Predicting Human Metaphor Paraphrase Judgments with Deep Neural Networks

Yuri Bizzoni and **Shalom Lappin**
University of Gothenburg
firstname.lastname@gu.se

Abstract

We propose a new annotated corpus for metaphor interpretation by paraphrase, and a novel DNN model for performing this task. Our corpus consists of 200 sets of 5 sentences, with each set containing one reference metaphorical sentence, and four ranked candidate paraphrases. Our model is trained for a binary classification of paraphrase candidates, and then used to predict graded paraphrase acceptability. It reaches an encouraging 75% accuracy on the binary classification task, and high Pearson (.75) and Spearman (.68) correlations on the gradient judgment prediction task.

1 Introduction

Metaphor is an increasingly studied phenomenon in computational linguistics. But while metaphor detection has received considerable attention in the NLP literature (Dunn et al., 2014; Veale et al., 2016) and in corpus linguistics (Krennmayr, 2015) in recent years, not much work has focused on the task of metaphor paraphrasing - assigning an appropriate interpretation to a metaphorical expression. Moreover, there are few (if any) annotated corpora of metaphor paraphrases (Shutova and Teufel, 2010). The main papers in this area are Shutova (2010), and Bollegala and Shutova (2013). The first applies a supervised method combining WordNet and distributional word vectors to produce the best paraphrase of a single verb used metaphorically in a sentence. The second approach, conceptually related to the first, builds an unsupervised system that, given a sentence with a single metaphorical verb and a set of potential paraphrases, selects the most accurate candidate through a combination of mutual information scores and distributional similarity.

Despite the computational and linguistic interest of this task, little research has been devoted to it.

Some quantitative analyses of figurative language have involved metaphor interpretation and paraphrasing. These focus on integrating paraphrase into automatic Textual Entailment frames (Agerri, 2008), to explore the properties of distributional semantics in larger-than-word structures (Turney, 2013). Alternatively, they study the sentiment features of metaphor usage (Mohammad et al., 2016; Kozareva, 2015). This last aspect of figurative interpretation is considered a particularly hard task and has generated several approaches

The task of metaphor interpretation is a particular case of paraphrase detection, although this characterization is not unproblematic, as we will see in Section 6.

In Bollegala and Shutova (2013), metaphor paraphrase is treated as a ranking problem. Given a metaphorical usage of a verb in a short sentence, several candidate literal sentences are retrieved from the Web and ranked. This approach requires the authors to create a gradient score to label their paraphrases, a perspective that is now gaining currency in broader semantic similarity tasks (Xu et al., 2015; Agirre et al., 2016).

Mohammad et al. (2016) resort to metaphor paraphrasing in order to perform a quantitative study on the emotions associated with the usage of metaphors. They create a small corpus of paraphrase pairs formed from a metaphorical expression and a literal equivalent. They ask candidates to judge the degree of ”emotionality” conveyed by the metaphorical and the literal expressions. While the study has shown that metaphorical paraphrases are generally perceived as more emotionally charged than their literal equivalents, a corpus of this kind has not been used to train a computational model for metaphor paraphrase scoring.

In this paper we present a new dataset for

Proceedings of the Workshop on Figurative Language Processing, pages 45–55
New Orleans, Louisiana, June 6, 2018.

metaphor paraphrase identification and ranking. In our corpus, paraphrase recognition is treated as an ordering problem, where sets of sentences are ranked with respect to a reference metaphor sentence.

The main difference with respect to existing work in this field consists in the syntactic and semantic diversity covered by our dataset. The metaphors in our corpus are not confined to a single part of speech. We introduce metaphorical examples of nouns, adjectives, verbs and a number of multi-word metaphors.

Our corpus is, to the best of our knowledge, the largest existing dataset for metaphor paraphrase detection and ranking.

As we describe in Section 2, it is composed of groups of five sentences: one metaphor, and four candidates that can be ranked as its literal paraphrases.

The inspiration for the structure of our dataset comes from a recent work on paraphrase (Bizzoni and Lappin, 2017), where a similarly organized dataset was introduced to deal with paraphrase detection.

In our work, we use an analogous structure to model metaphor paraphrase. Also, while Bizzoni and Lappin (2017) present a corpus annotated by a single human, each paraphrase set in our corpus was judged by 20 different Amazon Mechanical Turk (AMT) annotators, making the grading of our sentences more robust and reliable (see Section 2.1).

We use this corpus to test a neural network model formed by a combination of Convolutional Neural Networks (CNNs) and Long Short Term Memory Recurrent Neural Networks (LSTM RNNs). We test this model on two classification problems: (i) binary paraphrase classification and (ii) paraphrase ranking. We show that our system can achieve significant correlation with human judgments on the ranking task as a by-product of supervised binary learning. To the best of our knowledge, this is the first work in metaphor paraphrasing to use supervised gradient representations.

2 A New Corpus for Metaphor Paraphrase Evaluation

We present a dataset for metaphor paraphrase designed to allow users to rank non-metaphorical candidates as paraphrases of a metaphorical sentence or expression. Our corpus is formed of 200 sets of five sentence paraphrase candidates for a metaphorical sentence or expression.[1]

In each set, the first sentence contains a metaphor, and it provides the reference sentence to be paraphrased. The remaining four sentences are labeled on a 1-4 scale based on the degree to which they paraphrase the reference sentence. This is on analogy with the annotation frame used for SemEval Semantic Similarity tasks (Agirre et al., 2016). Broadly, our labels represent the following categories:

1 Two sentences cannot be considered paraphrases.

2 Two sentences cannot be considered paraphrase, but they show a degree of semantic similarity.

3 Two sentences could be considered paraphrases, although they present some important difference in style or content (they are not strong paraphrases).

4 Two sentences are strong paraphrases.

On average, every group of five sentences contains a strong paraphrase, a loose paraphrase and two non-paraphrases, one of which may use some relevant words from the metaphor in question.[2]

The following examples illustrate these ranking labels.

- Metaphor: *The crowd was a river in the street*
  - The crowd was large and impetuous in the street. *Score: 4*
  - There were a lot of people in the street. *Score: 3*
  - There were few people in the street. *Score: 2*
  - We reached a river at the end of the street. *Score: 1*

We believe that this annotation scheme is useful. While it sustains graded semantic similarity labels, it also provides sets of semantically related

[1] Our annotated data set and the code for our model is available at https://github.com/yuri-bizzoni/Metaphor-Paraphrase .

[2] Some of the problems raised by the concept of paraphrase in figurative language are discussed in Section 6.

elements, each one of which can be scored or ordered independently of the others. Therefore, the metaphorical sentence can be tested separately for each literal candidate in the set in a binary classification task.

In the test phase, the annotation scheme allows us to observe how a system represents the similarity between a metaphorical and a literal sentence by taking the scores of two candidates as points of relative proximity to the metaphor.

It can be argued that a good literal paraphrase of a metaphor needs to compensate to some extent for the expressive or sentimental bias that a metaphor usually supplies, as argued in Mohammad et al. (2016). In general a binary classification can be misleading because it conceals the different levels of similarity between competing candidates.

For example, the literal sentence *Republican candidates during the convention were terrible* can be considered to be a loose paraphrase of the metaphor **The Republican convention was a horror show**, or alternatively, as a semantically related non-paraphrase. Which of these conclusions we adopt depends on our decision concerning how much interpretative content a literal sentence needs to provide in order to qualify as a valid paraphrase of a metaphor. The question whether the two sentences are acceptable paraphrases or not can be hard to answer. By contrast, it would be far fetched to suggest that *The Republican convention was a joy to follow* is a better or even equally strong literal paraphrase for **The Republican convention was a horror show**.

In this sense, the sentences **Her new occupation was a dream come true** and *She liked her new occupation* can be considered to be loose paraphrases, in that the term *liked* can be judged an acceptable, but not ideal interpretation of the more intense metaphorical expression **a dream come true**. By contrast, *She hated her new occupation* cannot be plausibly regarded as more similar in meaning than *She liked her new occupation* to **Her new occupation was a dream come true**.

Our training dataset is divided into four main sections:

1. Noun phrase Metaphors : *My lawyer is an angel*.
2. Adjective Metaphors : *The rich man had a cold heart*.
3. Verb Metaphors : *She cut him down with her words*.
4. Multi-word Metaphors : *The seeds of change were planted in 1943*.

All these sentences and their candidates were manually produced to insure that for each group we have a strong literal paraphrase, a loose literal paraphrase and two semantically related non-paraphrases. Here "semantically related" can indicate either a re-use of the metaphorical words to express a different meaning, or an unacceptable interpretation of the reference metaphor.

Although the paraphrases were generated freely and cover a number of possible (mis)interpretations, we did take several issues into account. For example, for sentiment related metaphors two opposite interpretations are often proposed, forcing the system to make a choice between two sentiment poles when ranking the paraphrases (*I love my job – I hate my job* for *My job is a dream*). In general, antonymous interpretations (*Time passes very fast – Time is slow* for *Time flies*) are listed, when possible, among the four competing choices.

Our corpus has the advantage of being suitable for both binary classification and gradient paraphrase judgment prediction. For the former, we map every score over a given gradient threshold label to 1, and scores below that threshold to 0. For gradient classification, we use all the scoring labels to test the correlation between the system's ordered predictions and human judgments. We will show how, once a model has been trained for a binary detection task, we can evaluate its performance on the gradient ordering task.

We stress that our corpus is under development. As far as we know it is unique for the kind of task we are discussing. The main difficulty in building this corpus is that there is no obvious way to collect the data automatically. Even if there were a procedure to extract pairs of paraphrases containing a metaphoric element semi-automatically, it does not seem possible to generate alternative paraphrase candidates automatically.

The reference sentences we chose were either selected from published sources or created manually by the authors. In all cases, the paraphrase candidates had to be crafted manually. We tried to keep a balanced diversity inside the corpus. The dataset is divided among metaphorically used Nouns, Adjectives and Verbs, plus a section of

Multi Word metaphors. The corpus is an attempt to represent metaphor in different parts of speech.

A native speaker of English independently checked all the sentences for acceptability.

2.1 Collecting judgments through AMT

Originally, one author individually annotated the entire corpus. The difference between strong and loose literal paraphrases can be a matter of individual sensibility.

While such annotations could be used as the basis for a preliminary study, we needed more judgments to build a statistically reliable annotated dataset. Therefore we used crowd sourcing to solicit judgments from large numbers of annotators. We collected human judgments on the degree of paraphrasehood for each pair of sentences in a set (with the reference metaphor sentence in the pair) through Amazon Mechanical Turk (AMT).

Annotators were presented with four *metaphor - candidate paraphrase* pairs, all relating to the same metaphor. They were asked to express a judgment between 1 and 4, according to the scheme given above.

We collected 20 human judgments for each pair *metaphor - candidate paraphrase*. Analyzing individual annotators' response patterns, we were able to filter out a small number of "rogue" annotators (less than 10%). This filtering process was based on annotators' answers to some control elements inserted in the corpus, and evaluation of their overall performance. For example, an annotator who consistently assigned the same score to all sentences is classified as "rogue".

We then computed the mean judgment for each sentence pair and compared it with the original judgments expressed by one of the authors. We found a high Pearson correlation between the annotators' mean judgments and the author's judgment of close to 0.93.

The annotators' understanding of the problem and their evaluation of the sentence pairs seem, on average, to correspond very closely to that of our original single annotator. The high correlation also suggests a small level of variation from the mean across AMT annotators. Finally, a similar correlation strengthens the hypothesis that paraphrase detection is better modeled as an ordering, rather than a binary, task. If this had not been the case, we would expect more polarized judgments tending towards the highest and lowest scores, instead of the more evenly distributed judgment patterns that we observed.

These mean judgments appear to provide reliable data for supervision of a machine learning model. We thus set the upper bound for the performance of a machine learning algorithm trained on this data to be around .9, on the basis of the Pearson correlation with the original single annotator scores. In what follows, we refer to the mean judgments of AMT annotators as our gold standard when evaluating our results, unless otherwise indicated.

3 A DNN for Metaphor Paraphrase Classification

For classification and gradient judgment prediction we constructed a deep neural network. Its architecture consists of three main components:

1. Two encoders that learn the representation of two sentences separately
2. A unified layer that merges the output of the encoders
3. A final set of fully connected layers that operate on the merged representation of the two sentences to generate a judgment.

The encoder for each pair of sentences taken as input is composed of two parallel Convolutional Neural Networks (CNNs) and LSTM RNNs, feeding two sequenced fully connected layers. We use an "Atrous" CNN (Chen et al., 2016). Interestingly, classical CNNs only decrease our accuracy by approximately two points and reach a good F1 score, as Table 1 indicates.

Using a CNN (we apply 25 filters of length 5) as a first layer proved to be an efficient strategy. While CNNs were originally introduced in the field of computer vision, they have been successfully applied to problems in computational semantics, such as text classification and sentiment analysis (Lai et al., 2015), as well as to paraphrase recognition (Socher et al., 2011). In NLP applications, CNNs usually abstract over a series of word- or character-level embeddings, instead of pixels. In this part of our model, the encoder learns a more compact representation of the sentence, with reduced vector space dimensions and features. This permits the entire DNN to focus on the information most relevant to paraphrase identification.

The output of each CNN is passed through a max pooling layer to an LSTM RNN. Since the CNN and the max pooling layer perform discriminative reduction of the input's dimensions, we can run a relatively small LSTM RNN model (20 hidden units). In this phase, the vector dimensions of the sentence representation are further reduced, with relevant information conserved and highlighted, particularly for the sequential structure of the data. Each encoder is completed by two successive fully connected layers, of dimensions 15 and 10 respectively, the first one having a 0.5 dropout rate.

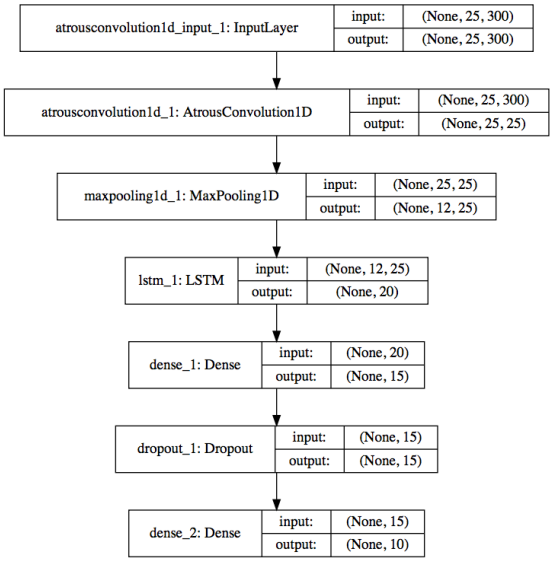

Figure 1: Example of an encoder. Input is passed to a CNN, a max pooling layer, an LSTM RNN, and finally two fully connected layers, the first having a dropout rate of .5. The input's and output's shape is indicated in brackets for each layer

Each sentence is thus transformed to a 10 dimensional vector. To perform the final comparison, these two low dimensional vectors are passed to a layer that merges them into a single vector. We tried several ways of merging the encoders' outputs, and we found that simple vector concatenation was the best option. We produce a 20 dimensional two-sentence vector as the final output of the DNN.

We do not apply any special mechanism for "comparison" or "alignment" in this phase. To measure the similarity of two sequences our model makes use only of the information contained in the merged vector that the encoders produce. We did not use a device in the merging phase to assess similarity between the two sequences. This allows a high degree of freedom in the interpretation patterns we are trying to model, but it also involves a fair amount of noise, which increases the risk of error.

The merging layer feeds the concatenated input to a final fully connected layer. The last layer applies a sigmoid function to produce the judgments. The advantage of using a sigmoid function in this case is that, while it performs well for binary classification, it returns a gradient over its input, thus generating an ordering of values appropriate for the ranking task. The combination of these three kinds of Neural Networks in this order (CNN, LSTM RNN and fully connected layers) has been explored in other works, with interesting results (Sainath et al., 2015). This research has indicated that these architectures can complement each other in complex semantic tasks, such as sentiment analysis (Wang et al., 2016) and text representation (Vosoughi et al., 2016).

The fundamental idea here is that these three kinds of Neural Network capture information in different ways that can be combined to achieve a better global representation of sentence input. While a CNN can reduce the spectral variance of input, an LSTM RNN is designed to model its sequential temporal dimension. At the same time, an LSTM RNN's performance can be strongly improved by providing it with better features (Pascanu et al., 2014), such as the ones produced by a CNN, as happens in our case. The densely connected layers contribute a clearer, more separable final vector representation of one sentence.

To encode the original sentences we used Word2Vec embeddings pre-trained on the very large Google News dataset (Mikolov et al., 2013). We used these embeddings to create the input sequences for our model.

We take as a baseline for evaluating our model the cosine similarity of the sentence vectors, obtained through combining their respective pre-trained lexical embeddings. This baseline gives very low accuracy and F1 scores.

4 Binary Classification Task

As discussed above, our corpus can be applied to model two sub-problems: binary classification and paraphrase ordering.

To use our corpus for a binary classification task

Model	Accuracy	F1
Baseline (cosine similarity)	50.8	10.1
Our model	**75.2**	74.6
Encoders without LSTM	64.4	64.9
Encoders without ACNN	62.6	61.5
Using CNN instead of ACNN	61.0	61.6
ACNN with 10 filters	73.4	71.7
LSTM with 10 filters	72.3	70.6
Merging via multiplication	53.4	69.6
Aligner	49.4	61.6
Aligner + our model	73.4	75.

Table 1: Accuracy for different versions of the model, and the baseline. Each version ran on our standard train and test data, without performing cross-validation. We use as a baseline the cosine similarity between the mean of the word vectors composing each sentence.

we map each set of five sentences into a series of pairs, where the first element is the metaphor we want to interpret and the second element is one of its four literal candidates.

Gradient labels are then replaced by binary ones. We consider all labels higher than *2* as positive judgments (Paraphrase) and all labels less than or equal to *2* as negative judgments (Non-Paraphrase), reflecting the ranking discussed in Section 2. We train our model with these labels for a binary metaphor paraphrase detection task.

Keeping the order of the input fixed (we will discuss this issue below), we ran the training phase for 15 epochs.

We reached an average accuracy of 67% for 12 fold cross-validation.

Interestingly, when trained on the pre-defined training set only, our model reaches the higher accuracy of 75%.

We strongly suspect that this discrepancy in performance is due to the small training and test sets created by the partitions of the 12 fold cross validation process.

In general, this task is particularly hard, both because of the complexity of the semantic properties involved in accurate paraphrase (see 4.1), and the limited size of the training set. It seems to us that an average accuracy of 67% on a 12 fold partitioning of training and test sets is a reasonable result, given the size of our corpus.

We observe that our architecture learned to recognize different semantic phenomena related to metaphor interpretation with a promising level of accuracy, but such phenomena need to be represented in the training set.

In light of the fact that previous work in this field is concerned with single verb paraphrase ranking (Bollegala and Shutova, 2013), where the metaphorical element is explicitly identified, and the candidates don't contain any syntactic-semantic expansion, our results are encouraging.[3]

Although a small corpus may cause instability in results, our DNN seems able to generalize with relative consistency on the following patterns:

- **Sentiment**. *My life in California was a nightmare – My life in California was terrible*. Our system seems able to discriminate the right sentiment polarity of a metaphor by picking the right paraphrase, even when some candidates contain sentiment words of opposite polarity, which are usually very similar in a distributional space

- **Non metaphorical word re-use**. Our system seems able, in several cases, to discriminate the correct paraphrase for a metaphor, even when some candidates re-use the words of the metaphor to convey a (wrong) literal meaning. *My life in California was a dream – I lived in California and had a dream*

- **Cases of multi-word metaphor** Although well represented in our corpus, multi-word metaphors are in some respects the most difficult to correctly paraphrase, since the interpretation has to be extended to a number of words. Nonetheless, our model was able to correctly handle these in a number of situations. *You can plant the seeds of anger – You can act in a way that will engender rage*

However, our model had trouble with several others cases.

It seems to have particular difficulty in discriminating sentiment intensity, with assignment of higher scores to paraphrases that value the sentiment intensity of the metaphor, which creates problems in several instances. Also, cases of metaphoric exaggeration (*My roommate is a sport maniac – My roommate is a sport person*), negation (*My roommate was not an eagle – My roommate was dumb.*) and syntactic inversions pose difficulties for our models.

We found that our model is able to abstract over specific patterns, but, predictably, it has difficulty in learning when the semantic focus of an interpretation consists in a phrase that is under represented in the training data.

[3] It should be noted that Bollegala and Shutova (2013) employ an unsupervised approach.

In some cases, the effect of data scarcity can be observed in an "overfit weighting" of specific terms. Some words that were seen in the data only once are associated with a high or low score independently of their context, degrading the overall performance of the model. We believe that these idiosyncrasies, can be overcome through training on a larger data set.

4.1 The gray areas of interpretation

We observe that, on occasion, the model's errors fall into a gray area between clear paraphrase and clear non-paraphrase. Here the correctness of a label is not obvious.

These cases are particularly important in metaphor paraphrasing, since this task requires an interpretative leap from the metaphor to its literal equivalent. For example, the pair *I was home watching the days slip by from my window – I was home thinking about the time I was wasting* can be considered as a loose paraphrase pair. Alternatively, it can be regarded as a case of non-paraphrase, since the second element introduces some interpretative elements (*I was thinking about the time*) that are not in the original.

In our test set we labeled it as 3 (loose paraphrase), but if our system fails to label it correctly in a binary task, it is not entirely clear that it is making an error. For these cases, the approach presented in the next section is particularly useful.

5 Paraphrase Ordering Task

The high degree of correlation we found between the AMT annotations and our single annotator's judgments indicate that we can use this dataset for an ordering task as well. Since the human judgments we collected about the "degree of paraphrasehood" are quite consistent, it is reasonable to pursue a non-binary approach.

Once the DNN has learned representations for binary classification, we can apply it to rank the sentences of the test set in order of similarity.

We apply the sigmoid value distribution for the candidate sentences in a set of five (the reference and four candidates) to determine the ranking.

To do this we use the original structure of our dataset, composed of sets of five sentences. First, we assign a similarity score to all pairs of sentences (reference sentence and candidate paraphrase) in a set. This is the similarity score learned in the binary task, so it is determined by the sigmoid function applied on the output.

The following is an example of an ordered set with strong correlation between the model's predictions (marked in bold) and our annotations (given in italics)

- The candidate is a fox
  - **0.13** *1* The candidate owns a fox
  - **0.30** *2* The candidate is stupid
  - **0.41** *3* The candidate is intelligent
  - **0.64** *4* The candidate is a cunning person

We compute the average Pearson and Spearman correlations on all sets of the test corpus, to check the extent to which the ranking that our DNN produces matches our mean crowd source human annotations.

While Pearson correlation measures the relationship between two continuous variables, Spearman correlation evaluates the monotonic relation between two variables, continuous or ordinal.

Since the first of our variables, the model's judgment, is continuous, while the second one, the human labels, is ordinal, both measures are of interest.

We found comparable and meaningful correlations between mean AMT rankings and the ordering that our model predicts, on both metrics. On the balanced training and test set, we achieve an average Pearson correlation of 0.75 and an average Spearman correlation of 0.68. On a twelve fold cross-validation frame, we achieve an average Pearson correlation of 0.55 and an average Spearman correlation of 0.54. We chose a twelve fold cross-validation because it is the smallest partition we can use to get meaningful results. We conjecture that the average cross fold validation performance is lower because of the small size of the training data in each fold. These results are displayed in Table 2.[4]

These correlations indicate that our model achieves an encouraging level of accuracy in predicting our gradient annotations for the candidate sentences in a set when trained for a binary classification task.

This task differs from the binary classification task in several important respects. In one way,

[4] As discussed above, the upper bound for our model's performance can be set at 0.9, the correlation between our single annotator's and the mean crowd sourced judgments.

it is easier. A non-paraphrase can be misjudged as a paraphrase and still appear in the right order within a ranking. In another sense, it is more difficult. Strict paraphrases, loose paraphrases, and various kinds of semantically similar non-paraphrases have to be ordered in accord with human judgment patterns, which is a more complex task than simple binary classification.

We should consider to what extent this task is different from a multi-class categorization problem. Broadly, multi-class categorization requires a system for linking a pair of sentences to a specific class of similarity. This is dependent upon the classes defined by the annotator and presented in the training phase. In several cases determining these ranked categories might be problematic. A class corresponding to our label "3", for example, could contain many different phenomena related to metaphor paraphrase: expansions, reformulations, reduction in the expressivity of the sentence, or particular interpretations of the metaphor's meaning. Our way of formulating the ordering task allows us to overcome this problem. A paraphrase containing an expansion and a paraphrase involving some information loss, both labeled as "3", might have quite different scoring, but they still fall between all "2" elements and all "4" elements in a ranking.

We can see that our gradient ranking system provides a more nuanced view of the paraphrase relation than a binary classification.

Consider the following example:

- My life in California was a dream
  - **0.03** *1* I had a dream once
  - **0.05** *2* While living in California I had a dream
  - **0.11** *3* My life in California was nice, I enjoyed it
  - **0.58** *4* My life in California was absolutely great

The human annotators consider the pair **My life in California was a dream** – *My life in California was nice, I enjoyed it* as loose paraphrases, while the model scored it very low. But the difference in sentiment intensity between the metaphor and the literal candidate renders the semantic relation between the two sentences less than perspicuous. Such intensity is instead present in **My life in California was absolutely great**, marked as a more valid paraphrase (score 4).

Measure	12-fold value	Baseline
Accuracy	67	51
Pearson correlation	0.553	0.151
Spearman correlation	0.545	0.113

Table 2: Accuracy and ranking correlation for Twelve Fold Cross-Validation. It can be seen that the simple cosine similarity between the mean vectors of the two sentences, which we use as baseline, returns a low correlation with human judgments.

On the other hand, it is clear that in the choice between *While living in California I had a dream* and *My life in California was nice, I enjoyed it*, the latter is a more reasonable interpretation of the metaphor.

The annotators relative mean ranking has been sustained by our model, even if its absolute scoring involves an error in binary classification.

The correlation between AMT annotation ordering and our model's predictions is a by-product of supervised binary learning. Since we are reusing the predictions of a binary classification task, we consider it a form of transfer learning from a supervised binary context to an unsupervised ordering task. In this case, our corpus allows us to perform double transfer learning. First, we used pretrained word embeddings trained to maximize single words' contextual similarity, in order to train on a supervised binary paraphrase dataset. Then, we use the representations acquired in this way to perform an ordering task for which the DNN had not been trained.

The fact that ranked correlations are sustained through binary paraphrase classification is not an obvious result. In principle, a model trained on $\{0,1\}$ labels could "polarize" its scores to the point where no meaningful ordering would be available. Had this happened, a good performance in a binary task would actually conceal the loss of important semantic information. The fact that there is no necessary connection between binary classification and prediction of gradient labels, and that an increase in one can even produce a loss in the other, is pointed out in Xu et al. (2015), who discuss the relation of paraphrase identification to the recognition of semantic similarity.

6 The Nature of the Metaphor Interpretation Task

Although this task resembles a particular case of paraphrase detection, in many respects it is something different. While paraphrase detection concerns learning content identity or strong cases of semantic similarity, our task involves the interpretation of figurative language.

In a traditional paraphrase task, we should maintain that "The candidate is a fox" and "The candidate is cunning" are invalid paraphrases. First, the superficial informational content of the two sentences is different. Second, without further context we might assume that the candidate is an actual fox. We ignore the context of the phrase.

In this task the frame is different. We assume that the first sentence contains a metaphor. We summarize this task by the following question.

Given that X is a metaphor, which one of the given candidates would be its best literal interpretation?

We trained our model to move along a similar learning pattern. This training frame can produce the apparent, but false paradox that two acceptable paraphrases such as *The Council is on fire* and *The Council is burning* are assigned a low score by our model. If the first element is a metaphor, the second element is, in fact, a bad literal interpretation. A higher score is correctly assigned to the candidate *People in the Council are very excited.*

7 Conclusions

We present a new kind of corpus to evaluate metaphor paraphrase detection, following the approach presented in Bizzoni and Lappin (2017) for paraphrase grading, and we construct a novel type of DNN architecture for a set of metaphor interpretation tasks. We show that our model learns an effective representation of sentences, starting from the distributional representations of their words. Using word embeddings trained on very large corpora proved to be a fruitful strategy. Our model is able to retrieve from the original semantic spaces not only the primary meaning or denotation of words, but also some of the more subtle semantic aspects involved in the metaphorical use of terms.

We based our corpus' design on the view that paraphrase ranking is a useful way to approach the metaphor interpretation problem.

We show how this kind of corpus can be used for both supervised learning of binary classification, and for gradient judgment prediction.

The neural network architecture that we propose encodes each sentence in a 10 dimensional vector representation, combining a CNN, an LSTM RNN, and two densely connected neural layers. The two input representations are merged through concatenation and fed to a series of densely connected layers.

We show that such an architecture is able, to an extent, to learn metaphor-to-literal paraphrase.

While binary classification is learned in the training phase, it yields a robust correlation in the ordering task through the softmax sigmoid distributions generated for binary classification. The model learns to classify a sentence as a valid or invalid literal interpretation of a given metaphor, and it retains enough information to assign a gradient value to sets of sentences in a way that correlates with our crowd source annotation.

Our model doesn't use any "alignment" of the data. The encoders' representations are simply concatenated. This gives our DNN considerable flexibility in modeling interpretation patterns. It can also create complications where a simple alignment of two sentences might suffice to identify a similarity. We have considered several possible alternative versions of this model to tackle this issue.

In future we will expand the size and variety of our corpus. We will perform a detailed error analysis of our model's predictions, and we will further explore different kinds of neural network designs for paraphrase detection and ordering. Finally, we intend to study this task "the other way around" by detecting the most appropriate metaphor to paraphrase a literal reference sentence or phrase.

Acknowledgments

We are grateful to our colleagues in the Centre for Linguistic Theory and Studies in Probability (CLASP), FLoV, at the University of Gothenburg for useful discussion of some of the ideas presented in this paper, and to three anonymous reviewers for helpful comments on an earlier draft. The research reported here was done at CLASP, which is supported by a 10 year research grant (grant 2014-39) from the Swedish Research Council.

References

Rodrigo Agerri. 2008. Metaphor in textual entailment. In *COLING 2008, 22nd International Conference on Computational Linguistics, Posters Proceedings, 18-22 August 2008, Manchester, UK*. pages 3–6. http://www.aclweb.org/anthology/C08-2001.

Eneko Agirre, Carmen Banea, Daniel M. Cer, Mona T. Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*. pages 497–511. http://aclweb.org/anthology/S/S16/S16-1081.pdf.

Yuri Bizzoni and Shalom Lappin. 2017. Deep learning of binary and gradient judgments for semantic paraphrase. *Proceedings of IWCS 2017* .

Danushka Bollegala and Ekaterina Shutova. 2013. Metaphor interpretation using paraphrases extracted from the web. *PloS one* 8(9):e74304.

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. 2016. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *CoRR* abs/1606.00915. http://arxiv.org/abs/1606.00915.

Jonathan Dunn, Jon Beitran De Heredia, Maura Burke, Lisa Gandy, Sergey Kanareykin, Oren Kapah, Matthew Taylor, Dell Hines, Ophir Frieder, David Grossman, et al. 2014. Language-independent ensemble approaches to metaphor identification. In *28th AAAI Conference on Artificial Intelligence, AAAI 2014*. AI Access Foundation.

Zornitsa Kozareva. 2015. Multilingual affect polarity and valence prediction in metaphors. In *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity,Sentiment and Social Media Analysis, WASSA@EMNLP 2015, 17 September2015, Lisbon, Portugal*. page 1. http://aclweb.org/anthology/W/W15/W15-2901.pdf.

Tina Krennmayr. 2015. What corpus linguistics can tell us about metaphor use in newspaper texts. *Journalism Studies* 16(4):530–546.

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. AAAI Press, AAAI'15, pages 2267–2273. http://dl.acm.org/citation.cfm?id=2886521.2886636.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, Curran Associates, Inc., pages 3111–3119.

Saif Mohammad, Ekaterina Shutova, and Peter D. Turney. 2016. Metaphor as a medium for emotion: An empirical study. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics, *SEM@ACL 2016, Berlin, Germany, 11-12 August 2016*. http://aclweb.org/anthology/S/S16/S16-2003.pdf.

Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. 2014. How to construct deep recurrent neural networks. *Proceedings of the Second International Conference on Learning Representations (ICLR 2014)* .

Tara N. Sainath, Oriol Vinyals, Andrew W. Senior, and Hasim Sak. 2015. Convolutional, long short-term memory, fully connected deep neural networks. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2015, South Brisbane, Queensland, Australia, April 19-24, 2015*. pages 4580–4584. https://doi.org/10.1109/ICASSP.2015.7178838.

Ekaterina Shutova. 2010. Automatic metaphor interpretation as a paraphrasing task. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, Stroudsburg, PA, USA, HLT '10, pages 1029–1037. http://dl.acm.org/citation.cfm?id=1857999.1858145.

Ekaterina Shutova and Simone Teufel. 2010. Metaphor corpus annotated for source-target domain mappings. In *LREC*. volume 2, pages 2–2.

Richard Socher, Eric H. Huang, Jeffrey Pennington, Andrew Y. Ng, and Christopher D. Manning+. 2011. Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection. In *Advances in Neural Information Processing Systems 24*.

Peter D. Turney. 2013. Distributional semantics beyond words: Supervised learning of analogy and paraphrase. *CoRR* abs/1310.5042. http://arxiv.org/abs/1310.5042.

Tony Veale, Ekaterina Shutova, and Beata Beigman Klebanov. 2016. *Metaphor: A Computational Perspective*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers. https://doi.org/10.2200/S00694ED1V01Y201601HLT031.

Soroush Vosoughi, Prashanth Vijayaraghavan, and Deb Roy. 2016. Tweet2vec: Learning tweet embeddings

using character-level cnn-lstm encoder-decoder. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, New York, NY, USA, SIGIR '16, pages 1041–1044. https://doi.org/10.1145/2911451.2914762.

Jin Wang, Liang-Chih Yu, K. Robert Lai, and Xuejie Zhang. 2016. Dimensional sentiment analysis using a regional CNN-LSTM model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*. http://aclweb.org/anthology/P/P16/P16-2037.pdf.

Wei Xu, Chris Callison-Burch, and Bill Dolan. 2015. Semeval-2015 task 1: Paraphrase and semantic similarity in twitter (pit). In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*. Association for Computational Linguistics, Denver, Colorado, pages 1–11. http://www.aclweb.org/anthology/S15-2001.