Phrase Analogies in Large VAEs

Joshua Rivera
New York University
jcr567@nyu.edu

Karan Sagar
New York University
karan@karansag.org

Evan Silverman
New York University
es4753@nyu.edu

Abstract

The geometry of sentence embedding is not a well studied area in NLP. While related to word embedding, there is additional complexity from syntactic and semantic variability that is not as easily comparable due to the infinite customizability of multi-word phrases. With the release of OPTIMUS providing a large-scale pre-trained Variational Auto-Encoder(VAE), we believe, using multiple forms of phrase analogies, we can better study the shape of vector representations of multi-word phrases and improve our understanding on their transformations between related forms in their vector space.

1 Introduction

In this paper, we explore the latent space of OPTIMUS, a new large-scale variational auto-encoder (VAE). We do so by assessing its performance on analogies. These analogies come in two varieties: syntactic and relationship. Since one of the goals of OPTIMUS is to produce a more semantically meaningful latent space than traditional sentence embeddings (Li et al., 2020), we assess its performance on that metric. Our evaluation methods are varied: BLEU, exact-match, manual, and external model evaluation.

Sentence embeddings can be useful for capturing properties of text and are frequently used in downstream tasks (Conneau et al., 2018b). While work has been done to produce universal sentence embeddings (Kiros et al., 2015; Conneau et al., 2018a; Cer et al., 2018) there remains room to explore regularities of these embeddings. Evaluating these regularities can be challenging due to the lack of a direct inverse function from the embedding (latent) space to the sentence space. Some approaches to mitigate this include probing (Conneau et al., 2018b) and natural language generation (Kerscher and Eger, 2020; Wang and Lepage, 2020). Zhu and de Melo (2020) also explored sentence analogies using a constructed syntactic and relationship analogy dataset. We take this same approach in our dataset construction.

Variational auto-encoders are auto-encoders that output a relatively low-dimensional latent space that can provide a high-level feature representation of the sentence. This can help, for example, guide sentence generation (Li et al., 2020). Because VAEs decode, intuitively, from regions in space, they provide an easy way to decode the latent space back into the sentence space. Until recently, VAEs (Kingma and Welling, 2014; Bowman et al., 2016) have not shown comparable performance on performance metrics like GLUE. However, OPTIMUS (Li et al., 2020), the first large-scale VAE, has shown good performance here. As a result, we believe exploration of sentence geometry via phrasal syntactic and relationship analogies $S_a : S_b :: S_c : S_d$ for the phrases S_a, S_b, S_c, S_d may allow us to better understand how models can understand and interpret transformation within a larger sentence.

Our paper is structured as follows: Section 2 presents related work, while Section 3 presents data and methods. Section 4 presents preliminary experimental results, while Section 5 provides a collaboration statement. Later sections have references.

2 Related Work

Previous work has explored sentence embeddings in some depth. Skip-thought (Kiros et al., 2015) trains an encoder-decoder architecture, while InferSent (Conneau et al., 2018a) uses NLI data to train a BiLSTM network. Cer et al. (2018) trains a "Universal Sentence Encoder", while Reimers and Gurevych (2019) trains a Siamese network on NLI data that significantly outperformed previous methods.

Work has been done to evaluate these models, particularly through probing and related methods.

SentEval (Conneau et al., 2018a) provides a toolkit to assess sentence embeddings by using them as features in transfer tasks. Conneau et al. (2018b) introduces a comprehensive set of probing tasks to assess sentence embeddings. Vec2Sent (Kerscher and Eger, 2020) uses an RNN optimized for natural language generation (NLG) to probe sentence embeddings.

Variational auto-encoders (VAE)s were introduced by (Kingma and Welling, 2014; Rezende et al., 2014) and expanded upon by (Bowman et al., 2016). Bowman et al. (2016) has a Gaussian prior on the latent space, which provides a lower-bound for the log-likelihood of the data. This allows arbitrary points that reasonably fall under the space determined by the prior to decode to valid sentences. Other properties include decoding of any homotopy, which, for the purposes here, is a linear interpolation between sentences. Bowman et al. (2016) also dealt with the so-called "KL-vanishing problem"; many hyperparameter choices converged the latent space toward the Gaussian prior.

However, Bowman et al. (2016) and other VAEs showed that performance was worse than corresponding LSTMs. OPTIMUS (Li et al., 2020) was the first large-scale VAE. It uses a BERT-based encoder and GPT-2-based decoder to achieve better results than BERT on GLUE benchmarks. It also claims strong latent space manipulation and guided language generation capabilities.

We take the same approach here in our dataset construction as in (Zhu and de Melo, 2020). That is, we use lexical analogies to derive syntactic analogies and use existing relationship analogies from MNLI. However, unlike (Zhu and de Melo, 2020) the language-generation capabilities of OPTIMUS allow us to assess our analogy directly rather than through a nearest-neighbors calculation. We believe this provides a potentially stronger result. Kerscher and Eger (2020) also uses NLG to assess sentence embeddings via analogy, but that trains a separate de-novo decoder for existing embedding methods and can have similar caveats to probing methods.

3 Methods and Data

3.1 Data

3.1.1 Data Format

For this paper we used two types of analogies, syntactical and relationship. To generate our data, we used a method similar to that which is mentioned in (Zhu and de Melo, 2020).

That is, for the syntax analogy datasets we used a modified version of the Google's lexical analogies dataset (Mikolov et al., 2013) for the syntactic pairs and used SNLI (Bowman et al., 2015) and MNLI (Williams et al., 2018) for the phrase templates as well as additional sentence analogies. After processing the newly generated syntactic phrase analogy pairs, we put them in the format $S_a : S_b :: S_c : S_d$, that is, a dataset with four columns S_a, S_b, S_c, S_d where on every row S_a, S_b and S_c, S_d are distinct syntactic phrase analogy pairs with the same syntactic transformation. The syntactic analogies used were opposites, comparative, plural, and tense. This allows us to see how analogies are interpreted on multiple parts of speech including nouns, verbs, adjectives, and adverbs.

For example:
S_a=:The woman is a competitive athlete.
S_b=:The woman is a noncompetitive athlete.
S_c=:A man is wearing comfortable shoes.
S_d=:A man is wearing uncomfortable shoes.

where we see $S_a : S_b :: S_c : S_d$ since

competitive:noncompetitive::comfortable:uncomfortable

To see how OPTIMUS can handle non-lexically-based analogies as well as abstract analogies, we also used SNLI and MNLI, as is, by using its "*gold_label*" tag to determine analogies based on entailment and contradiction that were then processed in the same format as the syntactical analogies.

3.1.2 Category Descriptions

Opposites: contains one word replaced with its antonym with a negation prefix such as "un-", "im-", etc.

Comparative: determines the comparative form of an adjective and modifies the phrase such that the standard adjectival form can replace it.

Plural: modifies some singular noun phrase within the sentence by changing the noun to its plural form and modifying any quantifiers.

Tense: changes the present tense form of some verb to its corresponding past tense.

Entailment: the first phrase implies the the second phrase, but they are not necessarily equivalent.

Contradiction: similar to syntactical opposites in meaning, but can have drastically different structure from its paired phrase.

Some lexical pairs can work as analogies in theory, but in actual usage may not function in their analogical position. This affects categories such "opposites" mostly as some pairs may not act as true antonyms. For those specific cases, we avoided using it as a lexical pair to process phrase pairs.

For example:

S_a =:The boy was possibly here.
S_b =:*The boy was impossibly here.

3.2 Methods

3.2.1 Scoring

In order to assess whether the decoded output from our analogy transformation matches the gold standard output, we use a variety of metrics. For syntactic data, we assess a BLEU score, an exact match, and a manually-scored sample of 200 data points. For relationship data, we use the NLI label predicted by a SoTA pre-trained model fine-tuned on NLI data.

3.2.2 BLEU

Our BLEU scoring uses the NLTK library's [1] translation BLEU function with the standard 4-gram.

3.2.3 Exact Match

The exact-match assesses a binary output between our $gold\_label$ and out prediction through case-insensitive string comparison.

3.2.4 Manual scoring

Because assessment of "correctness" in syntactic analogies can be challenging to do in an automated way, we take a representative sample of 200 predicted values and score them manually. These are scored three ways: CORRECT, PARTIALLY CORRECT, INCORRECT. From this, we do qualitative analysis.

3.2.5 NLI Model Prediction

In assessing analogies based on NLI relationships, we merely want the relationship between sentence pairs to be preserved in the analogy transformation. However, our prediction may be semantically and syntactically different from the $gold\_label$, S_d, while still being correct. As a result, we use the predicted label of a near SotA (Conneau et al., 2020) model fine-tuned on NLI data to assess the correctness of our prediction. In particular, we chose an XLM-R fine-tuned on XNLI data (Conneau et al., 2018c). The model fine-tunes XLM-R on both the MNLI training data and all the XNLI validation and test data. This was chosen for its ease of access and focus on zero-shot classification. Note: for the final draft, we plan to use the current MNLI SotA model (Raffel et al., 2020) on NLI fine-tuned appropriately.

4 Preliminary Results

Early results show multiple issues. First, many solutions created by OPTIMUS to the analogies are completely wrong. Some change the phrase into an unexpected one that seems to have no relation, while others make no changes at all. We have some ideas to potentially solve the issue, including changing some of OPTIMUS's temperature parameters.

There were many sentences that did generate correctly. A significant number did solve the analogy completely correctly. Many also solved it correctly semantically but not syntactically. For example, using "not" instead of the prefix "un-" before "aware" in the "opposites" dataset.

Overall, the results are too early to be conclusive, but they are reasonably promising.

5 Collaboration Statement

All team members participated in developing core ideas. Initially, Joshua and Evan connected with Alex Warstadt for guidance, Karan helped do initial research and clarified concepts.

All team members worked together on all parts of the project. Joshua focused primarily on working with the datasets while Karan and Evan focused on modifying and running OPTIMUS.

Everyone participated in the writing of this paper.

[1] https://github.com/nltk/nltk

6 Code

The code for this paper can be viewed on our repository on GitHub. [2]

References

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.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21, Berlin, Germany. Association for Computational Linguistics.

Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Lyn Untalan Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Céspedes, Steve Yuan, Chris Tar, Yun hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder. In *In submission to: EMNLP demonstration*, Brussels, Belgium.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2018a. 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, Copenhagen, Denmark. Association for Computational Linguistics.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018b. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018c. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Martin Kerscher and Steffen Eger. 2020. Vec2Sent: Probing sentence embeddings with natural language generation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1729–1736, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-thought vectors. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, page 3294–3302, Cambridge, MA, USA. MIT Press.

Chunyuan Li, Xiang Gao, Yuan Li, Baolin Peng, Xiujun Li, Yizhe Zhang, and Jianfeng Gao. 2020. Optimus: Organizing sentences via pre-trained modeling of a latent space. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4678–4699, Online. Association for Computational Linguistics.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Nils Reimers and Iryna Gurevych. 2019. 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)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1278–1286, Bejing, China. PMLR.

[2] https://github.com/karansag/phrase-analogies-large-vae

Liyan Wang and Yves Lepage. 2020. Vector-to-sequence models for sentence analogies. In *2020 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2020*, 2020 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2020, pages 441–446. Institute of Electrical and Electronics Engineers Inc.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. 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. Association for Computational Linguistics.

Xunjie Zhu and Gerard de Melo. 2020. Sentence analogies: Linguistic regularities in sentence embeddings. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3389–3400, Barcelona, Spain (Online). International Committee on Computational Linguistics.