# Towards Unsupervised Compositional Entailment with Multi-Graph Embedding Models

**Lorenzo Bertolini** **Julie Weeds** and **David Weir**
University of Sussex
Brighton, UK
{l.bertolini, juliwe, d.j.weir}@sussex.ac.uk

## Abstract

Compositionality and inference are essential features of human language, and should hence be simultaneously accessible to a model of meaning. Despite being theory-grounded, distributional models can only be directly tested on compositionality, usually through similarity judgements, while testing for inference requires external resources. Recent work has shown that knowledge graph embeddings (KGE) architectures can be used to train distributional models capable of learning syntax-aware compositional representations, by training on syntactic graphs. We propose to expand such work with Multi-Graphs embedding (MuG) models, a new set of models learning from syntactic and knowledge-graphs. Using a phrase-level inference task, we show how MuGs can simultaneously handle syntax-aware composition and inference, and remain competitive distributional models with respect to lexical and compositional similarity.

## 1 Introduction

Drawing an inference over structured text is considered to be a basic aspect of natural language understanding (Pavlick and Callison-Burch, 2016). To build structured meaning, humans rely on compositionality (Frege, 1892; Mollica et al., 2020). For this reason, much work has underlined the connection between composition, the construction of complex meaning from smaller units, and inference (MacCartney and Manning, 2008; Baroni et al., 2012; Pavlick and Callison-Burch, 2016; Pavlick and Kwiatkowski, 2019). With respect to recently popularised large language models (LLMs) like BERT (Devlin et al., 2019), the literature has produced contrasting evidence, both against (Keysers et al., 2020; Do and Pavlick, 2021; Bertolini et al., 2022) and in support (Brown et al., 2020; Nie et al., 2020) of these models being able to simultaneously handle composition and inferences with little to no supervision. However, most of the work has focused on sentence-level inference. Multiple pieces of evidence have shown that, when solving such tasks, models strongly rely on biases and spurious correlations in the benchmarks (Poliak et al., 2018; Dasgupta et al., 2018; McCoy et al., 2019). To address this issue, authors proposed to focus on phrase-level tasks (e.g., Yu and Ettinger (2020, 2021); Bertolini et al. (2022)). In particular, Bertolini et al. (2022) showed that LLMs learn to make robust compositional inferences regarding adjective-noun phrases only with direct supervision, and linked this ability to non-lexical subword units. While computationally effective, this solution is poorly grounded in linguistic and cognitive theories.

Recently, Bertolini et al. (2021) showed how training knowledge-graph embedding (KGE) architectures on syntactic graphs leads to distributional models able to learn syntax-aware compositional representations. While these models theoretically satisfy the compositionality principle (Frege, 1892; Partee et al., 1995), like LLMs, they still require external resources or training to be evaluated on inference. In this work, we propose to expand syntactic-graphs distributional models (SyG) with knowledge-graph, and propose Multi-Graph (MuG) models. We argue that, by training on both data sources, MuG could inherit compositional abilities from SyGs, and learn to manipulate the *hypernym* relation from KGE. Thus, MuG models should be able to handle both composition and inference simultaneously in the form of compositional entailment, in a fully unsupervised manner. Since previous results found rotation to better encode hierarchical relations (Chami et al., 2020) such as entailment, and reflection to be most suitable to represent syntactic information (Bertolini et al., 2021), we hypothesize that an attention-based hybrid model will be the best architecture to simulta-



*Proceedings of the 15th International Conference on Computational Semantics* pages 50–61
June 21–23, 2023. 

neously handle compositionality and inference.

Our contributions are four-fold. First, we introduce Multi-Graph (MuG) models, a new set of embedding models trained on syntactic and knowledge-graphs. Second, we provide evidence that, under the correct combination of training method and architecture, MuG models can tackle compositional entailment, using a syntax-aware composition. Third, we propose a detailed analysis describing the behaviour of the best MuG model, clearly showing how the three macro classes of adjectives and the structure of the inference shape the behaviour of the model. Fourth, we investigate which abilities, in terms of distributional and knowledge-based, are inherited by MuGs. We show that MuGs are competitive distributional models, but struggle under a graph-related task, likely due to an incompatibility with respect to negative samples rate during training.

The paper is organised as follows. Section 2, reviews the related work on compositional entailment and different embedding models. In Section 3, we lay out the methodology behind MuG models, in terms of training methods, compositional entailment predictions, and model's parameters (such as the composition strategy). Section 4 describes training and evaluation datasets, and other implementation details. In Section 5, we present and analyse results on compositional entailment, graph completion and distributional similarity. Section 6 presents a discussion on the overall findings of the work, and how they fit in the current literature.

## 2 Background and Related Work

**Compositional entailment** A niche of work exists on phrase-level entailment, mostly focusing on adjective-noun (AN) phrases (e.g., *brown dog* entails (⊨) *animal*). Baroni et al. (2012) used non-intensional adjectives solely in the form of AN ⊨ N. Kober et al. (2021) used AN phrases as a data augmentation technique to improve lexical entailment classification. Recently, Bertolini et al. (2022) introduced PLANE, a benchmark to train and evaluate models on phrase-level adjective-noun entailment, which will be used in this work. All instances of the dataset are built out of true (noun (N), hypernym (h(N))) pairs, modified by an adjective (A). Items can take three entailment structures (or inference types (ITs)): AN ⊨ N, AN ⊨ h(N), and AN ⊨ Ah(N). Instances are then automatically annotated using rules defined by the three classes of English adjectives: intersective (I), subsective (S) and intensional (O). Table 1 summarises PLANE's instances, classes, and annotation schema. The work showed how LLMs struggle to solve PLANE without supervision, and that the mechanism supporting out-of-distribution generalisation is poorly linguistically grounded, as it notably depends on non-linguistic subword tokens.

| Inference Type | Intersective | Subsective | Intensional |
|---|---|---|---|
| 1 AN ⊨ N | ✓ | ✓ | ✗ |
| 2 AN ⊨ h(N) | ✓ | ✓ | ✗ |
| 3 AN ⊨ Ah(N) | ✓ | ✗ | ✓ |

Table 1: PLANE annotation rules. Schema of how the interaction between each adjective class and inference type shapes the positive (✓) - negative (✗) value of a true noun (N) – hypernym (h(N))) entailment (⊨) pair.

**Knowledge-graph Embedding (KGE) Models** Multiple ways of encoding hypernymy and other entailment relationships with different transformations, including rotation and reflection, have been investigated (Balažević et al., 2019; Chami et al., 2020). Proposed models learn representations of entities and representations that encode a mapping of entities to their hypernyms. For example, we can learn representations of the entities *dog* and *animal* and the relationship *ISA* such that when the *ISA* transformation is applied to the representation of *dog*, we would expect to be close to the representation of *animal*. Among all, hierarchical relationships such as hypernymy were found to be best modelled by rotations (Chami et al., 2020).

Bertolini et al. (2021) showed that syntax-sensitive composition of adjective-noun phrases can be carried out by modelling syntactic relationships with geometric transformations. To form a phrase's encoding, such as *brown dog*, one or more of the constituent representations (according to the syntactic relationship between them) is transformed before combination. The work also tested multiple transformations, including attention, and found reflection to best model syntax.

**Joint-Embedding models (JEM)** Our work bears resemblances with work merging textual and KG data (Alsuhaibani et al., 2018; Roy and Pan, 2020; Wang et al., 2020). A more detailed survey of recent work in this area is provided in Roy and Pan (2020). Here, we note that Toutanova et al. (2015) add specific syntactic-triplets extracted from text, like $(Obama, \texttt{nsubj}, President)$ to the original

KG. These injected triplets are hence used only as a form of data augmentation. Alsuhaibani et al. (2018) expand GloVe's (Pennington et al., 2014) loss to incorporate knowledge from a KG, creating a new joint objective function. In contrast to our work, the scope was to use KG data to enhance distributional embeddings. Wang et al. (2020) propose a robust attention-based model that incorporates textual and KG information in parallel, using one encoder for each source. A mutual attention component is then used to combine the outputs of the two encoders. In this case, similarly to our experimental setting, the scope was to improve the performance from the KGE perspective. Shwartz et al. (2016) propose to augment a hypernym classification model using a PathLSMT, based on syntactic relations. Vashishth et al. (2019) incorporated syntactic and semantic graphs using a large graph convolutional network. However, the two modalities were never mixed within the same architecture, since joint models produced poor results.

## 3 Methodology

### 3.1 Multi-Graph (MuG) Models

Most mixed-sourced approaches use different architectures or objectives to model each source of data. Here, we propose to use the same model to encode two types of graphs, syntactic and knowledge-based. Specifically, we propose the Multi-Graph (MuG) Model which will be used to simultaneously encode entailment relationships from knowledge-graphs and distributional relationships from syntactic corpus data. While previous work has shown that these relationship types can be encoded independently in models based on geometric transformations (Chami et al., 2020; Bertolini et al., 2021) we propose a training method which will allow a single model to encode both types of relationship and thus use each to generalise the other. For example, if a model knows that *vehicle* is a hypernym of *car*, can it learn from the syntactic relationships in parsed corpus data, what predictions to make about the hypernyms of *red car*, *small car* and *fake car*?

To investigate which architecture is better suited to learn a MuG model, we study the three KGE architectures introduced by Chami et al. (2020), namely RotE (rotation), RefE (reflections) and AttE (which uses attention to combine rotations and reflections). Since rotation was found to best encode KG relations (Chami et al., 2020), and reflection to better model syntax (Bertolini et al., 2021), we expect that an AttE combining both rotation and reflection will be the best architecture for a MuG model.

### 3.2 Training Methods

To train Multi-Graph models (MuGs), we propose two training methods, `static` and `altern`, using the same architecture and weights, yet separately considering the two data sources in the training phase.

**`static`** Straightforwardly, `static` trains a MuG model by feeding it first one complete data source and then the other. Specifically, a selected model is first trained with syntactic graphs and then with the knowledge-graph.

**`altern`** The `altern` method takes a dynamic approach to the training. Training is alternated at regular intervals between the two different data sources. This adds an extra hyperparameter to the model, *every*, which we have kept stable at 5 samples, that dictates the frequency with which the two training data sources alternate. All other model hyperparameters (e.g., total epochs) are kept stable and equally distributed across the data sources. Note that `static` could be considered as an extreme version of `altern` where *every* is set to the size of the first training data source multiplied by the number of epochs.

### 3.3 Predicting compositional entailment

Compositional entailment is framed as a binary classification task where models have to label $(c_1, c_2)$ tuples such as *(red car, vehicle)*. To score these tuples we propose to make use of each architecture's scoring $s(head, relation, tail)$ and sigmoid $(\sigma(\cdot))$ functions. The proposed classification function is presented in Equation 1:

$$C(c_1, r, c_2) = \begin{cases} 1 & \text{if } \sigma(s(c_1, r, c_2)) >= 0.5 \\ 0 & \text{else} \end{cases} \quad (1)$$

$s(h, r, t)$ is the model-specific scoring function (see Chami et al. (2020); Bertolini et al. (2021)). $r$ is always considered to be the $hyponym$ relation. Given the nature of the task, one or both of each $(c_1, c_2)$ tuple components can contain a phrase. To generate these, we use the composition strategies from Bertolini et al. (2021) (adopting average instead of simple sum):

**add** simple addition: constructed by averaging the base representations of the constituent words

**Rh** Root-as-head: the syntactic root of the phrase is seen as the head of the dependency triple (e.g., <*dog*,amod,*brown*>) and is modified by the geometric transformation in the composition process

**Rt** Root-as-tail: the syntactic root of the phrase is seen as the tail of the dependency triple (e.g., <*brown*,$\overline{\text{amod}}$,*dog*>) and is not modified by the geometric transformation in the composition process

**BiD** Bi-directional: constructed by adding the representations obtained using **Rh** and **Rt**

### 3.4 Syntax Modelling

In contrast to Bertolini et al. (2021), we consider the composition strategy as another interchangeable aspect of a MuG model. The decision traces back to the difference between the two forms of the compositionality principle (Partee et al., 1995). If syntax is indeed a crucial feature of compositionality, then a model with a syntax-aware composition will yield better results. Otherwise, no differences should be observed.

## 4 Experimental Setup

Our investigation focuses on two main questions. First, can MuGs in fact manipulate both composition and inference? To test this, we will compare MuG and KGE models on a compositional entailment task. Second, what ability, if any, will be penalised or completely sacrificed, in order for a model to tackle compositional entailment? To answer this question, MuGs will be compared to KGE on a standard graph completion task, and to distributional models trained on syntactic graphs (SyGs) on multiple similarity benchmarks.

### 4.1 Tasks and Benchmarks

**PLANE** To test MuG and KGE on compositional entailment, we sample five validation and test splits from the portion of PLANE (Bertolini et al., 2022) that contains items also included in WN18RR (Bordes et al., 2013) and `text8`, available here[1]. Since preliminary experiments showed all models heuristically assigned a positive label to items with inference type 3 (e.g. *red car* $\models$ *red vehicle*), we only sampled items with inference types 1 (AN $\models$ N) and 2 (AN $\models$ h(N)). In each split, the ratio of positive and negative items is kept balanced, and so is each (noun, hypernym) tuple for every adjective.

**KG and Similarity Judgements** To compare MuG and KGE models on the uni-gram level, we adopt a standard filtered graph completion task (Chami et al., 2020). The performance of SyG and MuG models is compared using the same benchmarks from Bertolini et al. (2021). These include four lexical similarity tasks (Simlex (Hill et al., 2015), MEN (Bruni et al., 2014), WS353-sim, WS353-rel (Agirre et al., 2009)), and a compositional one (ML10) (Mitchell and Lapata, 2010), further divided into three syntactic classes (Adjective-Nouns, Verb-Objects, Noun-Nouns).

### 4.2 Implementation

We adopt the source code from Chami et al. (2020) to train each model. Using the hyperparameters from Chami et al. (2020) and Bertolini et al. (2021), we trained a set of three architectures: AttE, RefE, RotE. As training data for each MuG model, we follow Bertolini et al. (2021) and adopt a sense-stripped version of WN18RR as KG, and a parsed version of `text8` as syntactic graph. We use PLANE validation splits to tune hyperparameters for each combination of training method (KGE, MuG-`altern`, MuG-`static`), architecture (AttE, RefE and RotE), and composition strategy (add, Rh, Rt, BiD). Best hyperparameters are presented in Appendix A. Experiments are run on an NVIDIA GeForce RTX 3090 GPU.

## 5 Results

### 5.1 Compositional entailment

Table 2 reports average accuracies (± standard error) obtained by different models on the five test sets generated from PLANE. The close-to-random performance (50%) observed for KGE models — trained solely with the knowledge-graph — is to be expected, since the overlap between training data and PLANE is fairly scarce, especially with respect to adjectives. Furthermore, Bertolini et al. (2021) already showed how KGE models perform poorly on compositional benchmarks, especially with respect to adjective-noun phrases.

Overall, MuG models perform only marginally better than KGEs. The best-performing model is based on the attention architecture, trained with the `altern` method and makes use of a syntax-aware composition strategy (Rh). These results are in

[1] https://github.com/lorenzoscottb/IWCS_2023

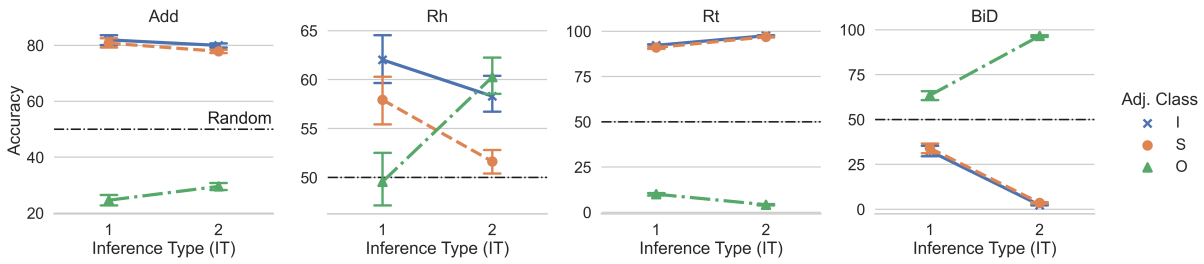


Figure 1: Models analysis. Break-down of the different AttE-altern models performance (mean accuracy ± standard error from different test splits), divided by adjective class (hue), composition strategy (columns) and inference type (IT, x-axis).

| Method | Model | Composition | Accuracy |
|---|---|---|---|
| KGE | AttE | add | 49.8 ± 0.2 |
| | RefE | add | 51.1 ± 0.2 |
| | RotE | add | 50.9 ± 0.2 |
| MuG-altern | AttE | add | 53.9 ± 0.4 |
| | | Rh | 56.2 ± 0.5 |
| | | Rt | 50.7 ± 0.1 |
| | | BiD | 49.1 ± 0.2 |
| | RefE | add | 51.3 ± 0.5 |
| | | Rh | 51.6 ± 0.4 |
| | | Rt | 50.1 ± 0.0 |
| | | BiD | 45.1 ± 0.4 |
| | RotE | add | 51.1 ± 0.3 |
| | | Rh | 51.7 ± 0.4 |
| | | Rt | 50.2 ± 0.0 |
| | | BiD | 45.2 ± 0.7 |
| MuG-static | AttE | add | 51.9 ± 0.4 |
| | | Rh | 53.3 ± 0.3 |
| | | Rt | 51.5 ± 0.3 |
| | | BiD | 47.1 ± 0.3 |
| | RefE | add | 53.3 ± 0.3 |
| | | Rh | 53.4 ± 0.2 |
| | | Rt | 50.9 ± 0.1 |
| | | BiD | 47.7 ± 0.4 |
| | RotE | add | 53.6 ± 0.3 |
| | | Rh | 54.2 ± 0.2 |
| | | Rt | 50.7 ± 0.1 |
| | | BiD | 47.4 ± 0.4 |

Table 2: Compositional entailment results. Accuracy scores (mean ± standard error) obtained by different combinations of training methods, architectures and composition strategies on different test–splits, generated from PLANE.

line with the two main hypotheses, suggesting that attention would better handle the two sources of data and that explicitly modelling syntax leads to more reliable compositional representations. Interestingly, the very same syntax-aware strategy (Rh) is also used by RotE-static, which seems to be the second-best performing model. However, aside from AttE-altern-Rh (and RotE-static-Rh) MuG models seem to generally struggle to correctly classify an item for compositional entailment. Hence, we now propose an in-depth analysis of what seems to be the best MuG model, comparing its behaviour to other AttE-altern models (i.e., tuned with different composition strategies), to better understand its prediction processes.

**Model analysis** Figure 1 breaks down the performance of AttE-altern models by composition strategies (columns), adjective class (hue) and inference type (x-axis). Overall, the figure shows that aside AttE-altern-Rh, models present a strongly heuristical behaviour, as suggested by the widespread lack of per-split variance (error bars). More specifically, add and Rt models produce almost exclusively positive predictions, as suggested by the very high performance with intersective (I) and subsective (S) adjectives. AttE-altern-BiD predictions seem to be slightly affected by the inference types (IT), fluctuating between random (under IT 1), and only-negative predictions (under IT 2). On the contrary, AttE-altern-Rh's results appear notably more complex, and suggest a strong interaction between inference type and adjective class, at least with respect to subsective and intensional adjectives. Recall that, since we have focused on IT 1 (AN $\models$ N) and 2 (AN $\models$ h(N)), intersective (I) and subsective (S) adjectives always have a positive label, while intensionals (O) are always associated with a negative label. As shown, when dealing with intersective (I) adjectives, the model is minimally impacted by the IT. The performance remains well above chance with items like *red car* $\models$ *car* (IT 1) and a *red car* $\models$ *vehicle* (IT 2).

On the other hand, the performance is significantly shaped by the inference type when instances contain subsective (S) or intensional (O) adjectives. More specifically, the performance of intensional (O) adjectives, always associated with negative la-

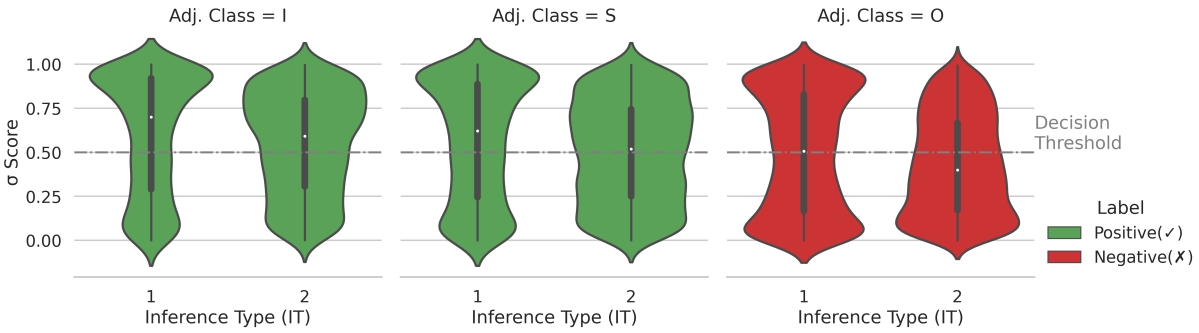


Figure 2: Distribution of the predicted scores of AttE-`altern`-Rh, divided by adjective class (columns), inference type (x-axis), and labels (hue). Dashed lines indicate the decision threshold, as in Equation 1.

bels, jumps from random to 60% moving from IT 1 to IT 2. In other words, the model better identifies scenarios like *fake car* $\not\models$ *vehicle* than *fake car* $\not\models$ *car*. The opposite happens for subsective (S) items. The model is better at classifying instances like *big car* $\models$ *car* than *big car* $\models$ *vehicle*. The fact that intersective (I) and intensional (O) adjectives produce virtually identical results on IT 2 instances, despite carrying opposite labels, while subsective's (S) performance drops to chance (although having the same label as intersective's adjectives) suggests two conclusions. First, the model's behaviour is not random or heuristical. Second, in contrast with previous evidence (Boleda et al., 2013), the theoretical distinction between adjective classes is likely reflected in the model's representations.

To understand if similar results derive from similar behaviours, Figure 2 summarises the model's prediction distribution after applying the sigmoid function in Equation 1. The plots of Figure 2 show two distinct patterns. Considering inference type 1 (i.e., AN $\models$ N), a large number of scores are towards the boundaries of the interval, generating peaky distributions. Distributions become increasingly bimodal moving through the three adjective classes (I, S and O). This suggests the model is often reasonably confident about the decision being made. On the other hand, the predictions under IT 2 (i.e., AN $\models$ h(N)) generate notably flatter distributions. Intersective (I) and intensional (O) adjectives do maintain a peak towards one of the boundaries, but the model is much less confident about decisions on IT 2 for all adjective classes.

We will now investigate if MuG models in general (i.e., not just AttE-`altern`-Rh) remain competitive with their KGE and SyG counterparts, starting with graph completion (Section 5.2), followed by distributional similarity benchmarks (Sections 5.3 and 5.4).

## 5.2 Graph completion

Figure 3 compares the performance of KGE and MuG models on the graph completion task. Error bars report the standard error obtained from collapsing MuG models tuned on different composition strategies. Overall, KGEs always outperform MuG models, while MuGs trained with the `static` method appear to notably outperform the ones trained with the `altern` method. This suggests that the recency of the KG training (i.e., the `static` method) is indeed influential in obtaining good results in the graph completion task. Figure 6 further breaks down the results, and compares the performance of MuGs against the amount of negative samples used in training. For comparison with the main results (Figure 3), a dashed grey line reports the best score obtained by a KGE model.

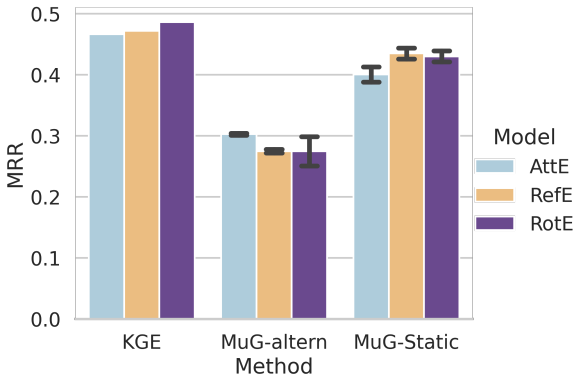


Figure 3: Mean reciprocal rank (MRR) scores of KGE and MuG models on the graph completion task. Error bars report standard error, obtained collapsing models trained with different composition strategies.

The figure shows how the optimal performance of the two training methods is reached at very different amounts of negative samples. Mug-`static`

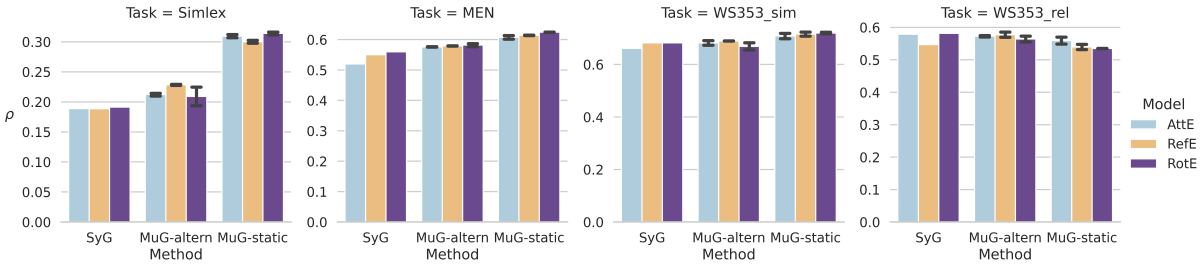


Figure 4: Spearman $\rho$ for SyG and MuG models on word–level similarity judgement tasks. Error bars report standard error obtained collapsing models tuned on different composition strategies.

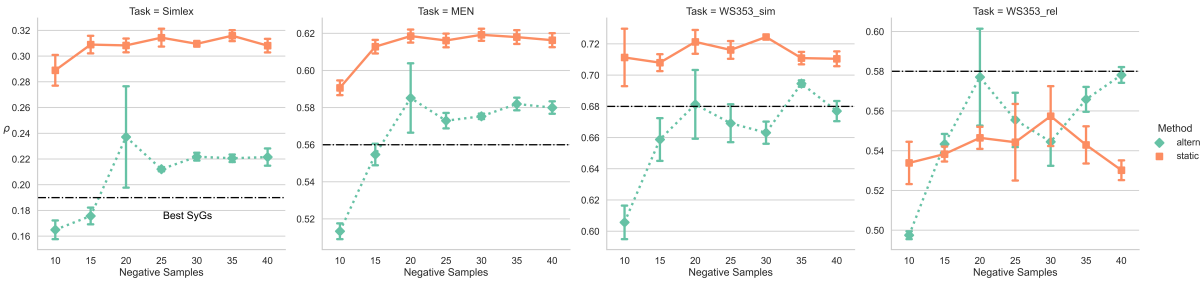


Figure 5: Lexical similarity with respect to the negative samples used during training. For comparison with Figure 4, a dashed black line outlines best results obtained by a SyG model.

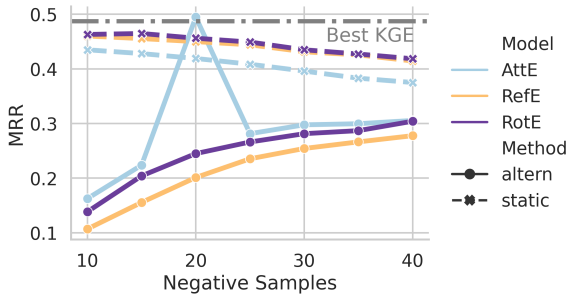


Figure 6: Graph completion analysis with respect to negative samples used during training.

models require few negative samples and are negatively affected by increasing amounts. On the other hand, the performance of models trained with the `altern` method increases with the number of negative samples used for training. However, other than a seemingly spurious peak at 20 negative samples, the performance obtained by `altern` models remains far from competitive. In line with results from Chami et al. (2020), RotE and RefE outperform AttE with the `static` method. Lastly, we note that the best AttE model from Table 1 is not the outlier observed in Figure 6.

### 5.3 Lexical similarity

We now consider whether MuGs remain competitive with SyGs (i.e., distributional models trained with syntactic graphs). Spearman's $\rho$ is used to measure the correlation between model's predictions and human judgements on similairty benchmarks. The first comparison is presented in Figure 4. Results are divided with respect to training methods (x-axis) and trained models (hue). Error bars reflect the standard error produced by MuG models tuned with different composition strategies. Overall, MuGs tend to outperform SyGs, especially MuG-`static`, suggesting that KG data helps with lexical similarity tasks. A notable exception is WS353-rel, which uses relatedness (e.g., *journey* is related to *car*) rather than similarity. The KG training data is taken from WordNet, thus including many examples of hypernym/hyponym pairs which one might expect to help more with similarity. However, Bertolini et al. (2021) found a generally poor performance of KGE models on lexical similarity tasks. Altogether, these results suggest that, even in the `static` training, the KG data and distributional information were successfully merged, leading to a performance which cannot be achieved by one of the data sources on their own.

Similarly to Section 5.2, Figure 5 shows how the number of negative samples impacts the performance. In this case, both training methods are positively impacted by higher negative samples, although the effect remains more marked for `altern` models. WS353-rel aside, `static` models appear to consistently outperform SyGs and

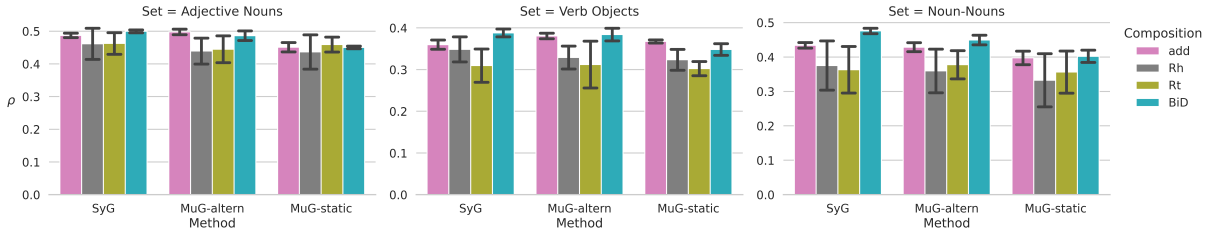


Figure 7: Spearman $\rho$ for SyG and MuGs (divided by composition strategy) on the different subsets of the ML10 dataset. Error bars report standard errors obtained collapsing results from different architectures.

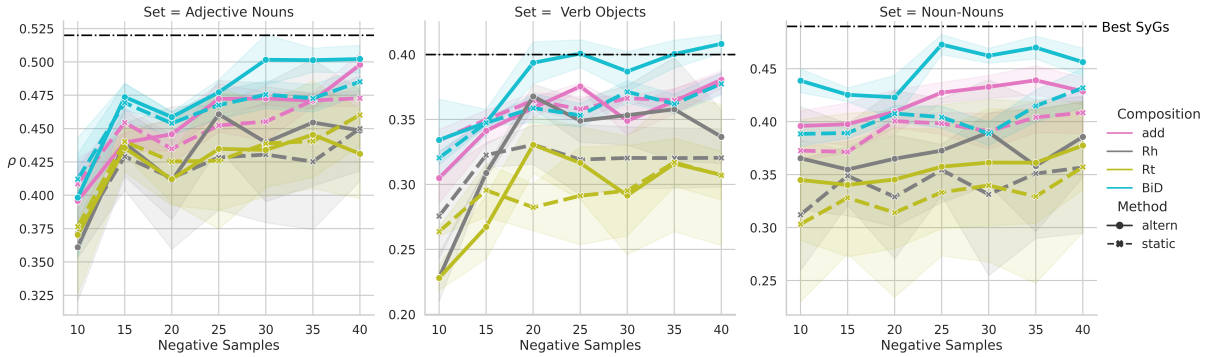


Figure 8: Compositional similarity with respect to number of negative samples used during training. For comparison with Figure 7, a dashed black line outlines best results obtained by a SyG model.

seem notably more robust, while `altern` models require a high number of negative samples.

### 5.4 Compositional similarity

A similar analysis is proposed for compositional similarity. Figure 7 summarises the results of best-performing models on PLANE against SyG models on the ML10 datasets. Results are split between the adjective-noun (AN), verb-object (VO), and noun-noun (NN) subsets. In this case, the focus is on training methods (x-axis) and composition strategy (hue). Compared to lexical similarity results, MuG models don't seem to outperform SyG models, but they remain a competitive alternative. Contrary to the previous results, the best performance with respect to MuG models is achieved via the `altern` training method. With respect to composition strategy, results seem to support Bertolini et al. (2021) findings: add and BiD are the best and most stable composition strategies, with BiD outperforming add. It is interesting to note that ML10 is based on bi-gram instances (e.g., how similar *hot tea* is to *cold water*), which is comparable to PLANE instance having inference type 3 (e.g., *hot tea* $\models$ *hot liquid*), that no model could solve in the compositional entailment task (see Section 4.1). The fact that MuG-`altern` models remain competitive to SyGs on ML10 suggests that their issue under IT 3 is more related to the manipulation of the hypernym relation, rather than a systematic problem of each model.

Figure 8 presents a last negative samples analysis. For comparison, a dotted line signals the best SyGs' results. As for lexical similarity, results indicate once more that performance improves with the negative samples' rate. Note that, contrary to the results on compositional entailment, Rh's performance is fairly poor across the board.

## 6 Discussion

Our work introduced MuGs, a set of embedding models learnt from multiple graph-based sources. Under specific and predicted conditions (i.e., using an attention-based model and syntax-aware composition), MuGs can be shown to simultaneously tackle compositionality and inference with some success. Experiments revealed that MuGs tuned for compositional entailment are competitive distributional models, with respect to both lexical and compositional similarity, yet struggle with graph completion. Our analysis suggests that a considerable part of the trade-off can be explained by the negative samples rate used for training. The best MuG model at compositional entailment, AttE-`altern`-Rh, was tuned on validation data to 35 negative samples. As summarised by the analy-

sis in Figures 6, 5, and 8, whilst similarity tasks, especially compositional ones, also benefit from high negative samples rate, graph completion tends to require low negative samples (and the `static` training method) to achieve the best performance.

Of note, the compositional entailment experiment presented in this work can also be interpreted with respect to knowledge-graphs. Despite a different evaluation method (accuracy instead of rank), the proposed task is a type of graph completion. The evaluation is still binary and requires the manipulation of hierarchical structures through the hypernym relation. Hence, MuGs can be interpreted as compositional KGE models.

Indeed, an LLM like BERT can achieve better results on compositional entailment as defined by PLANE. However, it can only do so with direct supervision, and relying on an effective yet not theoretically-sound mechanism (Bertolini et al., 2022). Since MuG are trained only with uni-grams, our approach to phrase-level inference (i.e., compositional entailment) is fully unsupervised, requires significantly less training data, and has a deeper connection with the principle of compositionality (Partee et al., 1995). On each compositional task, linguistically-sound word encodings composed with a syntax-aware non-linear composition strategy yielded the best performance. Moreover, when a model does not present a strongly heuristical behaviour, we found that the three adjective classes pose as many different challenges to the models, similarly to what already observed in Bertolini et al. (2022).

## 7 Conclusions and Future Work

In this work, we introduced Multi-Graph embedding models (MuGs), a set of models trained on syntactic and knowledge-graphs. Under specific conditions, MuGs can partially tackle compositional entailment, making use of syntax-aware composition, based on attention. We provided evidence that MuGs are competitive with distributional counterparts on lexical and compositional similarity benchmarks. Our analysis suggested that compositionality is supported by a higher number of negative samples, and connected this evidence to the low performance of MuGs on graph completion. Future work will have to primarily focus on developing a training strategy to overcome the negative samples issue, able to obtain a better integration of the two sources of data and produce a more stable performance across tasks. Lastly, MuG models will have to be tested on other types of compositional entailment (e.g., noun-noun or verb-object phrases), as well as full sentences.

## Acknowledgements

This research was supported by the EPSRC project *Composition and Entailment in Distributed Word Representations* (grant no. 2129720), and the EU Horizon 2020 project HumanE-AI (grant no. 952026). We thank the anonymous reviewers for their helpful comments and suggestions.

## References

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and WordNet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27, Boulder, Colorado. Association for Computational Linguistics.

Mohammed Alsuhaibani, Danushka Bollegala, Takanori Maehara, and Ken-ichi Kawarabayashi. 2018. Jointly learning word embeddings using a corpus and a knowledge base. *PLOS ONE*, 13(3):1–26.

Ivana Balažević, Carl Allen, and Timothy Hospedales. 2019. Multi-relational poincaré graph embeddings. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, page 4463–4473, Red Hook, NY, USA. Curran Associates Inc.

Marco Baroni, Raffaella Bernardi, Ngoc-Quynh Do, and Chung-chieh Shan. 2012. Entailment above the word level in distributional semantics. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 23–32, Avignon, France. Association for Computational Linguistics.

Lorenzo Bertolini, Julie Weeds, and David Weir. 2022. Testing large language models on compositionality and inference with phrase-level adjective-noun entailment. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4084–4100, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Lorenzo Bertolini, Julie Weeds, David Weir, and Qiwei Peng. 2021. Representing syntax and composition with geometric transformations. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3343–3353, Online. Association for Computational Linguistics.

Gemma Boleda, Marco Baroni, The Nghia Pham, and Louise McNally. 2013. Intensionality was only alleged: On adjective-noun composition in distributional semantics. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013) – Long Papers*, pages 35–46, Potsdam, Germany. Association for Computational Linguistics.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

E. Bruni, N. K. Tran, and M. Baroni. 2014. Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, 49:1–47.

Ines Chami, Adva Wolf, Da-Cheng Juan, Frederic Sala, Sujith Ravi, and Christopher Ré. 2020. Low-dimensional hyperbolic knowledge graph embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6901–6914, Online. Association for Computational Linguistics.

Ishita Dasgupta, Demi Guo, Andreas Stuhlmüller, Samuel J. Gershman, and Noah D. Goodman. 2018. Evaluating compositionality in sentence embeddings. In *Proceedings of the 40th Annual Conference of the Cognitive Science Society*, pages 1596–1601, Madison, WI.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Nam Do and Ellie Pavlick. 2021. Are rotten apples edible? challenging commonsense inference ability with exceptions. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2061–2073, Online. Association for Computational Linguistics.

Gottlob Frege. 1892. Über sinn und bedeutung. *Zeitschrift für Philosophie Und Philosophische Kritik*, 100(1):25–50.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. SimLex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Daniel Keysers, Nathanael Schärli, Nathan Scales, Hylke Buisman, Daniel Furrer, Sergii Kashubin, Nikola Momchev, Danila Sinopalnikov, Lukasz Stafiniak, Tibor Tihon, Dmitry Tsarkov, Xiao Wang, Marc van Zee, and Olivier Bousquet. 2020. Measuring compositional generalization: A comprehensive method on realistic data. In *International Conference on Learning Representations*.

Thomas Kober, Julie Weeds, Lorenzo Bertolini, and David Weir. 2021. Data augmentation for hypernymy detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1034–1048, Online. Association for Computational Linguistics.

Bill MacCartney and Christopher D. Manning. 2008. Modeling semantic containment and exclusion in natural language inference. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 521–528, Manchester, UK. Coling 2008 Organizing Committee.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive Science*, 34(8):1388–1429.

Francis Mollica, Matthew Siegelman, Evgeniia Diachek, Steven T. Piantadosi, Zachary Mineroff, Richard Futrell, Hope Kean, Peng Qian, and Evelina Fedorenko. 2020. Composition is the core driver of the language-selective network. *Neurobiology of Language*, 1(1):104–134.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Barbara Partee et al. 1995. Lexical semantics and compositionality. *An Invitation to Cognitive Science: Language*, page 311–360.

Ellie Pavlick and Chris Callison-Burch. 2016. Most "babies" are "little" and most "problems" are "huge":

Compositional entailment in adjective-nouns. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2164–2173, Berlin, Germany. Association for Computational Linguistics.

Ellie Pavlick and Tom Kwiatkowski. 2019. Inherent disagreements in human textual inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana. Association for Computational Linguistics.

Arpita Roy and Shimei Pan. 2020. Incorporating extra knowledge to enhance word embedding. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4929–4935. International Joint Conferences on Artificial Intelligence Organization. Survey track.

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving hypernymy detection with an integrated path-based and distributional method. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2389–2398, Berlin, Germany. Association for Computational Linguistics.

Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, and Michael Gamon. 2015. Representing text for joint embedding of text and knowledge bases. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509, Lisbon, Portugal. Association for Computational Linguistics.

Shikhar Vashishth, Manik Bhandari, Prateek Yadav, Piyush Rai, Chiranjib Bhattacharyya, and Partha Talukdar. 2019. Incorporating syntactic and semantic information in word embeddings using graph convolutional networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3308–3318, Florence, Italy. Association for Computational Linguistics.

Yashen Wang, Huanhuan Zhang, Ge Shi, Zhirun Liu, and Qiang Zhou. 2020. A model of text-enhanced knowledge graph representation learning with mutual attention. *IEEE Access*, 8:52895–52905.

Lang Yu and Allyson Ettinger. 2020. Assessing phrasal representation and composition in transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4896–4907, Online. Association for Computational Linguistics.

Lang Yu and Allyson Ettinger. 2021. On the interplay between fine-tuning and composition in transformers. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2279–2293, Online. Association for Computational Linguistics.

# A Hyperparameters

| Method | Architecture | Composition | Negative Samples |
|---|---|---|---|
| KGE | RefE | add | 10 |
| KGE | RotE | add | 10 |
| KGE | AttE | add | 10 |
| static | RefE | add | 20 |
| static | RefE | Rh | 20 |
| static | RefE | Rt | 40 |
| static | RefE | BiD | 35 |
| static | RotE | add | 40 |
| static | RotE | Rh | 40 |
| static | RotE | Rt | 35 |
| static | RotE | BiD | 20 |
| static | AttE | add | 30 |
| static | AttE | Rh | 30 |
| static | AttE | Rt | 40 |
| static | AttE | BiD | 10 |
| altern | RefE | add | 40 |
| altern | RefE | Rh | 40 |
| altern | RefE | Rt | 35 |
| altern | RefE | BiD | 40 |
| altern | RotE | add | 40 |
| altern | RotE | Rh | 40 |
| altern | RotE | Rt | 35 |
| altern | RotE | BiD | 15 |
| altern | AttE | add | 40 |
| altern | AttE | Rh | 35 |
| altern | AttE | Rt | 40 |
| altern | AttE | BiD | 35 |

Table 3: Final hyperparameters for each model.