p. 4; emphasis added):

For years, Sonia Gandhi has struggled to convince Indians that she is fit to wear the mantle of the political dynasty into which she married, let alone to become premier.

Here fit appears to be this kind of a word. As the authors write in their explication ([12], p. 8):

The adjective fit has a different meaning to do with being healthy and physically strong, as in Running around after the children keeps me fit. We note that the “suitability” meaning is historically older than the “healthy” meaning; the Shorter Oxford English Dictionary on Historical Principles (SOEDHP) gives the “suitability” meaning as from medieval English and used in Shakespeare, whereas the earliest record of the sport meaning is 1869. However, we decided that the “healthy” meaning can be considered as more basic (using the description of “basic” set out earlier) because it refers to what is directly physically experienced.
This description suggests that establishing the basic meaning of a word, and thus deciding whether it is being used figuratively or not, still involves a certain amount of subjectivity. This problem is addressed by Steen et al. in [39], where the authors suggest that the term metaphor should be in fact thought of as short for “metaphorical to some language user” ([39], p. 771) as opposed to absolutely metaphorical.

Another issue regarding using the original MIP guidelines in our work is posed by the fact that while we are dealing with token-level metaphor detection, MIP recognizes multi-word expressions and treats them, not their component tokens, as lexical units. This can be illustrated with *let alone* from the example sentence above, which in the datasets based on VUAMC is treated as two separate tokens with two separate labels. Another question worth asking with regard to this example is whether the idiomatic sense of *let alone*, identified in the example sentence should not be treated as secondary to a sense of ‘stop bothering’ (e.g., *will you finally let me alone*?).

2.2.2. SPV

Utilizing the concept of SPV (Selectional Preference Violation) as a tool in automatic metaphor recognition has been postulated most notably by Wilks [1,26,33]. Wilks argued that metaphors could be detected in a procedural manner by determining whether semantic preferences of linguistic units present in the sentence are not violated. To cite the author, such preference violation “can be caused either by some ‘total’ mismatch of word-senses (…) or by some metaphorical relation” ([26], p. 182). Such metaphorical relation may be illustrated with the famous Wilksian example: *My car drinks gasoline*, where the verb *drink* can be said to exhibit preference for an animate agent and a patient belonging to the semantic field of LIQUIDS. As it denotes an non-animate object, the noun *car* breaks the verb’s selectional preference. Shutova et al. contrast this example with: “My aunt always drinks her tea on the terrace” ([40], p. 310) in which selectional preference violation does not occur.

Metaphorical expressions breaking selectional preferences can be easily found in everyday language. Consider the sentence: “This new PlayStation is a beast”. Since *PlayStation* is referring to a video game console, *beast*, defined as ‘any animal other than a human’, violates the subject’s preference for an object belonging to the semantic field of MACHINES, which hints at the possibility of a metaphor being in use. In our work, similar to Choi et al. [10], we employ a rather simplified interpretation of the SPV concept. Specifically, as it is likely that it breaks some selectional preferences on the way, whenever some unexpected, unusual word occurs in a given context, we assume that it might be used figuratively. This is often the case and can be proven by the example above; if there was a rule that words can be used only in their literal senses, *device*, *console*, *machine*, or alike would be used in place of *beast*. Some other examples portraying this rule would be: “She took his life”, where *backpack*, *wallet*, *sandwich*, etc. would be expected in place of *life*; “I smell victory”, where *tomato soup*, *cigarettes*, or *gasoline* would have preference over *victory*; “They have been living in a bubble” where *house*, *mansion*, etc. would be used in place of *bubble*.

3. Materials and Methods

3.1. Model Structure

In this chapter, we present the architecture of MIss RoBERTa WiLDe, a model for Metaphor Identification using the RoBERTa language model. At its core, MIss WiLDe utilizes MelBERT (Metaphor-aware late interaction over BERT) published recently by Choi et al. [10] and therefore the architecture of the two models is almost identical. For the model overview, whose design was inspired by the aforementioned work, see Figure 1. Conceptually, MIss WiLDe and MelBERT take advantage of the same linguistic methods for metaphor detection, namely SPV (Selectional Preference Violation) and MIP (Metaphor Identification Procedure). While the implementation of the former in our model remains
mostly unchanged, the latter is affected by a different kind of input, which is the first novelty of our approach.

**Figure 1.** Model overview. Depending on the sub-model, either RoBERTa or Sentence-BERT encoder is utilized. One of the sub-models uses cosine similarity as the additional tool for measuring the semantic gap between input sequences. Elements marked with colors other than black and white signify introduced innovations in relation to MelBERT [10]. Gradients stand for partial novelty.

In order to determine whether the target word is used figuratively in the given context, Choi et al. utilize its isolated counterpart as a part of the input. This is done for the same purpose Miss WiLDe takes advantage of the target word’s definition (see Figure 2 and 3). Although using uncontextualized embedding of the target word proved to yield satisfactory results in the work of Choi et al., from a theoretical perspective, this approach does not seem to be absolutely free of flaws. Its main issue lies in the fact that during the pre-training stage, the word embedding is constructed by looking at the word’s usages in various contexts. In consequence, at least some of these usages (depending on the word in question, even most of them) are already metaphorically motivated. On the other hand, using the target word’s lexical definition—more specifically, using the first of the definitions listed in the dictionary—should be able to bypass this problem. This is because lexicographers tend to place the definition representing what is called the word’s basic meaning at the top of the definitions list. Given the definition is indeed representing the word’s basic sense, its embedding representation can be subsequently used to compare it with the contextualized embedding of the target word. If the gap between them is big enough, it can be estimated that the word is used figuratively.

Another motivation for using the definition rather than the target word itself is that—considering BERT was pre-trained on a large amount of text data with a sentence as its input unit—we anticipate that using sentences instead of single words might lead to some performance gains.

The second novelty of our method lies in the way in which the embedding representation of the sentence is constructed. In the work of Choi et al., sentence representation is calculated using the [CLS] special token. However, it has been experimentally established by Reimers and Gurevych in their work on Sentence-BERT [41] that this approach falls short in comparison to using the mean value of tokens without the [CLS] and [SEP] tokens. It was also further confirmed by the results we achieved in a number of experimental trials with and without the use of the aforementioned special tokens.
Figure 2. Input layer using RoBERTa. As single words are often divided into multiple tokens, most of the times the number of input words does not equal the number of output tokens. Context information is used to determine all the tokens that correspond to the target word.

Figure 3. Input layer using Sentence-BERT. In one of the sub-models, MiSS WiLDe takes advantage of Sentence-BERT instead of RoBERTa.

We present 3 variants of the MiSS WiLDe model, which we are going to interchangeably call the sub-models. These are:

- **MiSS WiLDe_base.** This is the core version of our model. See Figure 1 for the model overview and Figure 2 for its input layer using RoBERTa;
- **MiSS WiLDe_cos.** Both SPV and MIP are methods of using semantic gaps to determine if a target word is used metaphorically. Therefore, we also created a sub-model using cosine similarity to explicitly handle semantic gaps. This is shown by CS in Figure 1.
Specifically, similarity between the meaning of the sentence and the meaning of the target word is calculated within the SPV block, while similarity between the meaning of the target word’s definition and the meaning of the target word itself is calculated within MIP. The input layer for this sub-model is common with the base variant visualized in Figure 2:

- **Miss WiLDe_sbert.** Since the results published in [41] suggest that using Sentence-BERT should result in acquiring sentence embeddings of better quality than those produced by both [CLS] tokens and averaged token vectors, we have decided to confirm it experimentally. We have therefore replaced RoBERTa with Sentence-BERT as an encoder in one of our 3 sub-models. The input layer using Sentence-BERT is depicted in Figure 3.

The input to our model consists of the sentence comprising the target word on one side, and the definition of this target word on the other (depending on the part of speech and the availability of the definition in Wiktionary, lemma can be used instead of the definition; cf. Section 3.3 for the details). The conversion of words into tokens is then performed using the improved implementation of Byte-Pair Encoding (BPE), as proposed by Radford et al. in [42] and used by Choi et al. in [10] as well. This can be described as follows:

- \( \text{TOK}(w_1, w_2, \ldots, w_m) = t_1, t_2, \ldots, t_{p-1}, t_p; \)
- \( \text{TOK}(d_t w_1, d_t w_2, \ldots, d_t w_n) = d_t l_1, d_t l_2, \ldots, d_t q_{p-1}, d_t q_p; \)

where **TOK** stands for the tokenizer, \( w \) represents a single word within the analyzed sentence, with \( t_w \) being the target word or, to put it differently, a metaphor candidate; \( m \) in the subscript is the number of words in the input sentence. \( t \) represents an output token, while \( p \) is the number of the output tokens. Depending on a given target word \( t_w \), \( t_w \) should be considered an abbreviation for \( t_{w_1}, t_{w_2}, \ldots, t_{w_{z-1}}, t_{w_z} \), where \( z \) stands for the number of tokens the target word was split into. This can be observed in Figure 2 as well. In the formulas presented in this section, we use the abbreviated forms for simplicity (a single input word is often transformed into multiple tokens: for more details on Byte-Pair Encoding cf. https://huggingface.co/docs/transformers/tokenizer_summary; last accessed on 21 December 2021). In the second formula, \( d_t w \) stands for a component word of the target word’s definition and \( d_t \) for an output token; \( n \) and \( q \) in the subscripts denote the number of words in the definition and the number of related output tokens, respectively.

Afterwards, tokens are transformed into embedding vectors via the encoder layer. Input and output of our two encoders can be illustrated with the following formulas:

- \( \text{ENC}(t_{l_1}, t_{l_2}, \ldots, t_{l_{p-1}}, t_p) = t_{e_1}, t_{e_2}, \ldots, t_{e_{p-1}}, t_{e_p}; \)
- \( \text{ENC}(d_{l_1}, d_{l_2}, \ldots, d_{q_{p-1}}, d_{q_p}) = d_{t e_1}, d_{t e_2}, \ldots, d_{t e_{q-1}}, d_{t e_q}; \)

where **ENC** stands for the function producing contextualized vector representation for a given input, \( t \) represents a single token within the analyzed sentence, with \( t_w \) in the subscript denoting the target word. \( t_e \) is the vector embedding representation that corresponds to the input token with the same index, while \( t_{e_{t_w}} \) is the embedding representation of the target word’s tokens. Analogously, \( d_t \) stands for a token coming from the definition of the target word and \( d_{t e} \) for a vector embedding of the said token. Additionally, \( p \) and \( q \) denote the length of the sentence and the length of the target word’s definition measured in the number of tokens, respectively.

Subsequently, the mean value of the sentence’s token vectors on one side and the mean value of the definition’s token vectors on the other are computed within the pooling layer. To the output of this layer, dropout is subsequently applied. On both sides, an output vector is then concatenated with the vector representation of the target word’s tokens having undergone the same operations. In case of the cosine-similarity sub-model, an additional third vector representing similarity between the two respective vectors is also concatenated. In order to calculate the gap between these vectors, a multilayer perceptron is applied on the output of the concatenation function. The formulas for the hidden vectors obtained this way in SPV and MIP layers are presented below.
would improve the algorithm’s performance and in order to do so we have adopted the similarity vector becomes the cosine similarity value measured between the average sentence vector and the target word vector; analogously similarity\_mip represents the cosine similarity between the average definition vector and the target word vector. The \( \| \cdot \|_2 \) denotes the Euclidean norm, the \( \cdot \) stands for the dot product between the vectors, and \( \epsilon \) is a parameter of small value used to avoid division by zero (https://pytorch.org/docs/stable/generated/torch.nn.CosineSimilarity.html; last accessed on 21 December 2021). The output of the model is calculated by adding bias to the concatenation function that takes in the two hidden vectors and applies the log-softmax activation function onto the result. Finally, the candidate with the higher probability score is chosen as the predicted label. The process can be represented with the following formula:

\[
\begin{align*}
y_t &= \log(\sigma(W^\top \Phi(hv\_spv, hv\_mip) + b)); \\
\hat{y} &= \text{argmax}(y_t)
\end{align*}
\]

where \( \hat{y} \) is the label predicted by the model, such that \( \hat{y} \in \{0, 1\} \). This prediction is the result of the argmax operation applied on \( y_t \), which in turn stands for the natural logarithm of the value output by the softmax function denoted with \( \sigma \). Softmax outputs two values that are the probabilities for each class (literal and metaphor) that range from 0 to 1 and sum up to 1. \( W^\top \) denotes the weights matrix, \( \Phi \) stands for concatenation, and \( b \) signifies bias.

We use the negative log-likelihood loss function, which in combination with log-softmax activation, acts essentially the same as cross-entropy combined with softmax, but has improved numerical stability in PyTorch [43].

Visualization of the model is provided in Figure 1. Two variants of its input layer can be compared as shown in Figure 2 and 3. The code allowing for training the three sub-models and reproducing our results can be found at: https://github.com/languagemedialaboratory/ms_wilde; last accessed on 21 December 2021).

3.2. Datasets

In this section, we present the datasets used in the experiments. We wanted to confirm the validity of our hypothesis that utilizing lexical definitions of target words would improve the algorithm’s performance and in order to do so we have adopted the same datasets as in the work of Choi et al. [10]. The original data is available at the authors’ drive, a link to which can be found on their GitHub (https://drive.google.com/file/d/1738aqFObjfcoG2OiFknrElmUHu1HqQvRz/view?usp=sharing via https://github.com/jin530/MelBERT; last accessed on 21 December 2021). The downloadable repository consists of MOH-X, TroFi, VUA-20 (variant of VUA-ALL-POS known from Metaphor Detection Shared Task [15,44]), VUA-18 (variant of VUA-SEQ known from Gao et al. [35] and Mao et al. [36]), VUA-VERB, 8 subsets of VUA-18, 4 of which are selected based on
the POS tags of the target words (nouns, verbs, adjectives, and adverbs), and another 4 on the genre to which the sentence comprising target word belongs (academic, conversation, fiction, and news). Both Genres and POS are used only for testing. The same datasets enriched with the Wiktionary definitions can be downloaded directly from our GitHub (https://github.com/languagemedialaboratory/ms_wilde/tree/main/data; last accessed on 21 December 2021).

MOH-X (Mohammad et al. [2016] dataset) [27] and TroFi (Trope Finder) [24] are relatively small datasets annotated only for verbs. MOH-X is built with the example sentences taken from WordNet [45], and TroFi with the ones from the Wall Street Journal [46]. For the sake of fair comparison with Choi et al., we use these two datasets only as the test sets for the models trained beforehand on the VUA-20, in the same way as performed by the authors of MelBERT. As they note in the paper, this can be viewed as the zero-shot transfer learning. VUAMC (Vrije Universiteit Amsterdam Metaphor Corpus [20,47], http://www.vismet.org/metcor/documentation/home.html; last accessed on 21 December 2021) is the biggest publicly available corpus annotated for token-level metaphor detection and it is seemingly the most popular one in the field. It comprises text fragments sampled from the British National Corpus (http://www.natcorp.ox.ac.uk/; last accessed on 21 December 2021). Sentences it contains were labeled in accordance with MIPVU (Metaphor Identification Procedure VU University Amsterdam), the refined and adjusted version of the already described MIP (Metaphor Identification Precedure). Both VUA-ALL-POS (All-Part-Of-Speech) and VUA-SEQ (Sequential) are based on VUAMC, which has been used in the Metaphor Detection Shared Task, first in 2018 and later in 2020 [15,44]. The repository prepared for the Metaphor Detection Shared Task is provided under the following URL: https://github.com/EducationalTestingService/metaphor/tree/master/VUA-shared-task1 (last accessed on 21 December 2021). Inside, we can find links allowing for downloading:

- VUAMC corpus in XML format;
- Starter kits for obtaining training and testing splits of VUAMC corpus (vuamc_corpus_train, vuamc_corpus_test);
- Lists of ids (all_pos_tokens, all_pos_tokens_test, verb_tokens, verb_tokens_test) specifying the tokens from VUAMC to be used as targets for classification in the two tracks of the Metaphor Detection Shared Task: All-Part-Of-Speech and Verbs.

In the 12,122 sentences comprised by vuamc_corpus_train, all of their component words are labeled for metaphoricity, irrespective of the part of speech they belong to. For example, in the input sentence “The villagers seemed unimpressed, but were M_given no choice M_in the matter .”, there are altogether 14 tokens, including punctuation marks. Two of the tokens are labeled as metaphors (the verb given and the preposition in), and the remaining ten as non-metaphors. This is indicated by the prefix “M” attached to the metaphors. In all_pos_tokens, only six out of these 14 tokens are chosen as targets for classification. These tokens are: villagers, seemed, unimpressed, given, choice, and matter. In this work, after Neidlein et al. [19], the name VUA-ALL-POS refers to the dataset utilizing only the target words specified by all_pos_tokens and all_pos_tokens_test. The dataset called VUA-20, which we adopt from Choi et al. [10] and which we use in the experiments, comprises the same testing data, yet it produces more training samples from vuamc_corpus_train than specified by all_pos_tokens. In the example sentence above, VUA-20 uses all of the available tokens, excluding punctuation, as targets for classification. VUA-20 takes advantage of both content words (belonging to verbs, nouns, adjectives and adverbs) and function words (members of remaining parts of speech), while VUA-ALL-POS is said to limit itself to the content words only (excluding verbs have, do and be). This difference results in a much bigger number of target tokens available in the former’s training set (160,154 and 72,611 for VUA-20 and VUA-ALL-POS, respectively). At the same time, for reasons unknown, VUA-20 lacks 86 of the target tokens used in VUA-ALL-POS. With this exception, VUA-20 can be therefore viewed as an extended variant of VUA-ALL-POS. VUA-20 includes all of the sentences utilized by VUA-ALL-POS plus those excluded from the
latter due to the POS-related restrictions. As a result, while there are 12,122 unique sentences provided in total by vuamc_corpus_train, the numbers of sentences used for training in VUA-20 and VUA-ALL-POS are 12,093 and 10,894, respectively. The 29 “sentences,” which VUA-20 is lacking with respect to vuamc_corpus_train, were excluded, presumably because they are either empty strings or single punctuation characters (”, ”, ”!” and ”?”). As mentioned, testing data is common for both datasets: they comprise the same 22,196 target tokens coming from 3698 sentences selected from 4080 available in total in vuamc_corpus_test.

VUA-SEQ (we use this name after Neidlein et al. [19]) is another dataset built upon VUAMC. It was used in the works of Gao et al. [35] and Mao et al. [36], among the others. It differs from VUA-ALL-POS in that it employs different splits of VUAMC and in that it uses all of the tokens available in a sentence as targets for classification (including punctuation marks). This results in a much bigger number of target tokens used by VUA-SEQ in comparison with VUA-ALL-POS (205,425 and 94,807, respectively). However, VUA-SEQ uses a smaller number of unique sentences than VUA-ALL-POS (10,567 and 14,974, respectively). Unlike VUA-ALL-POS, VUA-SEQ has a development set as well. VUA-18, which we adopt from Choi et al. [10], is very similar to VUA-SEQ, as it uses the same sentences in each of the subsets (6323, 1550, and 2694 sentences for training, development, and testing sets, respectively). What does not allow for calling the two datasets identical is that VUA-18 does not count contractions and punctuation marks as separate tokens (there is a very small number of exceptions to this general rule). For example, the sentence coded in VUAMC as: “M_Lot of M_things daddy has n’t seen .” is divided into 8 tokens in VUA-SEQ, whereas in VUA-18 it is presented as “Lot of things daddy hasn’t seen.”, which results in using only 6 tokens and 6 corresponding labels (without “n’t” and “.”). In consequence, the numbers of tokens are: 116,622 and 101,975 in the training sets, 38,628 and 34,253 in the development sets, 50,175 and 43,947 in the testing sets of VUA-SEQ and VUA-18, respectively. To stay in uniformity with Choi et al., we are not utilizing VUA-18’s development set in the experimental trials.

VUA-VERB, which we adopt from Choi et al. [10], utilizes the same sentences as those selected in the lists prepared for the Metaphor Detection Shared Task (verb_tokens and verb_tokens_test), although it splits the original training data into training and validation subsets. While in verb_tokens there are 17,240 target tokens used for training, in VUA-VERB there are 15,516 and 1724 tokens comprised by its training and development sets, respectively. The number of tokens used for testing equals 5873, which is the same in both cases. In the experimental trials, we are not taking advantage of VUA-VERB’s development set.

Although Neidlein et al. [19] claim that it is only the content words (verbs, nouns, adjectives, and adverbs), whose labels are being predicted in VUA-ALL-POS, this is not entirely accurate. There are instances of interjections (“Ah, yeah”), prepositions (“like” in “It would be interesting to know what he thought children were like”), conjunctions (“either” in “Criminal behaviour is either inherited or a consequence of unique, individual experiences”), etc.

While in their paper Su et al. formulate the opinion that “POS such as punctuation, prepositions, and conjunctions are unlikely to trigger metaphors” ([21], p. 32), at the same time they provide the evidence for the opposite, at least with regard to the prepositions: from Figure 5 they attach, it is clear that the adpositions are annotated as being used metaphorically more often than any other part of speech in VUAMC ([21], p. 35). This should come as no surprise, as, for example, there are entire tomes devoted to the analysis of the primarily spatial senses of temporal prepositions and related metaphorical meaning extensions (cf. [48–50]).

3.3. Data Preprocessing

For data preprocessing, we use Python equipped with WordNetLemmatizer (https://www.nltk.org/_modules/nltk/stem/wordnet.html; last accessed on 21 Decem-
ber 2021) for obtaining the dictionary forms of given words and Wiktionary Parser (https://github.com/Suyash458/WiktionaryParser; last accessed on 21 December 2021) for retrieving their definitions. As the outputs are different for different parts of speech, we take advantage of the POS tags already included in the datasets. These, however, have to be first mapped to the format used by Wiktionary. It is noteworthy that not all of these POS tags are accurate, which sometimes prevents the algorithm from retrieving the appropriate definition.

In Delahunty and Garvey’s book [51], nouns, verbs, adjectives, and adverbs are termed as the major parts of speech, and as such are contrasted with the minor parts of speech (all the others). Alternatively, they can be called the content words and the function words, respectively (cf. Haspelmath’s analysis [52]). The former have more specific or detailed semantic content, while the latter have a more non-conceptual meaning and fulfill an essentially grammatical function [53]. As a result of this, in general, definitions of the function words do not add much of semantic information that could be used in metaphor detection. On the contrary, using averaged vectors of their component tokens could become a source of unnecessary noise, leading to performance decay (consider the first definition of the very frequently occurring determiner the found in Wiktionary: “Definite grammatical article that implies necessarily that an entity it articulates is presupposed; something already mentioned, or completely specified later in that same sentence, or assumed already completely specified”). When it comes to function words, we therefore decided to use their lemmas instead of definitions. In this regard, we have made an exception for 12 prepositions, namely: above, below, between, down, in, into, on, over, out, through, under, and up. These are all very frequent, as they all constitute a part of the stopwords from NLTK’s corpus package (https://www.nltk.org/_modules/nltk/corpus.html; last accessed on 21 December 2021). When present in utterances, they often manifest the underlying image schemata well known from the Conceptual Metaphor Theory first popularized by Lakoff and Johnson in [54] and further described in detail by other authors, for example by Gibbs in [55]. Admittedly, the choice of the words from the outside of the major parts of speech category is subjective and could be made differently.

We assumed that it is likely that the first definition available in the Wiktionary is going to be the one representing a word’s basic meaning and therefore we retrieve only the first of the definitions available for the given part of speech. This choice was preceded with the lecture of Wiktionary guidelines (https://en.wiktionary.org/wiki/Wiktionary:Style_guide#Definitions; last accessed on 21 December 2021), where—at least for the complex entries—it is explicitly recommended to use the logical hierarchy of the word senses, meaning: core sense at the root. The relation between basic meanings and metaphoricity within the logical hierarchy is explained in [56] (p. 285; emphasis added):

The logical ordering runs from core senses to subsenses. Core meanings or basic meanings are the meanings which are felt as the most literal or central ones. The relation between core sense and subsense may be understood in various ways, e.g., as the relation between general and specialised meaning, central and peripheral, literal and non-literal, concrete and abstract, original and derived.

Following the general strategy of using only the first of the available definitions, we make an exception for the words whose first definitions include the tags archaic and obsolete. Although, as mentioned in Section 2.2.1, the MIP (Metaphor Identification Procedure) postulates considering the historical antecedence as one of the cues in establishing a given word’s basic meaning, studying the data coming from Wiktionary led us to the conclusion that very often the definitions comprising the aforementioned labels stand for the senses that are no longer accessible to contemporary language users. For this reason, we argue that they should not be treated as the basic senses and that it is not appropriate to compare them with the contextual senses in order to decide whether a word is used figuratively. In practice, our algorithm collects the first definition of a given target word without the words
archaic and obsolete (or their derivatives) inside the brackets. For example, the definitions of the verb consist available in the Wiktionary are as follows:

1. (obsolete, copulative) To be.
2. (obsolete, intransitive) To exist.
3. (intransitive, with in) To be comprised or contained.
4. (intransitive, with of) To be composed, formed, or made up (of).

Out of these four, it is the third definition that includes neither of the tags mentioned above and thus becomes accepted by our algorithm. Furthermore, as with all other definitions we collect, the brackets along with their content become erased. The final shape of the definition adopted for the target word contain is therefore: To be comprised or contained. In the case where all the definitions of a given word include either archaic or obsolete labels, we keep the first definition from the list.

Using the algorithm illustrated above, the definitions are collected automatically and without supervision, which significantly reduces costs. Experimental results presented in the following section prove that this simple method is in fact very efficient.

4. Experiments

In this chapter, we first present the models whose results we use as the baselines for comparison in Section 5. Subsequently, we provide a brief description of the setup used throughout the experiments.

4.1. Models for Comparison

We compare the performance of Miss WiLDe’s 3 variants with 9 other models, which are the following:

• **MDGI-Joint-S** and **MDGI-Joint** (denoted as MDGI-J-S and MDGI-J). These are the two variants of the model designed by Wan et al. in [23]. The outline of the model has already been presented in Section 2.1. The first variant of the model (MDGI-Joint-S) shares parameters between the context encoder and definition encoder, while the other one uses independent encoders. Although in the paper, authors present part of their results as achieved using VUA-ALL-POS, this seems to be inaccurate (cf. the datasets available at their GitHub: https://github.com/sysulic/MDGI; last accessed on 21 December 2021), and therefore we place them in Table 2, which shows the results for VUA-SEQ/VUA-18. The results for VUA-VERB reported by the authors can be found in Table 3. As for TroFi, we use it only as the test set for the model trained on VUA-20 and thus the results reported in [23] are not comparable with ours.

• **BERT**. The model presented by Neidlein et al. [19] using the uncased base BERT model as its backbone and some standard hyperparameters. In the following tables, we present the results reported by the authors.

• **RNN_HG** and **RNN_MHCA**. The two models built upon Gao et al. [35] and presented by Mao et al. in [36]. The first model follows the guidelines of MIP (Metaphor Identification Procedure), while the other follows SPV (Selectional Preference Violation). The first model uses GloVe (Global Vectors) embedding as the representation of the target word’s literal meaning and the hidden state from the BiLSTM fed with the concatenation of GloVe and ELMo embeddings as the representation of its contextual meaning. In order to compute the contextual representation of the target word, RNN_MHCA uses multi-head contextual attention.

• **RoBERTa_BASE** (denoted as RoB_BASE). A vanilla version of the RoBERTa model designed for metaphor detection prepared as a baseline by Choi et al. Unlike MelBERT, it utilizes all-to-all interaction architecture.

• **RoBERTa_SEQ** (denoted as RoB_SEQ). To paraphrase Choi et al., this model takes one single sentence as an input, where the target word is marked as the input embedding token. The model predicts the metaphoricity of the target word using its embedding vector.
• **DeepMet.** The model designed by Su et al. [21], the winner of the Metaphor Detection Shared Task 2020. Its outline has already been presented in Section 2.1.

• **MelBERT.** Its outline has already been presented in Section 2.1 and, in comparison with our model, in Section 3.1.

• **MIss WiLDe_base, MIss WiLDe_riv1 and MIss WiLDe_riv2** (denoted as MsW_base, MsW_riv1, and MsW_riv2). These are the three variants of our model described in details in Section 3.1. As signalized by their names, the first one is a core model, the second one uses a cosine-similarity measure as an additional feature, and the third one uses the Sentence-BERT encoder in place of RoBERTa, which is utilized in the first two sub-models.

### 4.2. Experimental Setup

In order to confirm our suppositions concerning possible improvements acquired through the introduction of the innovations outlined above, the experimental setup is kept the same as in [10]. For the sake of brevity, we compare the models without using a bootstrapping aggregation technique. We use the same hyperparameters for training the models: The batch size is set to 32, the max sequence length is set to 150, and the number of epochs is set to 3. The AdamW optimizer is used with the initial learning rate of $3 \times 10^{-5}$ and the third epoch is set as the warm-up epoch. We perform 5 trials for every experiment. The results presented in the tables are calculated, taking the mean value of the scores achieved over 5 runs. To preserve reproducibility and exclude any form of cherry-picking, we use the same set of random seeds: 1, 2, 3, 4, and 5. The same GPU is also used for every experiment performed, specifically Tesla P100-PCIE-16GB, provided by Google Colab (https://colab.research.google.com/; last accessed on 21 December 2021). The code is implemented using PyTorch.

### 5. Results

In this section we present the results achieved on the datasets described in Section 3.2 and compare them with the models introduced in Section 4.1. If in Section 4.1 it is not stated otherwise, the results in the tables yielded by the models other than the 3 variants of MIss WiLDe (our proposed method) are cited from Choi et al. [10]. The bold font and the underline stand for the best and second-best results, respectively. Choi et al. report the results up to one place after the decimal point, which sometimes did not allow for a definite assertion as to which model performed better. As a consequence, there are columns with two values marked in bold or underlined. Using a horizontal line, we separate the results achieved by our proposed method from those achieved by other models.

As seen from Tables 1 and 2, when using either VUA-20 or VUA-18, MIss WiLDe manages to outperform all other models in terms of F1-score and Recall. It achieves the best Recall on VUA-VERB as well, however performs significantly worse in regards to both the Precision and F1-score. Surprisingly, in other cases an overall weaker variant of our model based on Sentence-BERT yields the best scores out of the 3 variants. The results for VUA-VERB are compared in Table 3. Tables 4 and 5 show the results for Genres and Parts of Speech, respectively. As for Genres, at least one of our sub-models manages to either outperform all other models or at least to draw with one of the competitors. Again MIss WiLDe performs overall better than the other models in terms of Recall, while it loses in Precision. When it comes to Parts of Speech, specifically verbs and adjectives, the base variant of MIss WiLDe proves to be by far the best model in terms of F1-score. This raises the question as to why it is not performing similarly well on the VUA-VERB dataset. The answer might be that for Genres and POS, the results are presented for the models trained beforehand on VUA-18, which provides a significantly larger train set then VUA-VERBS. Lastly, Tables 6 and 7 show the results for MOH-X and TroFi. As described in Section 3.2, in the same way as in Choi et al. [10], these datasets are used only for testing with the models trained beforehand on VUA-20. When using MOH-X, the base variant of our model outperforms the competition in terms of F1-score. Neither of our models manages to win
on TroFi; it is the model of Choi et al. [10] achieving the best F1-score and Recall for the said dataset.

Table 1. Results for VUA-ALL-POS (All Parts of Speech) and VUA-20 (variant of VUA-ALL-POS from Choi et al.; cf. Section 3.2 for details).

| Dataset     | Model    | Precision | Recall | F1   |
|-------------|----------|-----------|--------|------|
| VUA-ALL-POS | BERT     | 75.4%     | 64.7%  | 69.7%|
| VUA-20      | RoB_BASE | 74.9%     | 68.0%  | 71.2%|
|             | RoB_BASE | 74.9%     | 68.0%  | 71.2%|
|             | RoB_SEQ  | 76.9%     | 66.7%  | 71.4%|
|             | DeepMet  | 76.7%     | 65.9%  | 70.9%|
|             | MelBERT  | 76.4%     | 68.6%  | 72.3%|
|             | MsW_base | 75.4%     | 69.9%  | 72.5%|
|             | MsW_cos  | 75.4%     | 70.3%  | 72.7%|
|             | MsW_sbert| 76.2%     | 66.5%  | 71.0%|

Table 2. Results for VUA-SEQ (Sequential) and VUA-18 (variant of VUA-SEQ from Choi et al.; cf. Section 3.2 for details).

| Dataset     | Model    | Precision | Recall | F1   |
|-------------|----------|-----------|--------|------|
| VUA-SEQ     | MDGI-J-S | 81.3%     | 73.2%  | 77.0%|
|             | MDGI-J   | 82.5%     | 72.5%  | 77.2%|
|             | BERT     | 78.0%     | 76.9%  | 77.5%|
| VUA-18      | RNN_HG   | 71.8%     | 76.3%  | 74.0%|
|             | RNN_MHCA | 73.0%     | 75.7%  | 74.3%|
|             | RoB_BASE | 79.4%     | 75.0%  | 77.1%|
|             | RoB_SEQ  | 80.4%     | 74.9%  | 77.5%|
|             | DeepMet  | 82.0%     | 71.3%  | 76.3%|
|             | MelBERT  | 80.1%     | 76.9%  | 78.5%|
|             | MsW_base | 79.6%     | 78.3%  | 78.9%|
|             | MsW_cos  | 79.3%     | 78.5%  | 78.9%|
|             | MsW_sbert| 75.6%     | 71.0%  | 72.7%|

Table 3. Results for VUA-VERB.

| Dataset     | Model    | Precision | Recall | F1   |
|-------------|----------|-----------|--------|------|
| VUA-VERB    | MDGI-J-S | 78.8%     | 71.5%  | 75.0%|
|             | MDGI-J   | 78.9%     | 70.9%  | 74.7%|
|             | RNN_HG   | 69.3%     | 72.3%  | 70.8%|
|             | RNN_MHCA | 66.3%     | 75.2%  | 70.5%|
|             | RoB_BASE | 76.9%     | 72.8%  | 74.7%|
|             | RoB_SEQ  | 79.2%     | 69.8%  | 74.2%|
|             | DeepMet  | 79.5%     | 70.8%  | 74.9%|
|             | MelBERT  | 78.7%     | 72.9%  | 75.7%|
|             | MsW_base | 62.1%     | 77.1%  | 68.8%|
|             | MsW_cos  | 60.9%     | 77.7%  | 68.3%|
|             | MsW_sbert| 67.2%     | 77.7%  | 72.0%|
Table 4. Results for Genres.

| Genre | Model       | Precision | Recall | F1    |
|-------|-------------|-----------|--------|-------|
| Acad  | RNN_HG      | 76.5%     | 83.0%  | 79.6% |
|       | RNN_MHCA    | 79.6%     | 80.0%  | 79.8% |
|       | RoB_BASE    | 88.1%     | 79.5%  | 83.6% |
|       | RoB_SEQ     | 86.0%     | 77.3%  | 81.4% |
|       | DeepMet     | 88.4%     | 74.7%  | 81.0% |
|       | MelBERT     | 85.3%     | 82.5%  | 83.9% |
|       | MsW_base    | 85.3%     | 82.4%  | 83.8% |
|       | MsW_cos     | 84.4%     | 83.5%  | 84.0% |
|       | MsW_sbert   | 85.6%     | 81.0%  | 83.2% |
| Conv  | RNN_HG      | 63.6%     | 72.5%  | 67.8% |
|       | RNN_MHCA    | 64.0%     | 71.1%  | 67.4% |
|       | RoB_BASE    | 70.3%     | 69.0%  | 69.6% |
|       | RoB_SEQ     | 70.5%     | 69.8%  | 70.1% |
|       | DeepMet     | 71.6%     | 71.1%  | 71.4% |
|       | MelBERT     | 70.1%     | 71.7%  | 70.9% |
|       | MsW_base    | 69.2%     | 73.8%  | 71.4% |
|       | MsW_cos     | 70.2%     | 72.7%  | 71.4% |
|       | MsW_sbert   | 68.0%     | 73.5%  | 70.6% |
| Fict  | RNN_HG      | 61.8%     | 74.5%  | 67.5% |
|       | RNN_MHCA    | 64.8%     | 70.9%  | 67.7% |
|       | RoB_BASE    | 74.3%     | 71.2%  | 73.2% |
|       | RoB_SEQ     | 73.9%     | 72.7%  | 73.3% |
|       | DeepMet     | 76.1%     | 70.1%  | 73.0% |
|       | MelBERT     | 74.0%     | 76.8%  | 75.4% |
|       | MsW_base    | 73.6%     | 77.6%  | 75.5% |
|       | MsW_cos     | 73.3%     | 77.1%  | 75.2% |
|       | MsW_sbert   | 72.9%     | 77.3%  | 75.0% |
| News  | RNN_HG      | 71.6%     | 76.8%  | 74.1% |
|       | RNN_MHCA    | 74.8%     | 75.3%  | 75.0% |
|       | RoB_BASE    | 83.5%     | 71.8%  | 77.2% |
|       | RoB_SEQ     | 82.2%     | 74.1%  | 77.9% |
|       | DeepMet     | 84.1%     | 67.6%  | 75.0% |
|       | MelBERT     | 81.0%     | 73.7%  | 77.2% |
|       | MsW_base    | 82.2%     | 76.1%  | 79.0% |
|       | MsW_cos     | 81.5%     | 76.0%  | 78.7% |
|       | MsW_sbert   | 81.6%     | 74.3%  | 77.8% |

Although, as mentioned before, in the tables we quote the results published by Choi et al. in [10], at the same time we would like to share the results of MelBERT’s reruns for both VUA-20 and VUA-18 that we have conducted ourselves using the same set of random seeds as well as the same GPU. These are as follows: VUA-20 (Precision: 75.8%, Recall: 69.6%, F1: 72.6%); VUA-18 (Precision: 79.9%, Recall: 77.3%, F1: 78.6%). Additionally, we have compared these F1-scores with the F1-scores achieved by MsW_base and MsW_cos, using two-tailed t-test. The value achieved was $p > 0.05$, meaning that the differences are not statistically significant.
Table 5. Results for POS (Parts of Speech).

| POS  | Model      | Precision | Recall | F1   |
|------|------------|-----------|--------|------|
| Verb | RNN_HG     | 66.4%     | 75.5%  | 70.7%|
|      | RNN_MHCA   | 66.0%     | 76.0%  | 70.7%|
|      | RoB_BASE   | 77.0%     | 72.1%  | 74.5%|
|      | RoB_SEQ    | 74.4%     | 75.1%  | 74.8%|
|      | DeepMet    | 78.8%     | 68.5%  | 73.3%|
|      | MelBERT    | 74.2%     | 75.9%  | 75.1%|
|      | MsW_base   | 74.9%     | 78.1%  | 76.5%|
|      | MsW_cos    | 74.0%     | 77.9%  | 75.9%|
|      | MsW_sbert  | 74.5%     | 76.1%  | 75.3%|
| Adj  | RNN_HG     | 59.2%     | 65.6%  | 62.2%|
|      | RNN_MHCA   | 61.4%     | 61.1%  | 61.6%|
|      | RoB_BASE   | 71.7%     | 59.0%  | 64.7%|
|      | RoB_SEQ    | 72.0%     | 57.1%  | 63.7%|
|      | DeepMet    | 79.0%     | 52.9%  | 63.3%|
|      | MelBERT    | 69.4%     | 60.1%  | 64.4%|
|      | MsW_base   | 68.5%     | 65.0%  | 66.7%|
|      | MsW_cos    | 67.5%     | 65.1%  | 66.3%|
|      | MsW_sbert  | 69.3%     | 63.8%  | 66.4%|
| Adv  | RNN_HG     | 61.0%     | 66.8%  | 63.8%|
|      | RNN_MHCA   | 66.1%     | 60.7%  | 63.2%|
|      | RoB_BASE   | 78.2%     | 69.3%  | 73.5%|
|      | RoB_SEQ    | 77.6%     | 63.9%  | 70.1%|
|      | DeepMet    | 79.4%     | 66.4%  | 72.3%|
|      | MelBERT    | 80.2%     | 69.7%  | 74.6%|
|      | MsW_base   | 76.4%     | 69.3%  | 72.7%|
|      | MsW_cos    | 78.9%     | 69.8%  | 74.0%|
|      | MsW_sbert  | 78.3%     | 67.5%  | 72.4%|
| Noun | RNN_HG     | 60.3%     | 66.8%  | 63.4%|
|      | RNN_MHCA   | 69.1%     | 58.2%  | 63.2%|
|      | RoB_BASE   | 77.5%     | 60.4%  | 67.9%|
|      | RoB_SEQ    | 76.5%     | 59.0%  | 66.6%|
|      | DeepMet    | 76.5%     | 57.1%  | 65.4%|
|      | MelBERT    | 75.4%     | 66.5%  | 70.7%|
|      | MsW_base   | 76.2%     | 64.9%  | 70.1%|
|      | MsW_cos    | 75.4%     | 66.0%  | 70.4%|
|      | MsW_sbert  | 73.9%     | 64.7%  | 69.0%|

Table 6. Results for MOH-X (Mohammad et al. [2016] dataset).

| Dataset | Model      | Precision | Recall | F1   |
|---------|------------|-----------|--------|------|
| MOH-X   | DeepMet    | 79.9%     | 76.5%  | 77.9%|
|         | MelBERT    | 79.3%     | 79.7%  | 79.2%|
|         | MsW_base   | 82.3%     | 80.0%  | 80.8%|
|         | MsW_cos    | 81.0%     | 80.0%  | 80.2%|
|         | MsW_sbert  | 83.3%     | 77.7%  | 80.1%|
Table 7. Results for TroFi (Trope Finder).

| Dataset | Model       | Precision | Recall | F1   |
|---------|-------------|-----------|--------|------|
| TroFi   | DeepMet     | 53.7%     | 72.9%  | 61.7%|
|         | MelBERT     | 53.4%     | 74.1%  | 62.0%|
|         | MsW_base    | 53.3%     | 73.4%  | 61.7%|
|         | MsW_cos     | 53.2%     | 72.8%  | 61.4%|
|         | MsW_sbert   | 53.7%     | 69.7%  | 60.6%|

6. Discussion

In this section we present our considerations regarding the experimental results introduced above. Example sentences presented in this section come from the output of the experiment performed on VUA-20, sharing common test data with VUA-ALL-POS, which is considered one of the most representative datasets for evaluating the performance of the algorithms designed for metaphor detection. Target words are marked in bold font. We compare the predictions of Choi et al.’s MelBERT and our model. Out of the three variants of Miss WiLDe, we have chosen the base one. In doing so, we can argue that the differences in the output stem mostly from our decision to use lexical definitions instead of the isolated target word. As mentioned earlier, for each of our models we have conducted 5 runs of experiments, using the same random seeds ranging from 1 to 5 for fairness. Since the output and its specific values may slightly vary between random seeds, for both MelBERT and Miss WiLDe_base, we use the outputs of the models having yielded the best F1-score among all 5 seeds.

6.1. Results Analysis

As can be told from the tables above, Miss WiLDe managed to outperform competing models in several categories, most significantly in adjectives (cf. Table 5). Consider the following example:

Eventually they will be replaced, but more than 60 years on they run with the rhythmic reliability of a Swiss watch.

Here our model voted for metaphorical use with a high degree of confidence (0.2:0.8), while MelBERT estimated it was literal (0.66:0.34). The definition of the target word retrieved from Wiktionary and used by our model is ‘Of or relating to rhythm.’ and this of the modified noun is ‘The quality of being reliable, dependable, or trustworthy.’ In terms of Wilks [1], Miss WiLDe managed to detect the violation of semantic restrictions present between basic senses of the target adjective rhythmic and the noun reliability.

Another example including an adjective is the following:

There is a refreshing simplicity and tenderness in Motion’s account of the way Francis nurses her, but she herself is too sketchily drawn for the episode to carry much weights.

The definition of refreshing is ‘That refreshes someone; pleasantly fresh and different; granting vitality and energy’, while simplicity is defined as ‘The state or quality of being simple’. Although the middle part of the adjective’s definition seems to already point at the figurative meaning of the word, it can still be argued that overall it is more literal than metaphorical and therefore it constitutes the word’s basic meaning. In the prototypical situation, for its modifier, refreshing would demand a noun belonging to the semantic field of fluids, drinks. Simplicity is not fulfilling this condition, which hints at figurative usage. Although the phrase refreshing simplicity is quite commonly used, our model managed to detect that it is metaphorically motivated. Due to its high frequency in corpora, it would be difficult to argue that such word matching is unnatural or exotic. As recent language models are pretrained also on the texts from the books, where juxtapositions similar to refreshing simplicity are observed quite often, without the use of definitions, it would be difficult for them to discern metaphoricity underlying such wordings. In other words, without the definitions conveying the basic meanings it would be hard to find any indication
of breaking some semantic restrictions. In our case, MelBERT voted 0.84:0.16 for refreshing to be used literally, while MiSS WiLDe leaned towards metaphoricity with the probabilities of 0.41:0.59. By looking at the estimation scores, it can be assumed that making the decision was not easy for our model, which can be explained with the reasoning just outlined.

An incoming Labour government would turn large areas of Whitehall upside down (…) Phrase incoming Labour government from the sentence above can be viewed as an example of TIME AS SPACE metaphor ([57,58]), where the target word incoming is used in the sense of ‘future’. Its primary meaning related to physical motion is described by the first definition found in Wiktionary and thus provided to our model (‘Coming in; arriving’). The contextual meaning is portrayed by the second of the Wiktionary’s definitions (‘Succeeding to an office’). This example also shows that, as a general rule, our choice to use only first definitions as the ones likely to convey basic meaning was right. In this example, MelBERT predicted incoming as being used literally (0.67:0.33), while MiSS WiLDe made a correct and firm decision voting for the metaphor (0.14:0.86).

6.2. Error Analysis

It should be noted that a word’s basic meaning sometimes does not provide much help in metaphor detection. Consider the following example:

There are always accusations of piracy and copy-catting, though they can’t usually be substantiated.

The target word’s definition in this case is ‘Robbery at sea, a violation of international law; taking a ship away from the control of those who are legally entitled to it’, which indeed probably should be considered the word’s basic sense, as it is historically older than ‘The unauthorized duplication of goods protected by intellectual property law’ (the third from the definitions listed in Wiktionary). One should notice that in this case the target word does not violate any selectional preferences. Both senses relate to the notion of CRIME and the target word used in either of them does not sound particularly awkward in the given context. In such cases, utilizing the information provided by the SPV (Selectional Preference Violation) module does not resolve the problem. As our method adopts only simplified interpretation of MIP (Metaphor Identification Procedure) as well, it cannot differentiate between the word senses based on historical reasons. When encountering samples of this kind, our model is rather powerless.

In our opinion, some of the missed predictions are the outcome of an inaccurate annotation process. Consider the following example:

See if you can rustle up a cup of tea for Paula and me, please.

Although the target word from the above sentence is annotated as a non-metaphor, we strongly believe that in this context, it is used figuratively. Following the MIP procedure, the annotator should first read the whole sentence to understand its meaning. Next, it should be determined whether the target word has a more basic meaning in other contexts. The first Wiktionary definition of the target word is ‘To perceive or detect someone or something with the eyes, or as if by sight’. At this point it should be already clear that in this case, the contextual meaning of the verb see differs from the meaning described by the first definition. The context suggests that semantically it is closer to ‘To determine by trial or experiment; to find out (if or whether)’, the eighth of the Wiktionary definitions for the verb see. As the basic meaning tends to be more concrete and related to bodily action ([12], p. 4), it should be rather straightforward to judge the contextual meaning as non-basic and therefore metaphorical. Assuming that we are correct, we have to give credit to MelBERT, which estimated very confidently that it deals with a metaphor (0.04:0.96). MiSS WiLDe also voted for the metaphor, however apparently it was not an easy choice (0.41:0.59).
7. Conclusions and Future Work

In this paper, we proposed MIss RoBERTa WiLDe (Metaphor Identification using Masked Language Model with Wiktionary Lexical Definitions), a model designed for automatic metaphor detection. Our method is logically consistent and supported theoretically, as utilizing literal basic meanings of words follows the guidelines of Metaphor Identification Procedure (MIP) and the concept of Selectional Preference Violation (SPV). We argue that there is no better source of purely literal word senses than the lexical definitions of said words. We have enhanced the existing algorithm [10] by introducing a different kind of sentence representation and collecting dictionary definitions of the target words in a fully automatic manner. The results we achieved in the set of experiments suggest that implementing our ideas can lead to performance gains in metaphor identification tasks.

As indicated by Mao et al. [36], having access to large-scale textual resources using words only in their basic literal meanings could elevate the performance of the algorithms used for metaphor detection. However, as already mentioned, metaphors are abundant in human language irrespective of the genre and, because of it, finding a perfect knowledge base of this kind does not seem possible. Nonetheless, we think that in comparison to other types of currently available resources, it is very probable that dictionaries are closest to that ideal.

Having witnessed that our method was successful using lexical data in English, in future we plan to introduce a similar method for Japanese as well. However, because Japanese Wiktionary is relatively small in comparison to its English counterpart, we may require a different source for collecting definitions.

Author Contributions: Conceptualization, M.T. and M.B.; methodology, M.T.; software, D.R. and M.T.; validation, M.B., investigation, M.B.; data curation, M.B.; writing—original draft preparation, M.B.; writing—review and editing, R.R. and K.A.; visualization, D.R. (figures) and M.B. (tables); supervision, R.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data can be found at the following address: https://github.com/languagemedialaboratory/ms_wilde; last accessed on 21 December 2021.

Conflicts of Interest: The authors declare no conflict of interest.

Sample Availability: Samples of the compounds are available from the authors.

Abbreviations
The following abbreviations are used in this manuscript:

| Abbreviation | Description                                      |
|--------------|--------------------------------------------------|
| Acad         | academic                                         |
| Adj          | adjective                                        |
| Adv          | adverb                                           |
| ALBERT       | A Lite BERT                                      |
| ALL-POS      | All Parts of Speech                              |
| API          | application programming interface                |
| BERT         | Bidirectional Encoder Representations from Transformers |
| BPE          | Byte-Pair Encoding                               |
| CLS          | classification                                   |
| Conv         | conversation                                     |
| cos, CS      | cosine similarity                                |
| ELMo         | Embeddings from Language Model                   |
| ENC          | encoder                                          |
| Fict         | fiction                                          |
| GloVe        | Global Vectors                                   |
| GPT          | Generative Pre-trained Transformer               |
GPU
MelBERT
MIP
MIPVU
MOH-X
NLP
POS
PSUCMC
RNN_HG
RNN_MHCA
RoBERTa
sbert
SEP
SEQ
SOEDHP
SPV
TOK
TroFi
VUAMC
XML

References
1. Wilks, Y. Making preferences more active. *Artif. Intell.* 1978, 11, 197–223. [CrossRef]
2. Miller, E.F. Metaphor and Political Knowledge. *Am. Political Sci. Rev.* 1979, 73, 155–170. [CrossRef]
3. Lakoff, G. Metaphor and War: The Metaphor System Used to Justify War in the Gulf. *Cogn. Semiot.* 2012, 4, 5–19. [CrossRef]
4. Lakoff, G. Metaphor, morality, and politics, or, why conservatives have left liberals in the dust. *Soc. Res.* 1995, 62, 177–213.
5. Siegelman, E.Y. *Metaphor and Meaning in Psychotherapy*; Guilford Press: New York, NY, USA, 1993.
6. Kopp, R.R. *Metaphor Therapy: Using Client Generated Metaphors in Psychotherapy*; Routledge: New York, NY, USA, 2013.
7. Cornelissen, J.P. Metaphor as a method in the domain of marketing. *Psychol. Mark.* 2003, 20, 209–225. [CrossRef]
8. Hellsten, I.; Renvall, M. Inside or outside of politics? Metaphor and paradox in journalism. *Nord. Rev.* 1997, 18, 41–47.
9. Partington, A. A corpus-based investigation into the use of metaphor in British business journalism. *ASp* 1995, 25–39. [CrossRef]
10. Choi, M.; Lee, S.; Choi, E.; Park, H.; Lee, J.; Lee, D.; Lee, J. MelBERT: Metaphor Detection via Contextualized Late Interaction using Metaphorical Identification Theories. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online*, 6–11 June 2021; pp. 1763–1773. [CrossRef]
11. Stevenson, A. *Oxford Dictionary of English*; Oxford University Press: New York, NY, USA, 2010.
12. Group, P. MIP: A Method for Identifying Metaphorically Used Words in Discourse. *Metaphor. Symb.* 2007, 22, 1–39. [CrossRef]
13. Rai, S.; Chakraverty, S. A Survey on Computational Metaphor Processing. *ACM Comput. Surv.* 2021, 53, 1–37. [CrossRef]
14. Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019; Burstein, J., Doran, C., Solorio, T., Eds.; pp. 4171–4186. [CrossRef]
15. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* 2019, arXiv:1907.11692.
16. Peters, M.E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep Contextualized Word Representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, Louisiana, June 2018; Association for Computational Linguistics: New Orleans, Louisiana, 2018; pp. 2227–2237. [CrossRef]
21. Su, C.; Fukumoto, F.; Huang, X.; Li, J.; Wang, R.; Chen, Z. DeepMet: A Reading Comprehension Paradigm for Token-level Metaphor Detection. In Proceedings of the Second Workshop on Figurative Language Processing, Online, 9 July 2020; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020, pp. 30–39. [CrossRef]

22. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All You Need. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Curran Associates Inc.: Red Hook, NY, USA, 2017; NIPS’17, pp. 6000–6010.

23. Wan, H.; Lin, J.; Du, J.; Shen, D.; Zhang, M. Enhancing Metaphor Detection by Gloss-based Interpretations. In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021. [CrossRef]

24. Birke, J.; Sarkar, A. A Clustering Approach for Nearly Unsupervised Recognition of Nonliteral Language. In Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics, Trento, Italy, April 2006; Association for Computational Linguistics: Trento, Italy, 2006; pp. 329–336.

25. Lu, X.; Wang, B.P.Y. Towards a metaphor-annotated corpus of Mandarin Chinese. Lang. Resour. Eval. 2017, 51, 663–694. [CrossRef]

26. Fass, D.; Wilks, Y. Preference semantics, ill-formedness, and metaphor. Comput. Linguist. 2002, 9, 178–187.

27. Mohammad, S.; Shutova, E.; Turney, P. Metaphor as a Medium for Emotion: An Empirical Study. In Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics, Berlin, Germany, 11–12 August 2016; Association for Computational Linguistics: Berlin, Germany, 2016; pp. 23–33. [CrossRef]

28. Chintalapudi, N.; Battineni, G.; Amenta, F. Sentimental Analysis of COVID-19 Tweets Using Deep Learning Models. Infect. Dis. Rep. 2021, 13, 329–339. [CrossRef]

29. Fass, D. Met*: A Method for Discriminating Metonymy and Metaphor by Computer. Comput. Linguist. 1991, 17, 49–90.

30. Mason, Z.J. CorMet: A Computational, Corpus-Based Conventional Metaphor Extraction System. Comput. Linguist. 2004, 30, 23–44. [CrossRef]

31. Shutova, E.; Sun, L.; Korhonen, A. Metaphor identification using verb and noun clustering. In Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), Beijing, China, 23–27 August 2010; pp. 1002–1010.

32. Neuman, Y.; Assaf, D.; Cohen, Y.; Last, M.; Argamon, S.; Howard, N.; Frieder, O. Metaphor Identification in Large Texts Corpora. PloS ONE 2013, 8, e62343. [CrossRef] [PubMed]

33. Wilks, Y.; Dalton, A.; Allen, J.; Galescu, L. Automatic Metaphor Detection using Large-Scale Lexical Resources and Conventional Metaphor Extraction. In Proceedings of the First Workshop on Metaphor in NLP, Atlanta, GA, USA 13 June 2013; Association for Computational Linguistics: Atlanta, Georgia, 2013; pp. 36–44.

34. Haagsma, H.; Bjerva, J. Detecting novel metaphor using selectional preference information. In Proceedings of the Fourth Workshop on Metaphor in NLP, San Diego, CA, USA, June 2016. [CrossRef]

35. Gao, G.; Choi, E.; Choi, Y.; Zettlemoyer, L. Neural Metaphor Detection in Context. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018. [CrossRef]

36. Mao, R.; Lin, C.; Guerin, F. End-to-End Sequential Metaphor Identification Inspired by Linguistic Theories. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; Association for Computational Linguistics: Florence, Italy, 2019; pp. 3888–3898. [CrossRef]

37. Heywood, J.; Semino, E.; Short, M. Linguistic metaphor identification in two extracts from novels. Lang. Lit. Int. J. Stylist. 2002, 11, 35–54. [CrossRef]

38. Semino, E.; Heywood, J.; Short, M. Methodological problems in the analysis of metaphors in a corpus of conversations about cancer. J. Pragmat. 2004, 36, 1271–1294. [CrossRef]

39. Steen, G.; Dorst, L.; Herrmann, J.; Kaal, A.; Krennmayr, T. Metaphor in usage. Cogn. Linguist. 2010, 21, 765-796. [CrossRef]

40. Shutova, E.; Teufel, S.; Korhonen, A. Statistical Metaphor Processing. Comput. Linguist. 2013, 39, 301–353. [CrossRef]

41. Reimers, N.; Gurevych, I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019. [CrossRef]

42. Rafad, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. OpenAI Blog 2019, 1, 9.

43. Mhamdi, E.M.E.; Guerraoui, R.; Rouault, S. Distributed Momentum for Byzantine-resilient Stochastic Gradient Descent. In Proceedings of the 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, 3–7 May 2021.

44. Leong, C.W.B.; Klebanov, B.B.; Shutova, E. A Report on the 2018 VUA Metaphor Detection Shared Task. In Proceedings of the Workshop on Figurative Language Processing, New Orleans, Louisiana, 6 June 2018; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 56–66. [CrossRef]

45. Fellbaum, C. WordNet: An Electronic Lexical Database; Bradford Books: Cambridge, MA, US, 1998.

46. Charniak, E. BLLIP 1987-1989 WSJ corpus. LDS 2000. [CrossRef]

47. Steen, G.J.; Dorst, A.G.; Herrmann, J.B.; Kaal, A.A.; Krennmayr, T. VU Amsterdam Metaphor Corpus. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Florence, Italy, 2019; pp. 3888–3898. [CrossRef]
48. Boers, F. *Spatial Prepositions and Metaphor: A Cognitive Semantic Journey Along the Up-Down and the Front-Back Dimensions*; Gunter Narr Verlag: Tübingen, Germany, 1996; Volume 12.

49. Haspelmath, M. *From Space to Time: Temporal Adverbials in the World’s Languages*; Lincom Europa: Munich, Germany, 1997. [CrossRef]

50. Pütz, M.; Dirven, R., Eds. *The Construal of Space in Language and Thought*; De Gruyter Mouton: Berlin, Germany; New York, NY, USA, 2011. [CrossRef]

51. Delahunty, G.P.; Garvey, J.J. *The English Language: From Sound to Sense*; Parlor Press LLC: Anderson, SC, USA, 2010.

52. Haspelmath, M. Word Classes and Parts of Speech. In *International Encyclopedia of the Social & Behavioral Sciences*; Elsevier: Amsterdam, The Netherlands, 2001; pp. 16538–16545. [CrossRef]

53. Corver, N.; van Riemsdijk, H., Semi-Lexical Categories. In *Semi-Lexical Categories: The Function of Content Words and the Content of Function Words*; Corver, N., van Riemsdijk, H., Eds.; De Gruyter Mouton: Berlin, Germany; Boston, MA, USA, 2013; pp. 1–20. [CrossRef]

54. Lakoff, G.; Johnson, M. *Metaphors We Live By*; University of Chicago Press: Chicago, IL, USA, 2003. [CrossRef]

55. Raymond W. Gibbs, J.; Colston, H.L. Chapter 7 Image schema. In *Cognitive Linguistics Research*; Mouton de Gruyter: Berlin, Germany; New York, NY, USA, 2006; pp. 239–268. [CrossRef]

56. Van Sterkenburg, P. (Ed.) *A Practical Guide to Lexicography*; John Benjamins Publishing Company: Amsterdam, The Netherlands; Philadelphia, PA, USA, 2003. [CrossRef]

57. Fauconnier, G.; Turner, M. Rethinking metaphor. In *The Cambridge Handbook of Metaphor and Thought*; Gibbs, R.W.J., Ed.; Cambridge University Press: Cambridge, UK, 2008; pp. 53–66. [CrossRef]

58. Radden, G. The metaphor TIME AS SPACE across languages. *Z. Interkulturellen Fremdsprachenunterricht* 2003, 8, 226–239.