Some Preliminary Results on Analogies Between Sentences Using Contextual and Non-Contextual Embeddings

Stergos Afantenos, Thomas Barbero

Some Preliminary Results on Analogies Between Sentences Using Contextual and Non-Contextual Embeddings

Thomas Barbero[1], Stergos Afantenos[2]

[1]IRIT, University of Toulouse, France

Abstract

Analogies have been characterized as fundamental to abstraction, concept formation, and perception, and are traditionally expressed as quadruplets in the form of proportional analogies $a : b :: c : d$ read "a is to b as c is to d". While Natural Language Processing (NLP) has primarily focused on word analogies and SAT problems, recent research has started exploring analogies between sentences and even documents. In this paper we explore the potential of identifying analogies between pairs of sentences via the identification of common latent relations between them. We exploit three different datasets generating pairs of sentences which can either share the same latent relation—forming thus an analogy—or not. We encode phrases into a higher dimensional vector space using embeddings from GloVe, BERT, and RoBERTa which we then feed to both a Multi Layer Perceptron (MLP) and a Convolutional Neural Network (CNN). Results show that architectures using contextual embeddings as inputs outperform those based on static embeddings.

1. Introduction

Analogies have preoccupied humanity at least since antiquity [1]. In recent years they have been characterized as being at "the core of cognition" [2] and have even been considered as being the fundamental mechanism via which abstraction, concept formation and perception are achieved [3, 4].

Traditionally analogies have been expressed as a quadruplet $a : b :: c : d$ read "a is to b as c is to d". Such quadruplets then form valid analogies if pairs (a, b) and (c, d) share the same underlying relation, forming thus a *proportional analogy*. The underlying relation has been viewed as the symbolic counterpart of arithmetic or geometric proportions: $a - b = c - d$ and $\frac{a}{b} = \frac{c}{d}$ respectively[1] [5].

In Natural Language Processing (NLP) various approaches adopt the framework of quadruplets focusing mostly on word analogies, such as *man is to woman as king is to queen* [6, 7, 8, 9], morphology [10] or on SAT problems [11]. More recently several researchers have focused on the problem of identifying analogies between sentences [12, 13, 14] or even documents [15].

IARML@IJCAI'2023: Workshop on the Interactions between Analogical Reasoning and Machine Learning, at IJCAI'2023, August, 2023, Macao, China

*This work was performed while the first author was working at IRIT, University of Toulouse, France. The first author is the corresponding author.

[1]Note though that any kind of latent relation can be used in order to form a valid analogy. For example, $2 : 4 :: 3 : 9$ is a valid analogy because $2 : 2^2 :: 3 : 3^2$

In this paper we are interested in further exploring the potential of analogies between sentences via the identification of common latent relations between them. We exploit three different datasets, namely the Microsoft Research Paraphrases Corpus (MSRP) [16], the Penn Discourse TreeBank (PDTB) [17] as well as the Stanford Natural Language Inference (SNLI) corpus [18] and we use GloVe [19], or transformer-based architectures such as BERT [20] and RoBERTa [21] for the encoding of phrases into a higher dimensional vector space. We show that architectures that are based on contextual embeddings outperform ones that are based on static embeddings.

The rest of the paper is structured as follows. In Section 2 we present the related work. In Section 3 we present the datasets that we have used in order to perform our experiments. The methodology for these experiments is described in Section 4 while the results are presented in Section 5. We conclude in Section 6.

2. Related Work

Initial work on analogies in NLP was performed by [11] who introduced *Latent Relational Analysis (LRA)* in order to identify analogies in the context of the Scholastic Aptitude Test (SAT), testing this approach in 20 scientific and metaphorical examples.

More recently Mikolov et al. [22, 23] have used analogies as a means to test the quality of static vectors representing word embeddings produced with *word2vec* for use in neural architectures. The authors showed that such embeddings could preserve the parallelogram rule that is found in analogies, evaluating thus the intrinsic qualities of such embeddings. Later work though has shown that this is not sufficient since most models appear to take shortcuts; no evidence exists of abstraction and analogical mapping, as one would expect from such claims. More precisely, [24] show the Google analogy test set that we used by [22, 23] is not well balanced and thus does not allow us to draw any safe conclusions concerning the underlying embeddings. They show that the vector offset approach is not enough to claim that the proposed method captures analogies. The authors thus introduce the Bigger Analogy Test Set (BATS). In this more sophisticate dataset the authors show that derivational and lexicographic relations remain a challenge. Similar conclusions are drawn by [25] both for the vector offset approach as well as the 3CosAdd [26]. They argue that such datasets cannot be used to evaluate the intrinsic qualities of such datasets.

In terms of word analogy classification [27] used the Google dataset [6] which they extended using permutation properties of analogies, presented in the same article. They then apply a Convolutional Neural Network using as input Glove embeddings representing each word. A similar approach was also adopted by [10] in the context of detecting morphological analogies. We also adopt this approach in this paper.

Recently, several researchers have explored sentential analogies. [12] explore analogies between sentences in order to identify D from a predefined set of possible candidates, given (A, B) and C such that $A : B :: C : D$ is a valid analogy. They use syntactic and semantic datasets and test various embedding methods. In a similar vein [28] perform a similar task but generate D instead. Both approaches show that analogies based on syntactic analogies obtain better results than semantic ones. [13, 14] explore sentential analogies based purely on semantic

information.

In another approach, [15] view analogies via the prism of the *Structure Mapping Theory* [29]. Their goal is to identify analogies in procedural texts focusing on the structural similarities between the texts. Underlying texts describe procedures in two different domains. The authors extract entities and their relationships. The latter are sets of ordered verbs. They extract those based on question answer pairs. The similarity measures that they propose reflect the fact that the two sets share more relations. Bert vectors representing the questions via which entities were extracted, are used in order to measure cosine similarity and thus identify potential mappings.

3. Data used

In order to perform our experiments we used three well known datasets. In what follows we provide a detailed description of the corpora used as well as the procedure which lead us to the creation of analogical quadruplets that were later used in our experiments. We should mention that we used the input datasets as they were released, no further additions or modifications were performed from us.

3.1. Paraphrases

The first corpus that we used was the Microsoft Research Paraphrases Corpus (MSRP) [16] which is composed of 5801 pairs of sentences labeled as paraphrase or not. The pairs are distilled from a database containing more than 13 million sentences pairs, itself extracted from a more than 9 million sentences corpus [16]. The 9M sentences corpus is composed of sentences extracted from +32k news clusters from internet. This corpus then has been largely reduced to contain sentences with a credited author only, leaving 49375 individual sentence pairs. So this corpus is composed of naturally occurring, non handcrafted sentences pairs. Sentences pairs with minimal variations such as typography error have been removed as they could have constituted "low quality" paraphrases.

A Support Vector Machine-Classifier (SVM-Classifier) is then used to identify a set of possible paraphrases from the 49375 sentences pairs. This set is validated by human annotators later. The SVM-Classifier is trained on a 10000 sentences pairs training set annotated by 2 human judges, and a 3rd who served the function of judge in case of disagreement. The distribution of this training set is 2968 positives examples and 7032 negatives. The classifier considered multiples features: string similarity, morphological variants, synonyms mapping with WordNet Lexical Mapping and Encarta Thesaurus, and finally composite features. The SVM-Classifier allowed to extract 20574 sentences pairs as possible paraphrases from the 4959375 previously considered. The number is high because the classifier's role was to separate possible sentences pairs to be evaluated by human judgment and not discriminate all non-paraphrases pairs, so the classifiers tend to classify inputs as positive rather than negative, at the assumed cost of having more false-positives.

Human judgment was applied to a 5801 subset of the 20574 previously extracted sentence pairs. Two judges annotated each sentence and a third one was used in case of disagreement. Each judge was asked if the pairs' sentences were semantically equivalent. About 3900 (67%) of the sentences pairs were labeled as semantically equivalent.

3.2. PDTB

The second corpus that we used was the Penn Discourse TreeBank (PDTB) [17] corpus which contains discourse annotations between sentences clauses extracted from the Wall Street Journal Corpus containing over 1 million words. The corpus describes a total of 36592 relations [17]. Discourse annotations can be triggered by an explicit or implicit discourse connective. The former are extracted from syntactically defined classes and are separated in 3 grammatical classes subordinating conjunctions, coordinating conjunctions and discourse adverbs. Explicit connectives can be connected to more than 1 clause or sentence, but the minimality principle is applied which requires minimum information to complete the interpretation. In the case of an implicit connection between the two clauses the annotators have been instructed to insert an explicit connective. Three other labels were available in order to correctly annotate three cases that prevented the annotators from inserting a coherent explicit connective. The AltLex indicating that the relation was already explicited by a non-connective expression, the insertion of a connective would then lead to a redundancy. The entRel indicating the existence of an entity based coherence relation between the two clauses, but no other relation. And finally noRel in case of no relation between the two clauses. PDTB relations are ordered hierarchically into class, type and subtype. For our experiments we used the first level of the hierarchy.

The inter-annotator agreement was high: 90,2% for explicit relations and 85,1% for implicit when exact match metric was considered; and respectively 94,5% and 92,6% when partial match metric was considered. Class level disagreement was resolved by a team of 3 experts, disagreement at lower levels were resolved by providing a tag for the direct higher level. Agreement for the class level reached 94%, 84% for type level and 80% for subtype level.

3.3. SNLI

The Stanford Natural Language Inference (SNLI) corpus [18] labels pairs of sentences as Contradiction, Entailment or semantic neutrality [18]. It contains 570k pairs of sentences by humans. Construction of the corpus was done using Mechanical Turks who were presented with a premise in the form of a sentence and were asked to provide three hypotheses, in a sentential form, for contradiction, entailment and semantic similarity. 10% of the corpus was validated by trusted Mechanical Turks. Overall a Fleiss κ of 0.70 was achieved.

The indeterminacies of event and entity co-reference are two well known issues during labeling of NLI data degrading the quality of the annotated corpus. They represent respectively a possible confusion between an Entailment and a Neutral relation, and between a Contradiction and a Neutral relation. This confusion comes from the fact that an assumption may or not have been made.

In order to solve this problem the annotation process was made in a grounded scenario aiming to reduce assumptions. Annotators were then able to generate sentences in the same scenario in order to illustrate the relations instead of relying on automatic data augmentation techniques. The work of 2500 employees permitted the data collection phase. When presented with an image caption without the matching image, the annotators had to write three sentences, one for each relation (the exact instructions are described in the SNLI paper). The image captions came from the Flickr30k corpus containing 160k captions from 30k individuals images. The

validation phase is completed on 10% of the 570k pairs of sentences by a set of 30 trusted workers. They were presented pairs of sentences and had to label them, each pair being presented to 4 annotators so there is 5 judgments considering the label from the data collection phase. The gold-label has been assigned to the pairs with at least a 3-annotators consensus, representing 98% of the data. The corpus is then separated in three individual files : test and dev (10k pairs each), train (the rest of the pairs).

3.4. Generation of analogical quadruplets

In order to create our analogical quadruplets we proceeded as follows. For each of the aforementioned datasets we randomly selected two pairs of sentences each one linked with a relation. Since our input datasets do not contain relations that have as arguments the same sentences, we never have analogies of the form $a : a :: b : b$. In case the relation linking the two pairs is the same we have a positive instance of an analogy otherwise a negative instance. For the SNLI corpus we considered neutral as not being a relation. For each input dataset we create a balanced training, test and development datasets containing the same number of positive and negative instances. Training consists of 400K instances while testing and development 40K instances each.

4. Methodology

Our problem can be formalized as follows. Given a set of quadruplets of sentences $a : b :: c : d$ which can either form an analogy (pairs $a : b$ and $c : d$ share the same latent relation) or not we need to estimate a function that predicts whether a new instance of four sentences is an analogy or not. Each quadruplet is represented by the input tokens of its sentences $s = \{w_1^s, \ldots, w_{|s|}^s\}$ with $s \in \{a, b, c, d\}$ and $|s|$ representing the length of the sentence. With each quadruplet we associate a $y \in \{0, 1\}$ which represents whether the quadruplet is an analogy or not. For each quadruplet we obtain embeddings using GloVe [19], BERT [20] and RoBERTa [21] which we then pass to two different architectures, a Multi-layer perceptron (MLP) and a Convolutional Neural Network (CNN).[2]

4.1. Embeddings

In order to perform classification we need to provide embeddings for each sentence. In the case of GloVe[3] static embeddings are provided for each word, while in the case of BERT[4] and RoBERTa[5] embeddings are dynamic. In order to obtain embeddings that represent sentences from the ones representing words a common approach [20, for example] is to take the mean of the embeddings representing each word. This is the approach that we have used as well. For

[2]Our code is available at https://github.com/ThomasBARBERO/EXPLO_ANALOGIE
[3]The Glove embeddings that we used are the following: https://huggingface.co/sentence-transformers/average_word_embeddings_glove.6B.300d
[4]https://huggingface.co/bert-base-uncased
[5]https://huggingface.co/roberta-base

GloVe	300
BERT	768
RoBERTa	1024

Table 1
Embedding dimensions of different encoders.

each sentence $s \in \{a, b, c, d\}$ we obtain an embedding

$$\mathbf{s} = \frac{1}{|s|} \sum_{w \in s} \texttt{emb}(w)$$

with $\texttt{emb} \in \{\texttt{glove}, \texttt{bert}, \texttt{roberta}\}$. Thus four different embeddings $\mathbf{a}, \mathbf{b}, \mathbf{c}$ and $\mathbf{d}$ are obtained for each of the sentences a, b, c and d. In the case of BERT we have also examined the use of the representation obtained for the final hidden state of the special symbol [CLS]. Embedding dimensions for each method are shown in Table 1. No further fine-tuning was performed on BERT or RoBERTa.

4.2. Classifiers

Multi-layer perceptron (MLP) The first classifier that we use is a multi-layer perceptron. The MLP takes as input the concatenation of the representations for the four sentences $\mathbf{a}, \mathbf{b}, \mathbf{c}$ and $\mathbf{d}$ as a vector $[\mathbf{a}; \mathbf{b}; \mathbf{c}; \mathbf{d}]$ and has two hidden layers, the first has a dimension of 100 and the second of 50.

$$\mathbf{z_1} = \mathbf{W_1}^T[\mathbf{a}; \mathbf{b}; \mathbf{c}; \mathbf{d}] + \mathbf{b_1}$$
$$\mathbf{z_2} = \mathbf{W_2}^T\mathbf{z_1} + \mathbf{b_2}$$

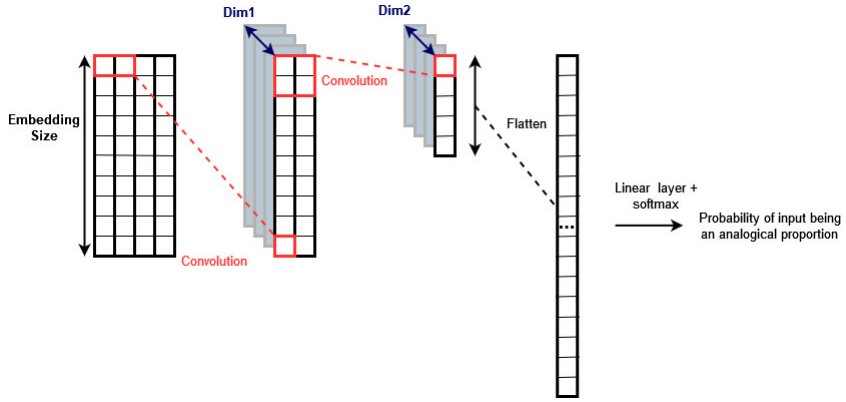
with $\mathbf{W_1}, \mathbf{b_1}, \mathbf{W_2}, \mathbf{b_2}$ learnable matrices. The output layer is of dimension 1 and we use a sigmoïd function providing a score for the final prediction

$$\hat{y} = \frac{1}{1 + e^{-\mathbf{z_2}}}$$

Convolutional Neural Network (CNN) The second classifier architecture that we have used are Convolutional Neural Networks which are widely used for image and audio processing, but are useful for Natural Language Processing tasks too as well, including analogies [27, 10, *inter alia*]. CNNs aim to recognize patterns, extracting features from the initial tensor given as input. The core of CNNs is their convolutional layers applying filters called kernels on the whole input. Kernels' weights and bias are learnt parameters, convolution between a kernel and a tensor allows to extract a learnt feature from the tensor. Parameters define the method for the application of kernels: the kernels' size, the stride which indicates the spatial distance between two kernel applications and padding which indicates the number of pixels to add on the considered tensor's borders. Our CNN's implementation is as follows, illustrated in Fig. 1:

1. The input goes through a first convolutional layer with 2×1 kernels and 2×1 stride allowing to firstly get feature maps for the pairs (a, b) and (c, d). At the end of this process (a, b) and (c, d) are reduced to one dimension regarding the width while the other dimension represents the embedding size. The size of the output is $2 \times EMBEDDING\_SIZE$.
2. We fed the output to a second convolutionnal layer with 2×2 kernels and 2×2 stride so the features maps of (a, b) and (c, d) are now reunited in one single dimension across width, the embedding size is divided by 2 too.
3. We then apply dropout and feed the output to a singular linear layer, we use sigmoid activation function to compute a confidence score for the 2 sentence pairs being in an analogical proportion relation.

Figure 1: Diagram representing the CNN architecture, each Convolutional layer is followed by a ReLU layer. Dropout is applied before linear layer. $Dim_1 = Embedding\_size * 2$ is the number of feature maps after passing through the first convolutional layer, $Dim_2 = Embedding\_size/2$ is the number of feature maps after passing through the second convolutional layer. The flatten operation outputs a tensor of size Dim_2^2.



5. Experiments and Results

For both architectures we ranged learning rate between 10^{-4} and 10^{-5}, and dropout from 0.1 to 0.3. We used Adam optimizer with default PyTorch settings and Binary Cross Entropy Loss. Results for both architectures and combinations of embeddings are shown in Table 2.

5.1. Transformer-based Language Models vs GloVe

Transformer-based Language Models outperform GloVe almost constantly in terms of accuracy and F1-score (ability to recognize valid analogical proportions). While the scores are not significantly higher, we can still conclude that contextual embeddings provide better handling of latent relations and analogies between sentences in comparison to static embeddings. Let us note also that representing a sentence by the mean of its contextual word vectors outperforms the CLS sentence representation.

		Precision	Recall	F1	Accuracy
	PDTB				
GloVe-mean	class 1	59.35	53.473	56.259	53.855
	class 0	48.36	54.331	51.172	
BERT-base-mean	class 1	59.39	55.847	**57.564**	56.218
	class 0	53.045	56.639	54.783	
BERT-base-CLS	class 1	56.02	56.132	56.076	56.12
	class 0	56.22	56.108	56.164	
roBERTa-base-mean	class 1	49.835	57.364	53.335	**56.398**
	class 0	62.96	55.655	59.083	
roBERTa-base-CLS	class 1	42.995	55.256	48.361	54.09
	class 0	65.185	53.347	58.675	
	SNLI				
GloVe-mean	class 1	64.17	62.346	63.245	62.708
	class 0	61.245	63.09	62.154	
BERT-base-mean	class 1	70.215	64.32	67.138	65.633
	class 0	61.05	67.21	63.982	
BERT-base-CLS	class 1	71.065	62.654	66.595	64.353
	class 0	57.64	66.578	61.787	
roBERTA-base-mean	class 1	70.315	64.474	**67.268**	**65.785**
	class 0	61.255	67.358	64.162	
roBERTa-base-CLS	class 1	70.27	61.713	65.714	63.338
	class 0	56.405	65.484	60.607	
	MRPC				
GloVe-mean	class 1	60.43	65.066	**62.662**	63.992
	class 0	67.555	63.062	65.231	
BERT-base-mean	class 1	51.03	65.473	57.356	62.06
	class 0	73.09	59.88	65.829	
BERT-base-CLS	class 1	42.94	62.2	50.806	58.422
	class 0	73.905	56.431	63.997	
roBERTa-base-mean	class 1	56.35	66.372	60.952	**63.9**
	class 0	71.45	62.076	66.434	
roBERTa-base-CLS	class 1	50.995	61.307	55.677	59.405
	class 0	67.815	58.051	62.554	

(a) Results for CNN

		Precision	Recall	F1	Accuracy
	PDTB				
GloVe-mean	class 1	39.39	55.296	46.007	53.773
	class 0	68.155	52.93	59.585	
BERT-base-mean	class 1	48.705	57.66	52.805	**56.47**
	class 0	64.235	55.6	59.607	
BERT-base-CLS	class 1	50.285	56.662	**53.284**	55.913
	class 0	61.54	55.314	58.261	
roBERTa-base-mean	class 1	45.41	56.873	50.499	55.487
	class 0	65.565	54.567	59.563	
roBERTa-base-CLS	class 1	48.66	56.297	52.2	55.442
	class 0	62.225	54.792	58.273	
	SNLI				
GloVe-mean	class 1	68.96	62.224	65.419	63.547
	class 0	58.135	65.192	61.462	
BERT-base-mean	class 1	69.635	66.681	68.126	67.42
	class 0	65.205	68.227	66.682	
BERT-base-CLS	class 1	65.72	64.787	65.25	65.0
	class 0	64.28	65.219	64.746	
roBERTa-base-mean	class 1	68.515	67.887	**68.2**	**68.053**
	class 0	67.59	68.221	67.904	
roBERTa-base-CLS	class 1	73.18	61.496	66.831	63.68
	class 0	54.18	66.889	59.867	
	MRPC				
GloVe-mean	class 1	44.605	60.167	51.23	57.537
	class 0	70.47	55.989	62.4	
BERT-base-mean	class 1	57.285	58.757	58.012	58.537
	class 0	59.79	58.329	59.05	
BERT-base-CLS	class 1	58.99	57.771	58.374	57.935
	class 0	56.88	58.106	57.486	
roBERTa-base-mean	class 1	59.375	58.222	58.793	58.385
	class 0	57.395	58.554	57.969	
roBERTa-base-CLS	class 1	60.565	59.117	**59.832**	**59.34**
	class 0	58.115	59.575	58.836	

(b) Results for MLP

Table 2
Results

5.2. Performance across corpora

Overall scores for the SNLI dataset are the highest with accuracy ranging from 62.708 to 68.01 and F1-score peaking at 68.2 across MLP and CNN. Scores for MRPC are a bit lower with accuracy ranging from 57.537 to 63.992 and F1-score peaking at 62.662 considering CNN only as it constantly outperforms MLP. The classifiers had a harder time grasping analogies on the PDTB corpus with accuracy ranging from 53.773 to 56.47 and F1-score peaking at 57.564, F1-score being below 50 for roBERTa-base-CLS/CNN and GloVe-mean/MLP. The classifiers had a harder time grasping analogies on the PDTB corpus with accuracy ranging from 53.773 to 56.47 and F1-score peaking at 57.564, F1-score being below 50 for roBERTa-base-CLS/CNN and GloVe-mean/MLP. This can be explained by the fact that the number of latent relations that we had to handle in PDTB is much higher (5 latent relations) than the latent relations that we have in the MRPC or the SNLI corpora. We assume that providing more data will yield better overall results.

5.3. BERT vs roBERTa

One main difference between BERT and roBERTa is respectively the presence and absence of the Next sentence prediction training task. While BERT considered this task to be beneficial for the learning of long range dependencies roBERTa considered this task counter-productive.

Although roBERTa performs slightly better than BERT (considering the mean-pooling sentence representation method) we cannot draw a definitive conclusion about the utility of the Next Sentence Prediction training task for analogical properties learning. A bigger training set may have enforced the tendency. roBERTa outperformed BERT for SNLI and MRPC, the two corpora for which the sentences from the sentence pairs do not follow each other in a natural context. The next sentence prediction may be detrimental in this case.

5.4. CNN vs MLP

Both MLP and CNN are relevant for the classification task we performed as they have almost similar results with the CNNs usually outperforming the MLPs, but not always. However the MRPC's results show a large difference in performance between the 2 classifiers, the CNNs performing significantly better than the MLPs. Meaningful features were extracted from the sentences representations. As we described Section 4 features are first extracted from the (a, b) pair and the (c, d) pair in tandem. This could probably be attributed due to teh fact that paraphrases use semantically similar words which probably are closer to the vector space which is better captured by CNNs than MLPs, although further analysis is needed in order for this claim to be verified.

6. Conclusions and Future Work

In this paper we have focused on the problem of identifying analogies between pairs of sentences based on common latent relations that exist or not between the pairs. We have used both contextual embeddings (BERT en roBERTa) as well as static embeddings (GloVe). Both BERT and roBERTa outperformed GloVe at the binary classification task we performed. We believe an error analysis or a different classification task might shed more light on those results. In conclusion this work scratches the surface of Transformer-based Language Models' ability to encode analogical properties. Our experiments show that embeddings issued from Transformer-based architectures can better capture analogies via the identification of common latent relations, in comparison to static embedding approaches. Nonetheless it is premature to conclude that such architectures can indeed capture more broadly the mechanism of analogy making.

Acknowledgments

The authors would like to thank the anonymous reviewers for their thoughtful and constructive comments. This work has been partially funded by the ANR AT2TA project, grant number ANR-22-CE23-0023.

References

[1] Aristotle, Poetics, 384—322 BCE.

[2] D. R. Hofstadter, Analogy as the Core of Cognition, in: D. Gentner, K. J. Holyoak, B. N. Kokinov (Eds.), The Analogical Mind: Perspectives from Cognitive Science, The MIT Press, Cambridge, Massachusetts, 2001, pp. 499–538.

[3] D. Hofstadter, E. Sander, Surfaces and Essences: Analogy as the Fuel and Fire of Thinking, Basic Books, 2013.

[4] F. Chollet, On the measure of intelligence, 2019. arXiv:1911.01547.

[5] N. Barbot, L. Miclet, H. Prade, Analogy between concepts, Artificial Intelligence 275 (2019) 487–539.

[6] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: C. J. C. B. et al. (Ed.), Advances in Neural Information Processing Systems 26, Curran Associates Inc., 2013, pp. 3111–3119.

[7] T. Mikolov, W.-t. Yih, G. Zweig, Linguistic regularities in continuous space word representations, in: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Atlanta, Georgia, 2013, pp. 746–751. URL: https://aclanthology.org/N13-1090.

[8] S. Lim, H. Prade, G. Richard, Classifying and completing word analogies by machine learning, International Journal of Approximate Reasoning 132 (2021) 1–25. URL: https://www.sciencedirect.com/science/article/pii/S0888613X21000141. doi:https://doi.org/10.1016/j.ijar.2021.02.002.

[9] S. Lim, H. Prade, G. Richard, Solving word analogies: A machine learning perspective, in: European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty, volume 11726, Springer, 2019, pp. 238–250. URL: https://doi.org/10.1007/978-3-030-29765-7_20. doi:10.1007/978-3-030-29765-7_20.

[10] S. Alsaidi, A. Decker, P. Lay, E. Marquer, P.-A. Murena, M. Couceiro, A neural approach for detecting morphological analogies, in: 2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA), 2021, pp. 1–10. doi:10.1109/DSAA53316.2021.9564186.

[11] P. D. Turney, The Latent Relation Mapping Engine: Algorithm and Experiments, Journal of Artificial Intelligence Research 33 (2008) 615–655.

[12] X. Zhu, G. de Melo, Sentence analogies: Linguistic regularities in sentence embeddings, in: Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 3389–3400. URL: https://aclanthology.org/2020.coling-main.300. doi:10.18653/v1/2020.coling-main.300.

[13] S. D. Afantenos, T. Kunze, S. Lim, H. Prade, G. Richard, Analogies between sentences: Theoretical aspects - preliminary experiments, in: J. Vejnarová, N. Wilson (Eds.), Symbolic and Quantitative Approaches to Reasoning with Uncertainty - 16th European Conference, ECSQARU 2021, Prague, Czech Republic, September 21-24, 2021, Proceedings, volume 12897 of *Lecture Notes in Computer Science*, Springer, 2021, pp. 3–18. URL: https://doi.org/10.1007/978-3-030-86772-0_1. doi:10.1007/978-3-030-86772-0_1.

[14] S. D. Afantenos, S. Lim, H. Prade, G. Richard, Theoretical study and empirical investigation of sentence analogies, in: M. Couceiro, P. Murena (Eds.), Proceedings of the Workshop on the Interactions between Analogical Reasoning and Machine Learning (International Joint

Conference on Artificial Intelligence - European Conference on Artificial Intelligence (IJAI-ECAI 2022)), Vienna, Austria, July 23, 2022, volume 3174 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 15–28. URL: http://ceur-ws.org/Vol-3174/paper2.pdf.

[15] O. Sultan, D. Shahaf, Life is a circus and we are the clowns: Automatically finding analogies between situations and processes, in: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2022, pp. 3547–3562. URL: https://aclanthology.org/2022.emnlp-main.232.

[16] W. B. Dolan, C. Brockett, Automatically constructing a corpus of sentential paraphrases, in: Proceedings of the Third International Workshop on Paraphrasing (IWP2005), 2005. URL: https://aclanthology.org/I05-5002.

[17] R. Prasad, N. Dinesh, A. Lee, E. Miltsakaki, L. Robaldo, A. Joshi, B. Webber, The Penn Discourse TreeBank 2.0., in: Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), European Language Resources Association (ELRA), Marrakech, Morocco, 2008. URL: http://www.lrec-conf.org/proceedings/lrec2008/pdf/754_paper.pdf.

[18] S. R. Bowman, G. Angeli, C. Potts, C. D. Manning, A large annotated corpus for learning natural language inference, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, 2015.

[19] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1532–1543. URL: http://www.aclweb.org/anthology/D14-1162.

[20] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, 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, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423. doi:10.18653/v1/N19-1423.

[21] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, 2019. URL: http://arxiv.org/abs/1907.11692, cite arxiv:1907.11692.

[22] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in Neural Information Processing Systems, volume 26, Curran Associates Inc., 2013, pp. 3111–3119. URL: https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html.

[23] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, in: International Conference on Learning Representations, Workshop, 2013. URL: https://arxiv.org/abs/1301.3781. doi:10.48550/ARXIV.1301.3781.

[24] A. Gladkova, A. Drozd, S. Matsuoka, Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't., in: North American Chapter of the Association for Computational Linguistics, Student Research Workshop, Association for Computational Linguistics, San Diego, California, 2016, pp.

8–15. URL: https://aclanthology.org/N16-2002. doi:10.18653/v1/N16-2002.

[25] A. Rogers, A. Drozd, B. Li, The (too many) problems of analogical reasoning with word vectors, in: Joint Conference on Lexical and Computational Semantics, Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 135–148. URL: https://aclanthology.org/S17-1017. doi:10.18653/v1/S17-1017.

[26] O. Levy, Y. Goldberg, Linguistic regularities in sparse and explicit word representations, in: Proceedings of the Eighteenth Conference on Computational Natural Language Learning, Association for Computational Linguistics, Ann Arbor, Michigan, 2014, pp. 171–180. URL: https://aclanthology.org/W14-1618. doi:10.3115/v1/W14-1618.

[27] S. Lim, H. Prade, G. Richard, Solving word analogies: A machine learning perspective, in: Proc.15th Europ. Conf. Symb. & Quantit. Appr. to Reas. with Uncert. (ECSQARU), LNCS 11726, 238-250, Springer, 2019.

[28] L. Wang, Y. Lepage, Vector-to-sequence models for sentence analogies, in: International Conference on Advanced Computer Science and Information Systems, 2020, pp. 441–446. doi:10.1109/ICACSIS51025.2020.9263191.

[29] D. Gentner, Structure Mapping: A Theoretical Framework for Analogy, Cognitive Science 7 (1983) 155–170. URL: https://doi.org/10.1207/s15516709cog0702_3. doi:10.1207/s15516709cog0702_3.