

Transformer-based hierarchical attention models for solving analogy puzzles between longer, lexically richer and semantically more diverse sentences

Benming YAN¹, Haotong WANG¹, Liyan WANG¹, Yifei ZHOU¹ and Yves LEPAGE^{1,*}

¹Waseda University, 2-7 Hibikino, Wakamatsu-ku, Kitakyushu-shi, 808-0123, Japan

Abstract

We address the task of solving analogy puzzles between sentences. Various possible neural architectures are envisageable for this task. However the scarcity of datasets of sentence analogies to assess the efficiency of such proposals is a problem. Our contribution is thus twofold. We firstly propose a method to build a resource of sentence analogies that uses sentence templates and leverages word analogy datasets. Back translation is used to produce paraphrases. Our method could produce a resource in English that is shown to be richer along three dimensions compared with a previously publicly available resource: our sentences are longer, their vocabulary is richer and their semantics is more varied. Secondly, to solve analogy puzzles between sentences, we introduce an architecture that makes use of a hierarchical attention mechanism in the Transformer model in conformity to properties of analogy puzzles. Our experiments show superior results compared to a baseline and other possible architectures on the newly created dataset.

Keywords

Sentence analogy puzzles, Datasets, Transformer, Attention

1. Introduction

With the advent of vector representations of language pieces in natural language processing (NLP), analogies have made a come-back. The over-repeated *male : female :: king : queen* has now long been known to every practitioner of NLP [1]. Solving analogy puzzles between words, like *male : female :: king : x ⇒ x = queen*, has been made possible, on the assumption that words in analogy constitute a parallelogram in embedding spaces, a hypothesis supported by the theoretical result that negative sampling skip-gram models are equivalent to a factorisation of a PPMI matrix of co-occurrences [2]. In order to assess the quality of various word embedding models, attempts at building various word analogy datasets have been made [1, 3, 4, 5].

This paper addresses the problem of solving sentence analogy puzzles, hence beyond

IARML@IJCAI'2024: Workshop on the Interactions between Analogical Reasoning and Machine Learning, at IJCAI'2024, August, 2024, Jeju, South Korea


*Corresponding author.

✉ nathan_yan@toki.waseda.jp (B. YAN); wanghaotong0925@toki.waseda.jp (H. WANG);
wangliyan0905@toki.waseda.jp (L. WANG); yifei.zhou@ruri.waseda.jp (Y. ZHOU); yves.lepage@waseda.jp.com
(Y. LEPAGE)

🌐 <http://lepage-lab.ips.waseda.ac.jp/> (Y. LEPAGE)

🆔 0000-0002-3059-4271 (Y. LEPAGE)

© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

word analogy puzzles. For example: $I \text{ talk to her} . : He \text{ talked to her} . :: I \text{ go to the park} . : x \Rightarrow x = He \text{ went to the park} .$ is an example of solving a simple sentence analogy puzzle that relies on elementary word analogy puzzles.

Analogies have properties that have been agreed upon since Antiquity and that have been reiterated in more recent works [6, 7, 8]. In particular, equivalent forms are in the number of eight:

$$\begin{array}{cccc} A : B :: C : D & B : A :: D : C & C : A :: D : B & \boxed{D : B :: C : A} \\ \boxed{A : C :: B : D} & B : D :: A : C & C : D :: A : B & D : C :: B : A \end{array}$$

From the viewpoint of solving analogy puzzles, by letting the unknown D at the same position, there is only one equivalent form, which was called *permutando* in Latin, or *permutation of the means*.¹ But in addition, it should be noticed that A does not play the same role as the means B and C relatively to the unknown D . As a matter of fact, in an analogy, A and D can be permuted thanks to the property of *permutation of the extremes*. The above two properties are indicated by frames in the list of eight equivalent forms above.

2. Previous dataset and newly created dataset

Datasets of sentence analogies are almost inexistant. The ACL pages relate to word analogies, not to sentence analogies². The Penn Discourse Treebank, which annotates sentence pairs for discourse relations, is not *stricto sensu* a sentence analogy dataset [9] because the exchange of the means is not verified in general [10, 11]. The same holds for the ANALOGICAL dataset that is also a set of pairs of sentences, not strictly speaking a set of sentence analogies [12].

2.1. Previous dataset

The only known sentence analogy dataset has been introduced in [13].³ It consists of 5,607 semantico-formal sentence analogies in English (see Table 1), *i.e.*, analogies which are partly formal (the sentences are almost fixed patterns) and partly semantic (word analogies are involved). The second example in Table 1 is an example of such analogies: it interleaves two syntactic patterns $I \text{ do not [like this N / feel like Ving]} .$ with a semantic word analogy $song : singing :: game : playing$. This dataset has been used as a benchmark to assess various methods for solving analogy puzzles between sentences in [15], including the use of compact language models like SBERT.

However, a thorough investigation of this dataset reveals flaws. Nearly half of the analogies fall within the category of formal analogy (first row in Table 1), while true semantico-formal analogies constitute a significantly smaller proportion (second and third row). Additionally, it contains many unacceptable sentence analogies because the analogies have been produced automatically by solving word analogy puzzles blindly. *E.g.*, high frequent words, like, *e.g.*, *girlfriend* (2489 occurrences) or *boyfriend* (1674 occurrences) were found in analogy with

¹In an analogy $A : B :: C : D$, B and C are the means and A and D are the extremes.

²[https://aclweb.org/aclwiki/Analogy_\(State_of_the_art\)](https://aclweb.org/aclwiki/Analogy_(State_of_the_art))

³The dataset has been extended to French in [14].

Example analogy				# of such analogies			
Hurry up , or you 'll miss the bus .	:	Hurry up , or you will miss the bus .	::	Hurry up , or you 'll be late .	:	Hurry up , or you will be late .	2,362
I do not like this song .	:	I do not feel like singing .	::	I do not like this game .	:	I do not feel like playing .	303
I had a dreadful dream last night .	:	I had a horrible dream last night .	::	I had a strange dream last night .	:	I had a weird dream last night .	
i do not need a nap .	:	i do not have a beard .	::	i do not need a boyfriend .	:	i do not have a girlfriend .	2,942
Total							5,607

Table 1

Classification of analogies found in the previous analogy dataset. The last number is the number of analogies of this type. The first row is purely formal analogies; the second and third rows are for analogies that might be considered acceptable semantico-formal analogies (possibly after some slight change); the last one is for unacceptable analogies.

unrelated words like, *i.e.*, *nap* and *beard*. This is caused by automatically solving analogy puzzles in the word embedding model used, and then selecting the closest sentence from Tatoeba to ensure correctness, but such analogies are unacceptable for the human eye. The last row in Table 1 is such an example

A statistical analysis of word and sentence lengths in the dataset shows that, on average, sentences consist of 7.1 words and span 26 characters. This points at the relative simplicity of the sentences in the dataset, its limited vocabulary and supposedly a limited semantic variety.

2.2. Newly created dataset

We address the flaws in the previous dataset and produce a sentence analogy dataset characterised by longer sentences, richer vocabulary and richer semantic content. For that, we leverage the Google Analogy Test Set [1]⁴ and the Tatoeba corpus⁵.

The Google Analogy Test Set encompasses various word analogy types of relation, from world-knowledge (capital : country) and semantic relations (male : female) to grammatical ones (adjective : opposite), as illustrated in Table 2. For each type of relation, the dataset offers a list of analogies, *i.e.*, a pair of ratios. We remember the list of all word pairs for each type of relation.

We retrieve any sentence from the Tatoeba corpus that contains a word in two of the types of analogies capitals-common-countries and family of the Google Analogy Test Set (sentence A in Figure 1). We filter out too short sentences. We apply back-translation to all retrieved sentences, *i.e.*, we translate into a language other than English (here we translate into Chinese) and keep the sentences that are different from the original ones. The purpose of back-translation is to introduce textual or syntactic variation while maintaining semantic similarity. Recent research

⁴<http://download.tensorflow.org/data/questions-words.txt>

⁵<https://tatoeba.org/en/>

Type of relation	Example analogy						
capital-common-countries	Athens	:	Greece	::	Bangkok	:	Thailand
	Beijing	:	China	::	London	:	England
currency	Algeria	:	dinar	::	Europe	:	euro
	Canada	:	dollar	::	Thailand	:	baht
city-in-state	Chicago	:	Illinois	::	Houston	:	Texas
	Huntsville	:	Alabama	::	Boston	:	Massachusetts
family	grandfather	:	grandmother	::	prince	:	princess
	husband	:	wife	::	stepfather	:	stepmother
gram2-opposite	competitive	:	uncompetitive	::	informed	:	uninformed
	known	:	unknown	::	tasteful	:	distasteful

Table 2

Some example analogies from the Google analogy test set with the type of relation indicated on the left

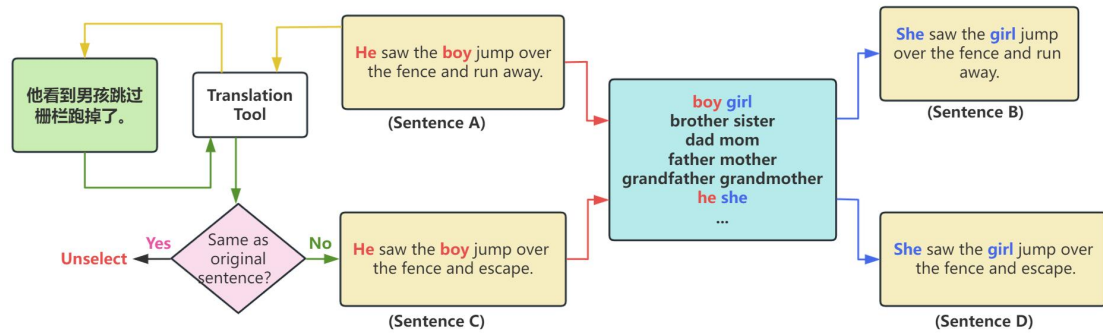


Figure 1: Production of a sentence analogy by relying on word ratios. We start with sentence A that contains words from a given list of word ratios with the same type of relation. We get sentence C by back-translation, and sentences B and D by replacing words using word ratios from the given list.

points at the fact that understanding syntactic structure might be crucial in understanding sentence analogies [16]. This provides us with pairs of sentences that we consider as sentence ratios (sentences A and C in Figure 1). We then revisit the set of word pairs in the same type of relation and replace the words in the original and back-translated sentences with corresponding words from other word pairs from the same type of relation. If this is not possible, we discard the sentence ratio. This creates two new sentences (sentences B and D in Figure 1) that form an analogy with the two previous sentences.

The entire process described above and illustrated in Figure 1 allows us to successfully expand sentence ratios into sentence analogies. Table 3 gives some examples of analogies. They show that the semantic type of relation between the sentences in analogy remains consistent. Simultaneously, the process guarantees the formation of correct analogies, because one or two words only are exchanged in one direction, while back-translation produces variations in the other direction.

She cannot bear moonlight and she cannot bear darkness.	He cannot bear moonlight and he cannot bear darkness.	She couldn't bear the moonlight or the darkness.	He couldn't bear the moonlight or the darkness.
Your son is very bright, much more than his age.	Your daughter is very bright, much more than her age.	Your son is very intelligent, much older than his age.	Your daughter is very intelligent, much older than her age.
He plays at aerobics just to please his girlfriend.	She plays at aerobics just to please her boyfriend.	He only plays aerobics to please his girlfriend.	She only plays aerobics to please her boyfriend.

Table 3

Some example analogies in the newly created sentence analogy dataset

	Sentence analogy dataset		Increase
	Previous one [13]	Newly created	
Number of analogies	5,607	22,161	+295%
Sentence length			
Number of words/sentence	7.1	10.1	+42%
Number of characters/sentence	26.0	45.4	+75%
Lexical richness			
Vocabulary size	1,135	8,159	+619%
Semantic diversity			
Std. dev. of vector dispersion			
all-mpnet-base-v2	0.908	0.949	+4%
multi-qa-mpnet-base-dot-v1	4.094	5.228	+28%
all-distilroberta-v1	0.887	0.945	+7%
all-MiniLM-L12-v2	0.924	0.943	+7%

Table 4

Comparison between the previous analogy dataset and the new created one along several criteria. All criteria show an increase in favour of the newly created dataset.

2.3. Comparison of the two sentence analogy datasets

We compare the previous sentence analogy dataset and the newly created one, using several criteria. Table 4 details the numbers for this comparison.

Number of analogies. The newly created dataset contains more analogies with an increase by 295%, *i.e.*, approximately four times more analogies.

Sentence length. The new dataset achieves an increase in average sentence length over the previous dataset, with +42% in average number of words per sentence or +75% in average number of characters per sentence, *i.e.*, the sentences are longer by around a half.

Lexical richness. The vocabulary size shows a substantial increase in the newly created dataset: more than 7 times more words are used. This indicates a higher lexical richness.

Semantic diversity. To measure semantic diversity, we convert sentences into vector rep-

representations and, similarly to what is done in [17], we compute the standard deviations on each dimensions of all vectors from both datasets in four different embedding spaces [18]. We aggregate these standard deviation values by taking their norms (which is proportional to the quadratic mean). The greater the norm, the more dispersed the vectors in the sentence embedding spaces, hence, supposedly, the wider the semantic variety. The last rows in Table 4 give the results obtained by applying the same procedure to four different sentences representation models. They show an increase in semantic variety in all used embedding spaces.

To summarise the content of Table 4, the new dataset that we constructed demonstrates improvements along all mentioned criteria: in the newly created dataset, there are more analogies and the sentences are longer, lexically richer and semantically more diverse.

3. Previous models and proposed models

3.1. Previous models

The first sentence analogy dataset presented above has been built by merging two techniques: a formal approach to solve sequence analogy puzzles and a semantic approach to solve word analogy puzzles. New sentences have been generated by decomposing sentence analogies into word analogies [1] along edit distance traces between sentences [19]. This means that sentences were interpreted as sequences of vector representations of words.

Vec2Seq. It is also classical to represent a sentence by just one vector, the sum of the vector representations of its words. It is then possible to apply the classical arithmetic interpretation of analogy on such vector representations of sentences. *I.e.*, the sentence solution D of an analogy between sentences $A : B :: C : D$ is computed as the sentence corresponding to the vector $\vec{D} = \vec{B} - \vec{A} + \vec{C}$. This induces the necessity of a decoder that decodes the vector representation of \vec{D} into a word sequence that makes sentence D . In the Vec2Seq model proposed in [15], the architecture chosen for such a decoder is that of a recurrent network, namely a Long Short-Term Memory model. Although ameliorations can be given, as is done in [20], we will use the Vec2Seq model as our baseline.

3.2. Proposed models

On the contrary to the Vec2Seq model which concentrates learning on the decoder side, we propose several models that focus on the encoder side. As our work is more recent, we rely on Transformers rather than on recurrent networks. We aim to leverage the attention mechanism on the three given sentences in an analogy puzzle, *i.e.*, sentences A , B and C , in a manner that better exploits the properties in an analogy puzzle. We introduce several models that incorporate step-by-step the two properties seen in the Introduction: the permutation of the extremes and the permutation of the means.

A.B.C-to-D Transformer. A first naïve model to address the task of solving sentence analogy puzzles is to construe it as a sequence-to-sequence task where the input sequence is just made of the concatenation of the three given sentences A , B and C . The output sequence should be

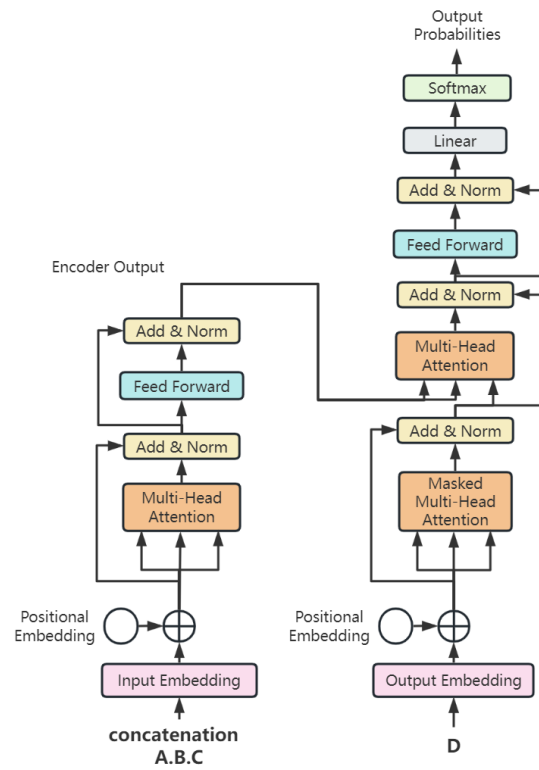


Figure 2: Architecture of the A.B.C-to-D Transformer model. It is a naïve use of the basic encoder-decoder Transformer model for solving analogy puzzles between sentences. The input of the encoder is the concatenation of A , B and C . This figure is just a slight modification of the figure in [21].

the fourth sentence D . A basic encoder-decoder Transformer model can be used for that as illustrated in Figure 2. This serves as a second, more robust, baseline in addition to the Vec2Seq model.

(B.C)xA-to-D Transformer. In an analogy puzzle, A plays a different role from B and C . This comes from the fact that, in an analogy, A and D can be exchanged by the permutation of the extremes. It pleads for a separation of A from B and C when solving an analogy puzzle.

Consequently, the second model we propose encodes B and C separately from A . The encoder side features two layers. The first layer comprises two self-attention blocks, one for the concatenation of sentences B and C , and another one for sentence A . The second layer consists of a cross-attention block. It takes the outputs of the two self-attention blocks from the first layer as inputs. The explanation for this cross-attention block is as follows.

As was the case in the basic encoder-decoder model where the attention block for D with respect to A , B and C was a cross-attention block, here too, on the encoder side, the attention block for A with respect to B and C should be of the same type, *i.e.*, a cross-attention block. As a result, on Figure 3, two cross-attention blocks are to be seen: one for A with respect to $B.C$ and another one for D with respect to $B.C$ and A (notice the same blue arrows for Key (K))

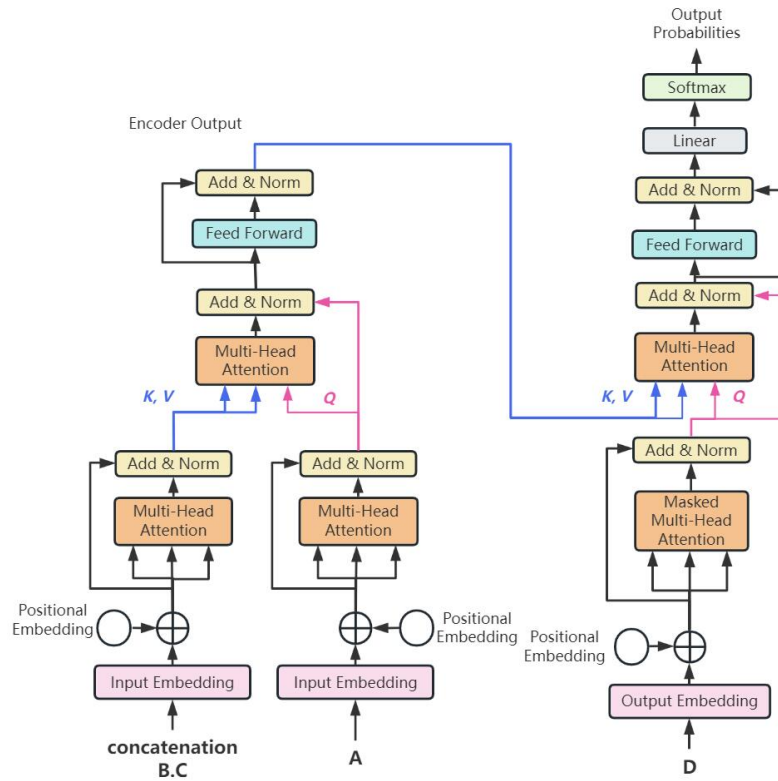


Figure 3: Architecture of the (B.C)xA-to-D Transformer model

and Value (V) and the same red arrows for Query (Q)).

(BxC)xA-to-D Transformer. Indeed, nothing prevents from inputting $C.B$ rather than $B.C$ in the previous (B.C)xA-to-D Transformer model because of the property of permutation of the means. We propose a model that copes with this property, i.e., B and C play the same role.

In this new model, as in the previous one, the encoder side features two layers. However, different from the previous model, the first layer of attention blocks provides two cross-attention blocks and one self-attention block, as illustrated in Figure 4. The first-layer cross-attention blocks each receive sentence B and sentence C as inputs and provide the required Key (K) and Value (V) for computing cross-attention one with another. Because of this double cross-attention mechanism, the positions of sentences B and C can be freely exchanged in conformity with the permutation of the means. Subsequently, the outputs of these two cross-attention blocks are combined (\oplus in Figure 4) and provide the Key (K) and Value (V) for the second-layer cross-attention block, where the Query (Q) comes from the self-attention block on input A in the first layer.

Summary on Transformer models (B.C)xA-to-D and (BxC)xA-to-D. In both models, the incorporation of cross-attention blocks, along with their respective Q , K , and V , conforms to the standard decoding side of the basic encoder-decoder Transformer model. In the decoder of

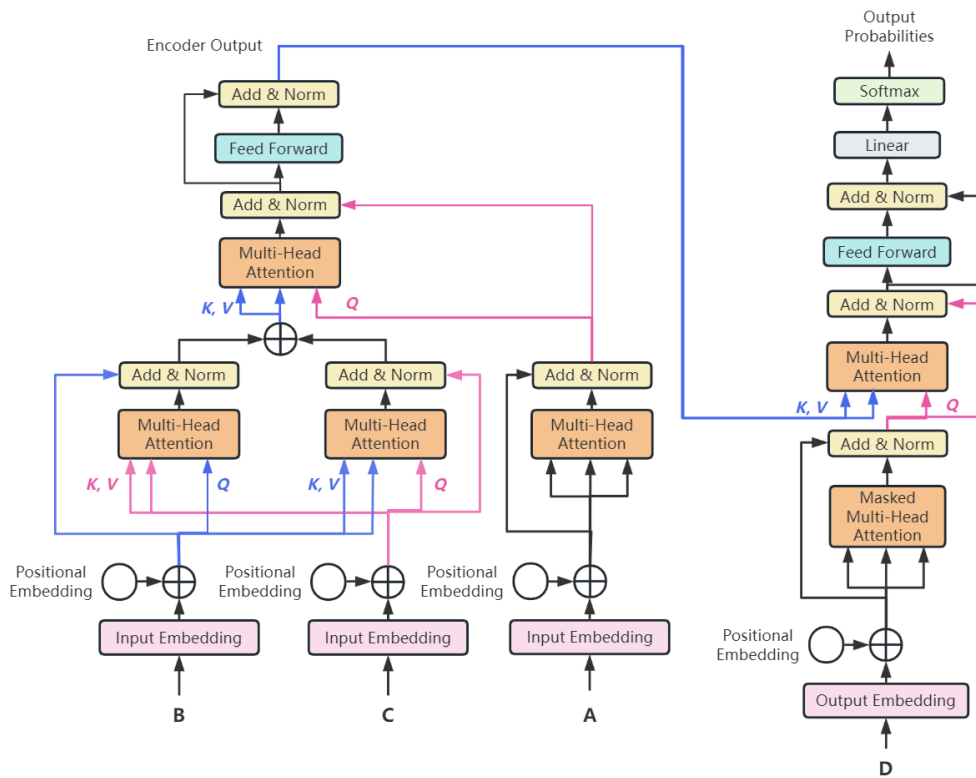


Figure 4: Architecture of the (BxC)xA-to-D Transformer model

a Transformer model, during training, the input encompasses both the target text itself and the output of the encoder. When performing cross-attention operations in the decoder, the K and V in the input data of the cross-attention block are derived from the encoder’s output, while Q originates from the result of self-attention on the target text, as depicted in the Figure 2. *I.e.*, the knowledge from the encoder guides the generation in the decoder. Thus, in both models, the second layer of the cross-attention block aims to enable sentence A to glean valuable knowledge from sentences B and C , so as to facilitate the generation of sentence D in the decoder.

4. Experiments

4.1. Data and models used

The two datasets described in Section 2, for which Table 4 provided statistics, are used in the experiments. They are cut into training, validation, and test sets with the classical proportions of 80%, 10%, and 10%, respectively. The tested models are the previously four described models with only one stack of Transformers. They are trained from scratch with early stopping with a batch size of 128 analogies. The training task is language modeling by next word completion.

Dataset	Size	Model	BLEU	Accuracy (%)	Levenshtein distance in words	Levenshtein distance in chars
Previous one	5,607	Recurrent model				
		Vec2Seq	90.02±1.89	82.2	0.4	1.4
		Transformer models				
		A.B.C-to-D	87.19±1.89	76.1	0.5	1.8
		(B.C)xA-to-D	82.64±2.49	70.1	0.7	2.6
1/2 Previous one + 1/2 Newly created	5,607	Recurrent model				
		Vec2Seq	56.58±3.02	26.7	2.2	10.4
		Transformer models				
		A.B.C-to-D	70.91±3.38	59.9	1.7	7.2
		(B.C)xA-to-D	60.77±3.49	47.8	2.4	9.5
Newly created	5,607	Recurrent model				
		Vec2Seq	52.89±2.72	12.3	3.0	14.1
		Transformer models				
		A.B.C-to-D	77.92±2.60	64.7	1.4	6.5
		(B.C)xA-to-D	64.04±3.64	59.0	2.8	11.0
Newly created	22,161	Recurrent model				
		Vec2Seq	61.17±1.41	27.6	2.5	11.8
		Transformer models				
		A.B.C-to-D	93.76±0.64	83.9	0.4	1.9
		(B.C)xA-to-D	80.37±1.25	67.2	1.3	5.6
		(BxC)xA-to-D	95.27±0.58	86.8	0.2	1.4

Table 5

Results of the experiments using the four models on the two datasets and more. The (BxC)xA-to-D Transformer model performs worse on the previous sentence analogy dataset. It performs best on the newly created dataset with longer, lexically richer and semantically more diverse sentences under two data size conditions and on a mixture of data from the two datasets.

4.2. Experiments and results

Comparing performance on the two datasets. The first round of experiments consists in comparing the performance of each model on the two datasets. The results are shown on the first and fourth groups of rows in Table 5 (Previous one (5,607) and Newly created (9,579)). A surprising phenomenon is observed: the trend over the four models are opposite on the two datasets. The recurrent network model Vec2Seq is the best on the previous dataset, but it achieves an accuracy of less than 30% on the newly created dataset.⁶ This suggests that this model, which relies on a simple aggregation of word vectors for sentence representation and employs a fixed formula for deriving vector representations of the solution sentence, faces difficulties in dealing with longer, lexically richer and semantically more diverse sentences.

The (BxC)xA-to-D Transformer model exhibits the poorest performance on the previous

⁶Accuracy is the percentage of output sentences that are exactly the same as the reference sentences, hence possibly a clue for overfitting.

analogy dataset. Nevertheless, its performance is not so low – a little bit less than 80 in BLEU. We find it hard to explain why it struggles to capture relevant information for sentence generation from the shorter and structurally simpler sentences within this dataset. We guess that this might come from the dataset itself.

Because half of the analogies in that dataset are formal ones and many are semantically hard to accept, the scores for the recurrent neural model might be explained by overfitting. This can be supported by the high accuracy (more than 80%, *i.e.*, 4 in 5) on this dataset, and the low accuracy scores on the datasets involving newly created data (below 30%).

The decrease in scores for our three proposed Transformer models on the previous dataset might be explained by an inability to handle unacceptable analogies. This is somewhat supported by the results on the newly created dataset, where the last Transformer model, which integrates the properties of permutation of the extremes and permutation of the means, and should thus have higher competency in identifying semantically corresponding parts in sentences thanks to cross-attention, surpasses all other models.

We also observe that, against our expectations, the decrease on the previous dataset and the increase on the newly created dataset are not gradual from the A.B.C-to-D model to the (BxC)xA-to-D model, with the (B.C)xA-to-D model exhibiting worse scores.

In order to test the hypothesis about the datasets themselves impacting the results, and in order to have a fairer comparison, we perform another round of experiments.

Comparing performance on equal size datasets. To study performance independently of the size of the data and neutralise the effect of the different sizes of the two datasets, we make the amount of training data equal in both conditions: we reduce the number of sentence analogies in the newly created dataset (originally 22,161) to match the quantity in the previous dataset (5,607). This is the third group of rows in Table 5 (Newly created (5,607)). Even on a reduced amount of data from the newly created dataset, the (BxC)xA-to-D Transformer model performs the best. We observe again that the ordering of the (B.C)xA-to-D and A.B.C-to-D models is contrary to our expectations.

To inspect the influence of the type of data contained in the two datasets, we construct yet another dataset that equally mixes data from both datasets. We build it to reach a number of 5,607 sentence analogies, *i.e.*, the size of the previous sentence analogy dataset. Half of the analogies in this new dataset are randomly sampled from each dataset. We expect results in-between the previously obtained results on equal size datasets. The second group of rows in Table 5 (1/2 Previous one + 1/2 Newly created (5,607)) does not follow our expectations: in BLEU, they are worse than on each dataset with equal size, except for the recurrent network model which is in-between. We are led to conclude that the data in the previous dataset have a strong impact on both types of models: a positive one on the recurrent network model and a negative one on the Transformer models. The ordering of Transformer models is the same as on the third and fourth groups of rows: (B.C)xA-to-D < A.B.C-to-D < (BxC)xA-to-D.

Behaviour of the (B.C)xA-to-D Transformer model As said above, the results of the (B.C)xA-to-D Transformer model are not in between the two other Transformer models; they are worse. In all experiments involving the newly created dataset, the Levenshtein distance in words or in characters is surprisingly much larger than in other Transformer models (and often

comparable to that of the recurrent model). In particular, an important decrease is observed on all metrics on the newly created dataset with the same size as the previous dataset. This questions the progressiveness of the integration of the two properties of analogy, and this might even question whether the way we integrate the permutation of the extremes is correct, calling for a better way of reflecting this property with the help of attention.

5. Conclusion

We addressed the problem of solving analogy puzzles between sentences, a task for which the scarcity of available datasets is problematic. We described an existing previous dataset, and, as a first contribution, created another dataset that corrects some of the weaknesses of the previous one. The newly created dataset contains more analogies and features longer sentences with a richer vocabulary and wider semantic variety in comparison to the previous dataset. The two datasets basically combine syntactic patterns with word analogies. The first dataset had the drawback of containing many formal analogies and semantically not so reliable analogies because it freely exploited word analogies in word embedding spaces. The newly created dataset introduces syntactic variations by relying on back translation to create paraphrases. In addition, by relying on the Google Analogy Test Set, the word analogies used are more constrained and therefore more reliable.

The second contribution of this paper is with models to solve sentence analogy puzzles. We introduced hierarchical attention models based on the encoder-decoder Transformer architecture to solve sentence analogy puzzles, *i.e.*, produce a fourth sentence that is in analogical relationship with three given sentences. We introduced two properties of analogy into our models, namely the permutation of the extremes and the permutation of the means, by making use of cross-attention. The resulting models adopt a hierarchical architecture. Our experiments showed that our proposed architectures, especially the last one, can address the challenges posed by analogies which involve longer, lexically richer and semantically more diverse sentences.

Still, our results call for further inspection. As for datasets, it would be interesting to examine datasets that would contain only formal analogies or that would contain only not so acceptable analogies. Experiments on these datasets could allow us to get more insight at why models are confused or whether they are more prone to overfitting on such datasets.

As for models, we introduced properties of analogy step-by-step, but our results did not exhibit a gradual increase in performance. The ordering of results for the (B.C)xA-to-D and A.B.C-to-D models is not the one we expected. This might call for the exploration of other models, *e.g.*, one of the type ((A.B)x(A.C))-to-D.

6. Acknowledgments

This work was supported in part by a grant from the Japanese Society for the Promotion of Science, Kiban C, n° 21K12038, entitled: “Theoretically founded algorithms for the automatic production of analogy tests in NLP”.

References

- [1] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, in: Proceedings of the International Conference on Learning Representations (ICLR), volume CoRR abs/1301.3781, 2013. URL: <https://arxiv.org/pdf/1301.3781.pdf>.
- [2] O. Levy, Y. Goldberg, Neural word embedding as implicit matrix factorization, in: Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, K. Weinberger (Eds.), Proceedings of Advances in Neural Information Processing Systems 27 (NIPS 2014), Curran Associates, Inc., 2014, pp. 2177–2185. URL: https://papers.nips.cc/paper_files/paper/2014.
- [3] A. Drozd, A. Gladkova, S. Matsuoka, Word embeddings, analogies, and machine learning: Beyond king - man + woman = queen, in: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, The COLING 2016 Organizing Committee, 2016, pp. 3519–3530. URL: <http://www.aclweb.org/anthology/C16-1332>.
- [4] A. Gladkova, A. Drozd, S. Matsuoka, Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't, in: Proceedings of the NAACL-HLT SRW, San Diego, California, 2016, pp. 47–54. doi:<https://www.aclweb.org/anthology/N/N16/N16-2002.pdf>.
- [5] M. Abdou, A. Kulmizev, V. Ravishankar, MGAD: Multilingual generation of analogy datasets, in: N. C. C. chair), K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, T. Tokunaga (Eds.), Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), European Language Resources Association (ELRA), Paris, France, 2018.
- [6] Y. Lepage, Languages of analogical strings, in: Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000), volume 1, Saarbrücken, 2000, pp. 488–494. URL: <https://aclanthology.org/C00-1071>.
- [7] N. Stroppa, Définitions et caractérisation de modèles à base d'analogies pour l'apprentissage automatique des langues naturelles, Thèse de doctorat, École nationale supérieure des télécommunications, 2005. URL: <https://hal.archives-ouvertes.fr/tel-00145147/>.
- [8] L. Miclet, H. Prade, Handling analogical proportions in classical logic and fuzzy logics settings, in: C. Sossai, G. Chemello (Eds.), Symbolic and Quantitative Approaches to Reasoning with Uncertainty, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 638–650.
- [9] R. Prasad, N. Dinesh, A. Lee, E. Miltsakaki, L. Robaldo, A. Joshi, B. Webber, The Penn Discourse TreeBank 2.0., in: N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, D. Tapias (Eds.), 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.
- [10] S. Afantenos, T. Kunze, S. Lim, H. Prade, G. Richard, Analogies between sentences: Theoretical aspects - preliminary experiments, in: J. Vejnárová, N. Wilson (Eds.), Symbolic and Quantitative Approaches to Reasoning with Uncertainty, Springer International Publishing, Cham, 2021, pp. 3–18.
- [11] S. Afantenos, S. Lim, H. Prade, G. Richard, Theoretical study and empirical investigation

- of sentence analogies, in: IJCAI-ECAI Workshop: Workshop on the Interactions between Analogical Reasoning and Machine Learning (IARML 2022)@ IJCAI-ECAI 2022, volume 3174, CEUR-WS. org, 2022, pp. 15–28.
- [12] T. Wijesiriwardene, R. Wickramarachchi, B. Gajera, S. Gowaikar, C. Gupta, A. Chadha, A. N. Reganti, A. Sheth, A. Das, ANALOGICAL - a novel benchmark for long text analogy evaluation in large language models, in: A. Rogers, J. Boyd-Graber, N. Okazaki (Eds.), Findings of the Association for Computational Linguistics: ACL 2023, Association for Computational Linguistics, Toronto, Canada, 2023, pp. 3534–3549. URL: <https://aclanthology.org/2023.findings-acl.218>. doi:10.18653/v1/2023.findings-acl.218.
- [13] Y. Lepage, Analogies between short sentences: A semantico-formal approach, in: Human Language Technology. Challenges for Computer Science and Linguistics: 9th Language and Technology Conference, LTC 2019, Poznan, Poland, May 17–19, 2019, Revised Selected Papers, Springer-Verlag, Berlin, Heidelberg, 2019, pp. 163–179. URL: https://doi.org/10.1007/978-3-031-05328-3_11. doi:10.1007/978-3-031-05328-3_11.
- [14] Y. Lepage, Human Language Technology – Challenges for Computer Science and Linguistics, number 13212 in Lecture Notes in Artificial Intelligence, Springer Nature Switzerland, 2022, pp. 163–179. URL: https://link.springer.com/content/pdf/10.1007%2F978-3-031-05328-3_11. doi:<https://doi.org/10.1007/978-3-031-05328-3>.
- [15] L. Wang, Y. Lepage, Vector-to-sequence models for sentence analogies, in: 2020 International Conference on Advanced Computer Science and Information Systems (ICACISIS), 2020, pp. 441–446. doi:10.1109/ICACISIS51025.2020.9263191.
- [16] T. Wijesiriwardene, R. Wickramarachchi, A. N. Reganti, V. Jain, A. Chadha, A. Sheth, A. Das, On the relationship between sentence analogy identification and sentence structure encoding in large language models, in: Y. Graham, M. Purver (Eds.), Findings of the Association for Computational Linguistics: EACL 2024, Association for Computational Linguistics, St. Julian's, Malta, 2024, pp. 451–457. URL: <https://aclanthology.org/2024.findings-eacl.31>.
- [17] W.-t. Yih, V. Qazvinian, Measuring word relatedness using heterogeneous vector space models, in: E. Fosler-Lussier, E. Riloff, S. Bangalore (Eds.), Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Montréal, Canada, 2012, pp. 616–620. URL: <https://aclanthology.org/N12-1077>.
- [18] N. Reimers, I. Gurevych, Sentence-BERT: Sentence embeddings using Siamese BERT-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3982–3992. URL: <https://aclanthology.org/D19-1410>. doi:10.18653/v1/D19-1410.
- [19] Y. Lepage, Solving analogies on words: an algorithm, in: Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL'98) and the 17th International Conference on Computational Linguistics (COLING'98), volume I, Association for Computational Linguistics, Montreal, 1998, pp. 728–735. URL: <https://www.aclweb.org/anthology/P98-1120/>. doi:10.3115/980845.980967.
- [20] W. Mao, Y. Lepage, Embedding-to-embedding method based on autoencoder for solv-

ing sentence analogies, in: Proceedings of the workshop Analogies: from Theory to Applications (ATA@ICCBR 2023), Aberdeen, Scotland, 2023, pp. 15–26.

- [21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, *Neural Information Processing Systems*, Neural Information Processing Systems (2017).