Sentence Analogies: Linguistic Regularities in Sentence Embeddings

Xunjie Zhu *
ByteDance
zhuxunjie@bytedance.com

Gerard de Melo
Hasso Plattner Institute, University of Potsdam
gdm@demelo.org

Abstract

While important properties of word vector representations have been studied extensively, far less is known about the properties of sentence vector representations. Word vectors are often evaluated by assessing to what degree they exhibit regularities with regard to relationships of the sort considered in word analogies. In this paper, we investigate to what extent commonly used sentence vector representation spaces as well reflect certain kinds of regularities. We propose a number of schemes to induce evaluation data, based on lexical analogy data as well as semantic relationships between sentences. Our experiments consider a wide range of sentence embedding methods, including ones based on BERT-style contextual embeddings. We find that different models differ substantially in their ability to reflect such regularities.

1 Introduction

Sentence embeddings are dense vectors that reflect salient semantic properties of a sentence. Similar to how commonly used word embedding methods such as word2vec (Mikolov et al., 2013a) capture semantic relationships between words, sentence embeddings are expected to encode semantic relationships between sentences. A number of different sentence embedding methods have been proposed (cf. Section 2.1 for an overview). In recent years, pretrained language models such as ELMo (Peters et al., 2018), BERT (Devlin et al., 2019), XLNet (Yang et al., 2019), and RoBERTa (Liu et al., 2019) have become the method of choice when encoding text. Thus, such models are also often invoked to represent sentences by means of individual embeddings.[1]

While important properties of word vector representations have been studied extensively, far less is known about the properties of sentence vector representations. A particularly prominent aspect of word vector representations induced by methods such as word2vec is that the vector space exhibits certain kinds of regularities. Many of these are of the sort considered in word analogies. Proportional analogies take the form A is to B as C is to D, e.g., *Paris* is to *France* as *Berlin* is to *Germany*. Rumelhart and Abrahamson (1973) first proposed identifying such analogies using vector representations in a Euclidean space. Given vector representations of concepts, derived from human similarity judgments using a multi-dimensional scaling algorithm, they proposed representing analogical relationships in terms of the difference vectors. Turney and Littman (2005) investigated identifying such analogies using bag-of-words vector space models. Mikolov et al. (2013b) showed that word2vec's word vector representations reflect certain kinds of word analogies surprisingly well. The widely used word analogy task that they proposed takes the following form. Given embeddings $\vec{v}_A$, $\vec{v}_B$, $\vec{v}_C$, $\vec{v}_D$ for words A, B, C, D for an analogy of the above form, the task consists in identifying the correct word D given A, B, and C. Most commonly, this is

*The work was done when Xunjie was a master's student at Rutgers University

[1]Consider e.g. https://github.com/hanxiao/bert-as-service and the work of Reimers et al. (Reimers and Gurevych, 2019).

Proceedings of the 28th International Conference on Computational Linguistics, pages 3389–3400
Barcelona, Spain (Online), December 8-13, 2020

achieved by optimizing

$$\begin{aligned} \underset{D \in V}{\text{argmax}} \quad & \text{sim}(\vec{v}_D, \vec{v}_B - \vec{v}_A + \vec{v}_C) \\ \textit{s.t.} \quad & D \notin \{A, B, C\}, \end{aligned} \tag{1}$$

where $\text{sim}(\vec{v}_1, \vec{v}_2)$ typically denotes cosine similarity between two vectors. This sort of analogy task is one of the most commonly invoked means of assessing the quality of word vector induction techniques.

However, little is known about the topology of vector representation spaces for entire sentences. In this paper we fill this gap, considering models with a dedicated sentence embedding objective as well as BERT-style pretrained embedding models. We study whether such sentence representation spaces as well exhibit regularities with regard to certain kinds of relationships. To this end, we devise new datasets that are similar to typical word analogy datasets (Mikolov et al., 2013b). These allow us to empirically assess whether existing sentence embedding models reflect analogical relationships between sentences.

2 Background and Related Work

2.1 Sentence Embedding Methods

In order to move from word vector representations towards representations for entire sentences, a simple baseline is to simply average the word embeddings of all words in a sentence. Although this method neglects the order of words, it performs surprisingly well in many downstream tasks. Pagliardini et al. (2017) proposed a method to learn word and n-gram embeddings such that the average of all words and n-grams in a sentence can serve as a high-quality sentence vector. Rücklé et al. (2018) improved the average pooling method by concatenating different power means of word embeddings. Almarwani et al. (2019) proposed the use of a Discrete Cosine Transform (DCT) to compress word vectors into sentence embeddings, while retaining word order information.

Several methods have been proposed to directly learn representations of sentences. The Skip-Thought Vector approach (Kiros et al., 2015), inspired by the skip-gram word2vec approach (Mikolov et al., 2013a), attempts to learn representations that enable the prediction of neighbouring sentences. It relies on an encoder–decoder structure based on Gated Recurrent Units. The Quick-Thought Vector approach (Logeswaran and Lee, 2018) improves both the efficiency and performance of Skip-Thought Vectors by replacing the decoder with a simple classifier that selects the correct sentence among a set of candidates. InferSent (Conneau et al., 2017) learns sentence representations by auxiliary supervised learning on Natural Language Inference (NLI) data, outperforming prior methods on tasks that require detailed semantic understanding (Zhu et al., 2018). Subramanian et al. (2018) proposed methods to learn general purpose sentence representations via Multi-Task Learning.

In recent years, contextualized word embeddings have drawn considerable attention in light of the formidable gains that they achieve across a wide range of NLP and IR tasks. The pioneering work on ELMo (Peters et al., 2018) showed that significant gains can be achieved across a range of NLP tasks by considering the intermediate layers of a deep BiLSTM-based language model. Instead of standard bidirectional language modeling as in ELMo, the BERT approach (Devlin et al., 2019) developed at Google uses a training regimen considering Cloze-style masked language modeling, in which both sides of the context are simultaneously used to reconstruct an artificially masked word, along with an additional neighbour sentence prediction task. XLNet (Yang et al., 2019) is an auto-regressive Transformer-XL (Dai et al., 2019) based model using a permutation language model as the training task. XLNet outperforms BERT on various downstream tasks when they share the same number of model parameters and training corpus size. RoBERTa (Liu et al., 2019) improves the pre-training task of the original BERT model by removing the Next Sentence Prediction task and randomly generating different masks for words in a sentence. It also improves the performance of BERT by adding more training data. Reimers and Gurevych (2019) proposed Sentence-BERT, which utilizes Siamese and Triplet Networks to fine-tune BERT on NLI and Semantic Textual Similarity (STS) data to obtain more semantically meaningful sentence embeddings that can be compared using cosine similarity.

2.2 Analysing Linguistic Representations

Whereas in the field of computer vision, there has been prominent work on understanding what is happening inside popular kinds of models (Zeiler and Fergus, 2014), the latent representations of recent NLP models have long remained impervious and opaque, in the sense that it is not well-understood how they represent the relevant properties of language. While recently there has been substantial research on assessing the capabilities of BERT-like architectures (Rogers et al., 2020), this research for the most part does not shed sufficient light on the topological properties of the representation space.

The most well-known way to inspect the capabilities of sentence embeddings has been via what has been dubbed *probing*, i.e., supervised training of models that predict specific linguistic phenomena given embeddings as input. Kiros et al. (2015) evaluated the quality of their embeddings by using them for supervised downstream tasks such as sentiment polarity and question type classification. Adi et al. (2016) attempted to gain more specific insights by predicting word occurrences, word order, and sentence lengths. Bacon and Regier (2018) considered this approach to predict verb tense. Ettinger et al. (2018) trained classifiers for semantic roles and negation detection. Dasgupta et al. (2018) studied the argument sensitivity of the InferSent model by probing with respect to an NLI classification (contradiction vs. entailment). Kann et al. (2019) investigated verb alternation acceptability classifications. Conneau et al. (2018) predicted a wide range of mostly syntactic phenomena such as major syntactic constituents, the depth of the syntactic tree, grammatical number of the subject, and grammatical number of the object. For each probing task, they provide 100,000 training instances.

Probing provides important insights about whether sufficient signals needed for a given downstream task are available if one has sufficient supervision. However, training on 100,000 instances does not reveal whether these signals are genuinely present in the sentence representations, as opposed to just being learnable from the training data. For instance, consider an email spam classification task trained using a simple bag-of-words TF-IDF vector representation. With 100,000 training examples, a model will likely be able to learn to recognize salient kinds of spam emails with an accuracy significantly above the level of chance. However, this does not license the conclusion that the bag-of-words representation inherently captures some notion of *spamicity*, as it were.

Hence, an important complementary endeavour is to study the topology of the representation space. Zhu et al. (2018) proposed assessing sentence embeddings from a relational perspective in terms of proximity. In this paper, we specifically examine to what extent analogical relationships are reflected in terms of regularities in the representation space. Diallo et al. (2019) explored analogical embeddings for the relationship between questions and answers in question answering. Zhang and Baldwin (2019) investigated to what extent analogical reasoning can be used for relationships between documents.

3 Methodology

Our goal is to explore to what extent different sentence embedding spaces reflect analogical regularities of the form A is to B as C is to D. In the remainder of this paper, we shall invoke the notation $A : B :: C : D$ to refer to this sort of relationship. We will assess such relationships using the same methods as considered for word vectors. A typical choice is the method given by Eq. 1 (see Section 4.1 for further discussion).

To be able to perform our analysis, we induce two kinds of data. In Section 3.1, we create sentence analogies based on lexical analogies. In Section 3.2, we induce sentence analogies based on predefined relationships between sentences.

3.1 Sentence Analogies from Lexical Analogy

The first kind of sentence-level analogy data considered in our paper is induced based on lexical analogy data. Specifically, we consult Google's word analogy dataset (Mikolov et al., 2013a) and use it to construct 5 types of semantic sentence analogies and 5 types of syntactic sentence analogy categories.

3.1.1 Semantic Relationships

For semantic instances, we first create general-purpose sentence templates. Then, we replace a certain word in the template with words from Google's word analogy dataset. We consider the following categories

of relationships.

- **Common Capital Cities.** We first consult corpora to extract sentence templates such as "I'm not sure if they can travel to France." Then we replace the word "France" in the template with words from the Google Analogy dataset to create sentence pairs, as shown in Table 1.
- **All Capital Cities.** We create sentence templates such as "I've never been to Thimphu." For each word analogy pair in the Google dataset, we replace the word "Thimphu" with pertinent words from the pairs to obtain sentence pairs.
- **Currencies.** For currency–country pairs in the Google dataset, we create different templates for currency and country, respectively. Then, we replace the target word in the currency and country templates with word pairs to generate sentence pairs as shown in Table 1.
- **City in State.** We create a unified template for both city and state. Sentence pairs are then created by replacing a target word in the template with the a city or state name from the Google dataset.
- **Gender.** Google's word analogy dataset provides stereotypical male–female pairs (e.g., *son – daughter*), disregarding other gender identities. For these pairs, we again create templates, but invoke them in more intricate ways. For example, given a template "My grandpa makes wooden crafts and arts.", we can replace the word "grandpa" with any word describing family members such as "grandma", "father", and "mother". However, when the candidate word is a word that describes an occupation, we replace the word "My" in the original template with "The". When the candidate word is a pronoun such as "he" or "she", we omit the word "My" from the template.

3.1.2 Syntactic Instances

For (morpho-)syntactic questions, we first perform part-of-speech tagging and dependency parsing[2] to analyze the structure of the sentences in the MNLI dataset (Williams et al., 2018a) and extract sentences that correspond to a certain structure. Subsequently, we invoke a set of rules to generate new sentences from the original ones. The specific sentence generation schemes invoked to generate the evaluation data for the syntactic categories are as follows.

- **Comparative.** We first find a sentence containing a comparative adjective followed by "than", and then replace the comparative adjective with its original form and remove the noun or clause after "than" to obtain comparative sentence pairs, as again exemplified in Table 1.
- **Nationality Adjectives.** We create templates for nationalities and their corresponding adjectives. We then replace a target word in the template with the nationality designation or with the adjective word.
- **Opposites.** We first find a sentence containing an adjective and then replace the adjective with its antonym to obtain sentence pairs. Note that these antonym pairs generally bear a derivational connection, e.g., *efficient – inefficient*.
- **Plurals.** We retrieve a sentence with a plural noun and a numeral word between the noun, and then replace the plural noun with its singular form and replace the numeral word with "one", "a", or "an".
- **Verb Conjugation.** We first identify sentences containing an auxiliary verb followed by a verb and then remove the auxiliary verb and replace the verb in the sentence with its inflected form.

3.2 Analogy based on Relationships Between Sentences

In addition to our sentence analogy data derived from word analogies, we also create new diagnostic sentence analogy data based on specific forms of relationships between sentences. We start off with sentences extracted from NLI datasets, including SNLI (Bowman et al., 2015), Multi-NLI (Williams et al., 2018b), and SICK (Marelli et al., 2014).

3.2.1 Relationships

Entailment. Given two sentence pairs S_A, S_B and S_C, S_D, an entailment analogy holds between these two sentence pairs if the respective relationships between S_A and S_B and between S_C and S_D are both *entailment*, as annotated in the NLI data.

[2]We rely on SpaCy's English models for these two tasks.

Table 1: Examples of Lexical Analogy

	S_A	S_B
Common Capital Cities	They traveled to **Havana**.	They took a trip to **Cuba**.
All Capital Cities	I've never been to **Amman**.	I've never been to **Jordan**.
Currencies	The economy in **Japan** was great.	The **yen** appreciated due to the strong economy.
City in State	They go down to **Chandler**.	They go down to **Arizona**.
Man – Woman	The **man** makes wooden crafts and arts.	The **woman** makes wooden crafts and arts.
Comparative	The second article was **long**.	The second article was **longer** than the first one.
Nationality Adjective	The man from **Egypt** tapped his cheek.	The **Egyptian** man tapped his cheek.
Opposites	It's **possible** to measure it.	It's **impossible** to measure it.
Plurals	The Harvard data examined one **city**.	The Harvard data examined six **cities**.
Verb Conjunction	Duke will **play** better this year.	Duke **plays** better this year.

Negation. We consider sentence pair S_A, S_B as standing in a negation relationship if one has a negated meaning compared to the other. Some of the negation pairs are extracted from the SICK dataset, but most of our negation pairs are created by dependency parsing and rule-based transformations. Given two sentence pairs S_A, S_B and S_C, S_D, a negation analogy holds between these sentence pairs if the respective relationship between S_A and S_B and between S_C and S_D are both *negation*. An example is given in Table 2.

Passivization. Sentence pair S_A, S_B is regarded as standing in a passivization relationship if S_B is a passive form of S_A. Given two sentence pairs S_A, S_B and S_C, S_D, a passivization analogy holds between these sentence pairs if the respective relationships between S_A and S_B and between S_C and S_D are both *passivization*. We create this data from NLI sentences using a dependency-driven rule-based transformation. Our passivization data is extracted from the argument sensitivity dataset from Zhu et al. (2018), which contains suitable passivization pairs.

Objective Clause. Given sentence pairs S_A, S_B and S_C, S_D, an objective clause analogy holds between these two sentence pairs if S_A and S_C include a verb that is indicative of stating a fact or opinion (e.g., *say*, *tell*, *think*, etc.), followed by clauses S_B and S_D, respectively.

Predicative Adjective Conversion We deem a predicative adjective conversion relationship as holding between sentences S_A and S_B if S_A contains an adjective, while S_B contains a predicative clause that shares the same meaning with the adjective. Given sentence pairs S_A, S_B and S_C, S_D a predicative adjective conversion analogy holds between them if the respective relationships between S_A and S_B and between S_C and S_D are both of this form.

3.2.2 Candidate Sets

For a given hypothesis, there may be a multitude of valid premises that entail it. Given an analogy of the form $S_A : S_B :: S_C : S_D$, we need to restrict the scope of candidate sentences for S_D so as to ensure the uniqueness of the correct answer. Instead of considering the entire corpus as a candidate sentence set, our candidate sentence sets for the relationships in Section 3.2.1 consist of one true candidate and several challenging distractor candidates that are similar to the true candidate at a superficial level but modified to be semantically different. Table 2 provides examples for a brief overview of the resulting task. In the following, we explain the distractor generation in further detail.

Not Negation. We insert the negation marker *not* after the first auxiliary verb in the original true target sentence to generate a new distractor sentence. If the sentence already contains the negation marker *not*, we instead remove it. Not Negation aims to detect whether a sentence embedding model is misled by a negation of the sentence relation caused by adding the word *not*.

Random Deletion. We randomly delete words in the original sentence with a probability of 20% to generate a new sentence. If the length of the sentence is less than 5, we delete at least one word. However, simply deleting arbitrary words in the original sentence may not always affect the semantic relationship.

Table 2: Example of candidate sets for relation-based analogy

	Entailment	Negation
S_A	The man is heaving barbells.	There is no deer jumping a fence.
S_B	The man is lifting barbells.	A deer is jumping over the fence.
S_C	A man is singing a song and playing the guitar.	There is no boy hitting the football.
Positive Candidate	A man is singing and playing the guitar.	A boy is hitting the football.
Not Negation	A man is not singing and not playing the guitar.	A boy is not hitting the football.
Random Deletion	A man is the guitar.	is the football.
Random Masking	A [MASK] is singing and playing the guitar.	A [MASK] is [MASK] the football.
Span Deletion	A man is singing the guitar.	A boy the football.
Word Reordering	and playing the guitar A man is singing.	The football a boy is hitting.

For example, consider the hypothesis “John ate a yummy sandwich.” vs. the premise “John ate a delicious sandwich.” If we delete the word *delicious* from the premise sentence, the relation between the hypothesis and the new sentence is also entailment. In order to avoid this situation, we pick words for deletion that are not adjectives, adverbs, determiners, or auxiliary verbs.

Random Masking. Following BERT (Devlin et al., 2019), we randomly replace tokens in the original sentence with BERT’s special "[MASK]" token, where the probability of a certain token being masked is 20%. For sentence embedding methods that are not based on BERT, the "[MASK]" token is treated as an "UNK" token, which represents unknown words. The purpose of random masking task is exploring whether replacing a word with a special meaningless token will affect a model’s performance in judging a semantic relation.

Span Deletion. A number of text spans are sampled, with span lengths drawn from a Poisson distribution with $\lambda = 3$. Each text span is deleted from the original sentence. Span Deletion is inspired by the text infilling operation in BART (Lewis et al., 2019). The only difference is that BART replaces text spans with a "[MASK]" token instead of deleting them. One difference between random deletion and span deletion is that we do not pose any restrictions on the word spans to be deleted. Since a continuous text span in a sentence often represents a phrase or a sub-clause, in most cases, the generated sentence’s meaning and relationship is different from the original sentence. But there may also be some exceptions to this. Hence, we rely on manual checking to avoid such issues.

Word Reordering. We randomly choose a word in the original sentence as a pivot, and then swap the words before and after the pivot to obtain a new sentence, which is likely grammatically incorrect and not an appropriate target to be selected. We invoke this sort of word reordering to test whether an embedding model is sensitive to semantic relation changes caused by changes of the word order. Clearly, a simple averaging of word embeddings is not able to distinguish this sort of example from the true target sentence, but it is not yet known to what extent more sophisticated sentence embedding models may suffer from this issue.

4 Experiments

In a sentence analogy task, we are given two pairs of sentences sharing a relation. For example, “He is very enamored with culture in Egypt” : “He is very enamored with Egyptian culture”, and “He is very enamored with culture in Bulgaria” : “He is very enamored with Bulgarian culture”. The goal is to identify the fourth sentence given the first three sentences. The kind of analogical relationship sought for the sentence pairs is not explicitly provided. The number of sentence pairs and analogies in each category of our analogy dataset is given in Table 3,

Table 3: Number of sentence and question pairs in each category

Category	Sentence Pairs	Analogies
Common Capital City	138	9,453
All Capital Cities	928	430,128
City in State	402	80,601
Currency	150	11,175
Gender	126	7,875
Comparative	466	108,345
Opposite	513	131,328
Nationality Adjective	205	20,910
Plural	512	130,816
Verb Conjugation	451	101,475
Entailment	673	226,128
Negation	511	130,305
Passivization	256	32,640
Objective Clause	563	158,203
Adjective Clause	550	150,975

4.1 Evaluation Metric

In word analogy tasks, the offset between word vectors is often used to determine relations between words. For example, in order to solve *man* is to *woman* as *king* is to W, we find a word W for which the corresponding vector is the closest to $\vec{v}_{\text{man}} - \vec{v}_{\text{woman}} + \vec{v}_{\text{king}}$. This amounts to optimizing Eq. 1. Levy and Goldberg (2014) studied this in more detail, referring to the aforementioned method as *3CosAdd*, while introducing a multiplicative variant called *3CosMul*, which often yields better empirical results.

Linzen (2016), Schluter (2018), and Nissim et al. (2019) highlighted the significance of excluding the other analogy words in Eq. 1. Given a word analogy problem of the form $A : B :: C : D$, the standard procedure is to disregard any D that is equal to A, B, or C. This constraint drastically improves the performance of word embedding models on word analogy datasets such as the Google dataset (Mikolov et al., 2013a), but may also lead to biased results.

In our experiments, we consider both 3CosAdd and 3CosMul, and evaluate these both with the additional constraint (3CosAdd, 3CosMul) and without it (3CosAdd-U, 3CosMul-U), where the suffix -U denotes an *unconstrained* evaluation.

4.2 Embedding Methods

In our experiments, we consider a number of embedding models. These include simple word vector aggregation methods such as the Average of GloVe embeddings (abbreviated as *GloVe*). The concatenation of Discrete Cosine Transform coefficients (*DCT*) embeddings are generated by concatenating the first k DCT coefficients. In our experiment, k ranges from 0 to 6, and for space reasons, we report the best-performing result. For sentence embeddings based on RNNs such as Skip-Thought Vectors (*SkipThought*), Quick-Thought vectors (*QuickThought*), and the General Purpose Sentence Encoder by Subramanian et al. (2018) (*GenSen*), we use the hidden state of the final RNN cell as the sentence embedding. For InferSent, we use max-pooling over all hidden states of RNN cells to produce sentence embeddings. Facebook released two versions of the InferSent model, the earlier version (*InferSentV1*) is trained based on GloVe word embeddings, while the second version (*InferSentV2*) is trained using fastText word embeddings. The Universal Sentence Encoder (Cer et al., 2018) comes in two versions, the first one based on Deep Average Networks (*USE-DAN*), the second based on Transformer networks (*USE-Transformer*).

For contextual embedding models such as BERT, XLNet, RoBERTa, along with Sentence-BERT (Reimers and Gurevych, 2019), we consider two popular methods to generate a sentence embedding. The first one (*-CLS*) consists in using the embedding of the special "[CLS]" token in the sentence, followed by a linear transformation and a tanh activation layer. Another method (*-AVG*) involves computing the element-wise sum of contextual word representations w_1, w_2, w_3, ..., w_n at the top level of the Transformer encoder and dividing it by the square root of the sentence length. For a given model, we only show the pooling method that obtained the highest accuracy in the experimental results. We consider different

versions of the models (*-Base*, *-Large*) as released by the original authors. In our result tables, the Sentence-BERT models by Reimers and Gurevych (2019) based on BERT and RoBERTa are referred to as *SBERT* and *SRoBERTa*, respectively.

4.3 Results and Analysis

Sentence Analogy from Lexical Analogy Table 4 provides the overall aggregate results for lexical analogy-based pairs, while Table 5 specifically assesses the semantic analogy and syntactic analogy categories. With an unconstrained evaluation, we observe that all considered sentence embedding models show relatively poor success rates. We conjecture that this is because $\mathrm{sim}(D, B) - \mathrm{sim}(D, A)$ tends to be fairly small in most cases, so 3CosAdd-U and 3CosMul-U often degenerate mainly to evaluating $\mathrm{sim}(D, C)$.

With a traditional constrained evaluation, several methods show substantial regularities. Averages of GloVe vectors draw on the linguistic regularities inherent in the GloVe vectors. The Discrete Cosine Transform method outperforms other sentence embedding methods for most of the categories. InferSent outperforms contextual embedding methods, with XLNet-Large obtaining the weakest results across all considered models. Despite the strength of InferSent, fine-tuning BERT on NLI datasets does not improve the performance of the model on lexical analogy based tasks. SBERT and SRoBERTa obtained a lower accuracy than BERT and RoBERTa, respectively.

From the results of Table 5, we find that capturing the kinds of semantic analogies considered in our study is more challenging than capturing the syntactic analogies. Most of the sentence embedding models we tested (except XLNet) excelled at solving syntactic question pairs using the 3CosAdd metric, while few of them perform well on semantic analogy pairs. In particular, contextual embedding-based models appear capable of reflecting syntactic phenomena, but do not appear to yield semantic knowledge at the same level as word embedding models such as GloVe, although they are known to be able to emit world knowledge when evaluated in language modeling settings (Petroni et al., 2019).

Table 4: Experimental results over all lexical analogy-based data

Representation	3CosAdd-U	3CosAdd	3CosMul-U	3CosMul
GloVe	0.4092	0.8189	0.4092	0.8039
DCT (k=0)	0.5193	0.8865	0.5193	0.8688
SkipThought	0.1805	0.6251	0.1805	0.4540
QuickThought	0.1337	0.6318	0.1337	0.5907
InferSentV1	0.2787	0.7118	0.2787	0.6594
InferSentV2	0.3405	0.8323	0.3405	0.5000
GenSen	0.3756	0.5366	0.3756	0.5038
USE-DAN	0.0316	0.4995	0.0316	0.0658
USE-Transformer	0.0518	0.5714	0.0518	0.0818
BERT-Base-AVG	0.1537	0.6471	0.1537	0.6415
BERT-Large-AVG	0.2375	0.6643	0.2375	0.5805
XLNet-Base-AVG	0.0234	0.4223	0.0234	0.4214
XLNet-Large-AVG	0.0105	0.2397	0.0105	0.2383
RoBERTa-Base-CLS	0.0645	0.6117	0.0645	0.6107
RoBERTa-Large-AVG	0.0793	0.6229	0.0793	0.6221
SBERT-Base-AVG	0.0977	0.5108	0.0977	0.1821
SBERT-Large-AVG	0.1513	0.5347	0.1513	0.3557
SRoBERTa-Base-AVG	0.0881	0.2548	0.0881	0.1459
SRoBERTa-Large-AVG	0.1073	0.2978	0.1073	0.2008

Sentence Analogy from Relationships Between Sentences In Figure 1, we provide the results on the set of all relation-based sentence analogies from Section 3.2. The first row in the table represents the accuracy, while the following rows show the probability that a certain adversarial candidate was chosen by the method. Note that because our relation-based analogy contains entailment and a sentence always entails itself, we omit the unconstrained versions of 3CosAdd and 3CosMul in this evaluation. The InferSentV2 and GenSen models achieve the highest accuracy on the relation-based analogy tasks when using 3CosAdd and 3CosMul, respectively, while the large version of XLNet achieves the lowest accuracy. By comparing the performance of the base and large versions of the pre-trained models based on Transformers, we find that large models have a tendency to confuse the real premise sentence with the not-negated form of the original sentence, which hampers their performance in this sort of evaluation.

Table 5: Experimental results on semantic (left) and syntactic sentence analogy (right)

Representation	3CosAdd-U	3CosAdd	3CosMul-U	3CosMul
GloVe	0.4159	0.7453	0.4159	0.7244
DCT (k=1)	0.4300	0.8477	0.4300	0.8019
SkipThought	0.2024	0.4382	0.2024	0.3563
QuickThought	0.0367	0.3974	0.0367	0.3129
InferSentV1	0.2655	0.6159	0.2655	0.5502
InferSentV2	0.3755	0.7967	0.3755	0.5490
GenSen	0.2374	0.3457	0.2374	0.2617
USE-DAN	0.0317	0.1471	0.0317	0.0598
USE-Transformer	0.0606	0.2773	0.0606	0.0831
BERT-Base-AVG	0.1128	0.4481	0.1128	0.4358
BERT-Large-AVG	0.2131	0.4852	0.2131	0.4344
XLNet-Base-AVG	0.0239	0.1521	0.0239	0.1510
XLNet-Large-AVG	0.0062	0.0327	0.0062	0.0318
RoBERTa-Base-CLS	0.0609	0.4171	0.0609	0.4162
RoBERTa-Large-CLS	0.0267	0.4507	0.0267	0.4496
SBERT-Base-AVG	0.0640	0.4190	0.0640	0.1473
SBERT-Large-AVG	0.1143	0.4898	0.1143	0.3656
SRoBERTa-Base-AVG	0.0135	0.0347	0.0135	0.0248
SRoBERTa-Large-AVG	0.0190	0.0886	0.0190	0.0587

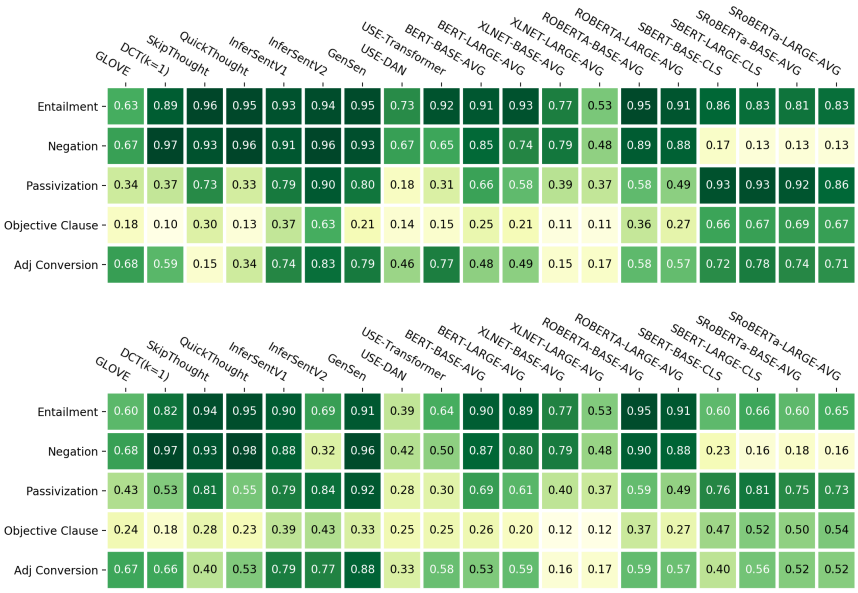
Representation	3CosAdd-U	3CosAdd	3CosMul-U	3CosMul
GloVe	0.4019	0.8993	0.4019	0.8908
DCT (k=0)	0.5276	0.9298	0.5276	0.9305
SkipThought	0.1565	0.8295	0.1565	0.5608
QuickThought	0.2397	0.8882	0.2397	0.8945
InferSentV1	0.2983	0.8547	0.2983	0.8222
InferSentV2	0.2883	0.8853	0.2883	0.4271
GenSen	0.5815	0.8212	0.5815	0.8645
USE-DAN	0.0315	0.8847	0.0315	0.0724
USE-Transformer	0.0423	0.8930	0.0423	0.0804
BERT-Base-AVG	0.1984	0.8647	0.1984	0.8665
BERT-Large-AVG	0.2641	0.8602	0.2641	0.7402
XLNet-Base-AVG	0.0228	0.7177	0.0228	0.7171
XLNet-Large-AVG	0.0152	0.4660	0.0152	0.4642
RoBERTa-Base-AVG	0.2155	0.8531	0.2155	0.8524
RoBERTa-Large-AVG	0.1161	0.8442	0.1161	0.8433
SBERT-Base-AVG	0.1348	0.6119	0.1348	0.2205
SBERT-Large-AVG	0.1921	0.5842	0.1921	0.3448
SRoBERTa-Base-AVG	0.1703	0.4970	0.1703	0.2792
SRoBERTa-Large-AVG	0.2046	0.5281	0.2046	0.3572

Figure 1: Error Analysis of Pre-trained Sentence Embeddings on Relation Based Analogy using 3CosAdd (top) and 3CosMul (bottom)

Another interesting finding is that BERT-based models improve their performance on distinguishing the actual premise from the negated version of the hypothesis after fine-tuning on the SNLI dataset. Yet, they have a higher probability of being misled by adversarial candidates created by Span Deletion and Word Reordering.

Figure 2 shows the accuracy of pre-trained sentence embeddings broken down by particular sentence relation-based analogy forms. We observe that the difficulty of some relation-based analogy tasks is substantially higher than for others. Most of the models, except for XLNet-Large and GloVe, achieve relatively high accuracy on Entailment analogy, while none of the models perform well on the proposed Objective Clause analogy. In addition, Transformer-driven models have made great progress at capturing syntactic analogies such as Passivization, Objective Clauses, and Predicative Adjective Conversion, but their ability to identify the Negation relation appears limited in this sort of evaluation. We also find that the performance of the pre-trained models on relational analogy tasks might be affected by the network architecture. For example, sentence embedding models built on RNNs fare better at recognizing Entailment and Negation analogy, but their performance on distinguishing Passivization and Objective

Figure 2: Accuracy of Pre-trained Sentence Embeddings on each Relation Based Analogy using 3CosAdd (top) and 3CosMul (bottom)



Clause is not as good as Transformer-driven models.

5 Conclusion

This paper presents several new datasets to test to what extent existing sentence embedding models exhibit regularities with regard to sentence analogies. Most of the sentence embedding models we tested succeeded in recognizing syntactic analogies based on lexical ones, but had a harder time capturing semantic regularities by means of an analogy task. Moreover, the remarkable success of BERT-style contextual embeddings does not always translate into better regularities in the vector space of fixed-length sentence embeddings. More training data and model parameters as well do not necessarily yield better results. In many cases, word vector averages or a Discrete Cosine Transform of word embeddings outperform more complex sentence embedding models. Resources related to this study are available online at `http://sentence.embeddings.org`.

References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2016. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. *arXiv preprint arXiv:1608.04207*.

Nada Almarwani, Hanan Aldarmaki, and Mona Diab. 2019. Efficient sentence embedding using discrete cosine transform. *arXiv preprint arXiv:1909.03104*.

Geoff Bacon and Terry Regier. 2018. Probing sentence embeddings for structure-dependent tense. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 334–336.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. 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.

Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. *arXiv preprint arXiv:1805.01070*.

Zihang Dai, Zhilin Yang, Yiming Yang, William W Cohen, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

Ishita Dasgupta, Demi Guo, Andreas Stuhlmüller, Samuel J Gershman, and Noah D Goodman. 2018. Evaluating compositionality in sentence embeddings. *arXiv preprint arXiv:1802.04302*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186.

Aïssatou Diallo, Markus Zopf, and Johannes Fürnkranz. 2019. Learning analogy-preserving sentence embeddings for answer selection. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 910–919, Hong Kong, China, November. Association for Computational Linguistics.

Allyson Ettinger, Ahmed Elgohary, Colin Phillips, and Philip Resnik. 2018. Assessing composition in sentence vector representations. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1790–1801.

Katharina Kann, Alex Warstadt, Adina Williams, and Samuel R Bowman. 2019. Verb argument structure alternations in word and sentence embeddings. In *Proceedings of the Society for Computation in Linguistics (SCiL) 2019*, pages 287–297.

Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Omer Levy and Yoav Goldberg. 2014. Linguistic regularities in sparse and explicit word representations. In *Proceedings of the eighteenth conference on computational natural language learning*, pages 171–180.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

Tal Linzen. 2016. Issues in evaluating semantic spaces using word analogies. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 13–18, Berlin, Germany, August. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Lajanugen Logeswaran and Honglak Lee. 2018. An efficient framework for learning sentence representations. *arXiv preprint arXiv:1803.02893*.

Marco Marelli, Luisa Bentivogli, Marco Baroni, Raffaella Bernardi, Stefano Menini, and Roberto Zamparelli. 2014. Semeval-2014 task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, pages 1–8.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. 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*, pages 746–751, Atlanta, Georgia, June. Association for Computational Linguistics.

Malvina Nissim, Rik van Noord, and Rob van der Goot. 2019. Fair is better than sensational: Man is to doctor as woman is to doctor. *arXiv preprint arXiv:1905.09866*.

Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2017. Unsupervised learning of sentence embeddings using compositional n-gram features. *arXiv preprint arXiv:1703.02507*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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)*, pages 2463–2473, Hong Kong, China, November. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A Primer in BERTology: What we know about how BERT works. *arXiv e-prints*, page arXiv:2002.12327, February.

David E. Rumelhart and Adele A. Abrahamson. 1973. A model for analogical reasoning. *Cognitive Psychology*, 5(1):1 – 28.

Andreas Rücklé, Steffen Eger, Maxime Peyrard, and Iryna Gurevych. 2018. Concatenated power mean word embeddings as universal cross-lingual sentence representations.

Natalie Schluter. 2018. The Word Analogy Testing Caveat. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 242–246, June.

Sandeep Subramanian, Adam Trischler, Yoshua Bengio, and Christopher J Pal. 2018. Learning general purpose distributed sentence representations via large scale multi-task learning.

Peter D. Turney and Michael L. Littman. 2005. Corpus-based learning of analogies and semantic relations. *Mach. Learn.*, 60(1-3):251–278.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018a. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2018b. The multi-genre nli corpus.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

Matthew D. Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 818–833, Cham. Springer International Publishing.

Jingyuan Zhang and Timothy Baldwin. 2019. Evaluating the utility of document embedding vector difference for relation learning. *CoRR*, abs/1907.08184.

Xunjie Zhu, Tingfeng Li, and Gerard de Melo. 2018. Exploring semantic properties of sentence embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 632–637.