Compositionality as an Analogical Process: Introducing ANNE

Giulia Rambelli, Emmanuele Chersoni, Philippe Blache, Lenci Alessandro

Compositionality as an Analogical Process: Introducing ANNE

Giulia Rambelli
University of Pisa
giulia.rambelli@phd.unipi.it

Emmanuele Chersoni
The Hong Kong Polytechnic University
emmanuelechersoni@gmail.com

Philippe Blache
Aix-Marseille University/CNRS
philippe.blache@univ-amu.fr

Alessandro Lenci
University of Pisa
alessandro.lenci@unipi.it

Abstract

Usage-based constructionist approaches consider language a structured inventory of *constructions*, form-meaning pairings of different schematicity and complexity, and claim that the more a linguistic pattern is encountered, the more it becomes accessible to speakers. However, when an expression is unavailable, what processes underlie the interpretation? While traditional answers rely on the principle of compositionality, for which the meaning is built word-by-word and incrementally, usage-based theories argue that novel utterances are created based on previously experienced ones through *analogy*, mapping an existing structural pattern onto a novel instance.

Starting from this theoretical perspective, we propose here a computational implementation of these assumptions. As the principle of compositionality has been used to generate distributional representations of phrases, we propose a neural network simulating the construction of phrasal embedding as an analogical process. Our framework, inspired by word2vec and computer vision techniques, was evaluated on tasks of generalization from existing vectors.

1 Introduction

While the generative tradition has dominated linguistic research for over half a century, the last decades have seen the emergence of an alternative paradigm in linguistics and cognitive sciences, which goes under the name of *usage-based models of language* (Langacker, 1987; Croft, 1991, 2001; Givón, 1995; Tomasello, 2009; Bybee, 2010), a variety of approaches grounded on the idea that linguistic structures emerge and are shaped through the use of language. Their claim is that language is not different from any other cognitive domain: Linguistic structures are not the result of a specific-language function but are explainable as the implementation of domain-general processes (Ibbotson, 2013). The usage-based position shares the fundamental assumption of Construction Grammar (Hoffman and Trousdale, 2013): language consists of meaningful and symbolic form–meaning mappings, called *constructions*. Words, idiomatic expressions (e.g., *kick the bucket* —"to die" or *jog <someone's> memory* —"to refresh <someone's> memory") and highly general and productive syntactic patterns (e.g., ditransitive structures) are all constructions varying along a continuum of schematicity and complexity.

In particular, usage-based constructionist approaches emphasize the notion of frequency: combinations that are more frequently encountered become more accessible (perhaps because they are stored in memory) and are preferred. Indeed, if the language system derives from language use, it follows that how often a speaker encounters a particular linguistic expression will affect the system itself. This assumption implies that any sequence of words – if used frequently enough – can be a construction, even if there are no idiosyncrasies of form and meaning (Goldberg, 2006). However, it is impossible to store any possible word combinations a speaker has or will ever produce. The traditional answer relies on the *principle of compositionality*: the meaning of a complex expression is entirely determined by its structure and the meanings of its constituents – once we specify what the parts mean and how they are put together, there is no more leeway regarding the meaning of the whole (Partee, 2004). Usage-based theories favor a different explanation: novel utterances are created based on previously experienced utterances thanks to the cognitive process of *analogy*.

The ability to make analogies – that is, to map familiar relations from one domain of experience to another – is a fundamental ingredient of human intelligence and creativity (Hofstadter, 2001). In the linguistic domain, analogy depends on similarity in form and meaning between constructions,

Proceedings of the Workshop on Cognitive Aspects of the Lexicon, pages 78–96

https://doi.org/10.26615/978-954-452-056-4_010

whether these constructions are of a concrete type or an abstract type: a novel instance is compared to those stored in our long-term memory to infer the new representation. In this perspective, the acceptability of a novel item is a gradient that depends on the extent of similarity to prior uses of a construction (Bybee, 2010). In a more radical stance, Ambridge (2020) proposed disregarding completely abstraction: unwitnessed forms are produced and comprehended "by on the fly analogy" across multiple stored exemplars. Without denying the existence of abstract representations, we also assume that analogical mechanisms play a key role in explaining systematic processes of language productivity.

This paper aims to articulate the hypotheses introduced above in computational terms. We address two interconnected questions: How can we represent (lexicalized) constructions? Is it possible to replicate the interpretation-as-analogy mechanism in computational terms? Specifically, we investigate how to model constructions as well as analogy-based compositionality using Distributional Semantic Models (DSMs). DSMs represent the lexicon in terms of vector spaces, where a lexical target is described in terms of a vector (also known as embedding) built by identifying in a corpus its syntactic and lexical contexts (Lenci, 2018).

As a first approximation, we decided to consider constructions any kind of frequent pairs of words linked by a syntactic relation. Traditionally, building distributional representations beyond individual words, such as phrases and sentences, is the focus of *Compositional Distributional Semantic Models*. Their proposed methodologies try to derive the meaning of an expression from the meanings of the sentence's constituents (Baroni et al., 2014): the simplest CSDMs represent words as vectors and obtain sentence vectors with sum or product operations between constituent vectors (Mitchell and Lapata, 2010), while more complex models represent predicates with matrices and tensors (Baroni and Zamparelli, 2010; Coecke et al., 2010; Baroni et al., 2014; Paperno et al., 2014) or reproduce the compositionality operation by means of a neural architecture learning so-called sentence embeddings (Socher et al., 2012; Cheng and Kartsaklis, 2015). It is interesting to notice that most distributional models for phrases/constructions/sentences assume more or less explicitly the principle of compositionality, while the idea that units above the word level could be stored and retrieved via analogy/similarity mechanisms has rarely been explored. [1]

The experiment presented here distances itself from these approaches, following a more usage-based perspective. Suppose frequently experienced word sequences are, to some extent, stored in memory, and the organization and productivity of language are understood as the result of analogical processes between form and meaning in this structured inventory of constructions. In that case, new phrases could be constructed by analogy with stored linguistic patterns. We propose a neural network model to infer a distributional representation of a new syntactic phrase by preserving the structural information encoded in the embeddings representing previously stored, high-frequency phrases.

As the main contributions of the paper, i) we introduce a new DSM in which both lemmas and syntactic relations in the form of <*head, dependent, syntactic role*> triples have a unique distributional representation; ii) we propose an analogical model to create the distributional embeddings of new relations by applying deep-learning techniques, and evaluate different architectures in terms of generalization and systematicity; iii) we discuss the implications of our analogical model from a theoretical and computational perspective.

2 Relational Embeddings

The first step consisted in developing a DSM for lexicalized constructions. We represent the meaning of phrases following a holistic approach (Turney, 2012): as a numeric vector can represent nouns like *space* and *race*, in the same way, phrases like *space race* are associated with a unique embedding. For our goal, we built embeddings corresponding to triples <*head, dependent, role*>, assuming that these vectors should keep track of the syntactic relation between words. For this reason, we called these **Relational Embeddings** (RelEmbs), and we assume they represent the meaning of lexicalized constructions.

We built our semantic space using `word2vecf` (Levy and Goldberg, 2014), a modification of the skip-gram model introduced by Mikolov et al. (2013a). While the original implementation assumes bag-of-words contexts, i.e., the model keeps

[1] Some partial exceptions are instance-based distributional models (Jones and Mewhort, 2007; Jamieson et al., 2018; Crump et al., 2020) and distributional models of event knowledge that store event occurrences in the form of syntactic graphs (Chersoni et al., 2019, 2021).

track of word counts and disregards the grammatical details and the word order, `word2vecf` allows us to use arbitrary context features. In detail, we extracted <target, context> occurrences from the concatenation of ukWaC and a 2018 dump of English Wikipedia, parsed using CoreNLP (Manning et al., 2014): targets are both words and <*head, dependent, role*> triples (e.g., <*bark, dog, nsubj*>), while context is always an open-class word (noun, verb, adjective) occurring with the target in the sentence within a window $\pm$ 10 (ten words before and ten words after the head of the relation excluding the dependent). Word2vecf parameters are reported in Appendix A. We built our DSM considering only words and relations with a frequency equal to or larger than 100 and filtering out <target, context> pairs with a frequency less than 20; lastly, we kept only <*head, dependent, role*> triples with a frequency $\geq$ 1,000, where both the head and the dependent lemmas have a frequency $\geq$ 10,000. This strategy is consistent with the idea that holistic representations of complex constructions are stored only for substantially frequent items. The final space contains 127,739 word embeddings and 173,496 RelEmbs, for a total of 301,235 items.

Semantic space evaluation We tested the quality of the semantic space over some most common benchmarks for the intrinsic evaluation of word and phrase embeddings. It is worth mentioning that we are not aiming at beating traditional DSMs, but rather at carrying out a general evaluation of the goodness of our distributional representations of lexicalized constructions.

For word embeddings, we ran the standard Word Similarity/Relatedness task using the well-known **WordSim-353** (Finkelstein et al., 2001) and **MEN** (Bruni et al., 2014). The task is to compute the cosine similarity between two words (e.g., *cup* and *mug*) and verify how their score correlates with the similarity rate given by humans. We also evaluated the DSM against **FAST** (Evert and Lapesa, 2021), a free associations dataset. The goal of this multiple-choice task is to determine the most frequent associate for a given stimulus among three candidates (e.g., which word between *neck, apple, wine* is most associated with *giraffe*?). As a baseline, we computed the performance of a DSM trained with the original word2vec Skip Gram model (Mikolov et al., 2013a) on the same concatenation of corpora.

Results are reported in Table 1. Considering the first task, we observe that Spearman's correlation scores for the baseline are a bit higher than our DSM in all settings, except for the MEN dataset. However, the differences are not statistically significant.[2] It is worth noticing that similarity results are better than relatedness results, showing the same trend reported in Agirre et al. (2009). We observe an opposite performance for the classification task: our space consistently beats the baseline, and the difference is statistically significant.

Dataset	RelEmbs.w	baseline	Coverage
WS353-all	0.684	0.721	333/353
WS353-sim	0.734	0.75	195/203
WS-353-rel	0.628	0.675	236/252
MEN	0.774	0.735	3000/3000
FAST-EAT	0.786***	0.737	5877/7610
FAST-USF	0.725***	0.719	4057/4719

Table 1: Word embeddings evaluation. On top: Spearman's correlation scores for Word Similarity/Relatedness task. Bottom: Accuracy scores for Free Association task. *** = p<0.01 using McNemar test.

Moving to the relational embeddings, we used the Mitchell et al. (2010) Phrase Similarity dataset (**ML10**), which includes 324 English phrase pairs, tripartite in noun phrases, verb phrases, and adjective phrases. Given two expressions (e.g., *general principle* and *basic rule*), the task consists in comparing the cosine similarity between the two corresponding vectors and then correlating the score with the human similarity rating. As a baseline, we represented the phrases as the sum of the word2vec vectors used for word embedding evaluation. Table 2 reveals that correlation scores are not homogeneous among the different sets: the noun phrase subset achieves a higher score (0.635) compared to the other two sets, whose score is lower than 0.5. Moreover, baseline results are consistently better than our model and are statistically significant for the AN subset.

Dataset	RelEmbs	baseline	Coverage
ML-vo	0.499	0.599	99/108
ML-nn	0.635	0.716	99/108
ML-an	0.462	0.683**	102/108

Table 2: Relational embeddings evaluation. Spearman's correlation scores for Phrase Similarity task. ** = p<0.01 using Fisher r-to-z transformation test.

[2] p>0.1, the p-value is computed with Fisher's r-to-z transformation, one-tailed test

Qualitative analysis Results in Table 2 suggest that RelEmbs perform worst than the baseline in the phrase similarity test. To gain more insight, we selected some problematic pairs from the ML10 dataset and manually inspected the k-nearest neighbors, i.e., the most similar words by cosine similarity. Let us look at the pair *reduce amount* and *cut cost*: the two expressions are judged very similar (6.55), but their cosine similarity is just 0.41. However, their distributional neighbors are coherent and somehow systematic in the sense that they are similar to relational embeddings in which the same head or dependent word occurs. So, *reduce amount* is mostly similar to *increase amount*, *reduce waste, a person reduce, large amount, high amount*; on the other hand, the neighbors of *cut cost* are *reduce cost, improve efficiency, increase profit, lower cost, save money*. Similar observations are for nominal phrases, like *government leader* and *health minister*. While ML10 reports a high score (4.95), the cosine similarity between the two is quite low (0.43). However, this is explainable by observing their neighbors. In the first case, *health minister* is similar to other types of ministers (*health secretary, transport minister, environment minister, minister for health*); conversely, *government leader* is more associated with situations (*invite a leader, include a leader*) or other offices (*chief whip, head of the committee, regional leader*) associated to leaders. In other words, while the phrases refer to government members, the two roles are not the same (and functions also differ).

To sum up, the qualitative analysis of the neighbors reveals that RelEmbs form a semantically coherent space, even though they do not outperform the baseline in the phrase similarity task.

3 Analogical Neural Network for Embeddings

Usage-based theories of language assume that systematic processes of language productivity can be explained mainly by analogical inferences rather than by sequential compositional operations. In this perspective, we present a system to expand the coverage of the RelEmbs space simulating the construction of phrasal meaning as an analogical process via deep learning techniques.

3.1 Architecture

We aim to infer a distributional representation of a new syntactic phrase (ANALOGICAL TARGET) by preserving the structural information encoded in an existing relational embedding (ANALOGICAL BASE). For simplicity, we represent this process using the familiar four-term formalism.[3] Approximately, solving the analogy A:B::C:? requires a system that generates an appropriate embedding to make a valid analogy: if we need to infer an embedding for the target phrase *drink cider* using *drink water* as the base, we can reformulate the analogy as: $water : drink\_water_{dobj} :: cider :?$

We framed the problem of analogy completion as a regression task: the aim is to build a phrasal vector given the embeddings of the other expressions in the analogy. While word embeddings have been widely employed to perform analogy by addition and subtraction of word embeddings (Mikolov et al., 2013b; Gladkova et al., 2016), we argue that directly training a deep neural network on the task of analogy completion could provide better results, as already proposed by Reed et al. (2015) for visual analogy-making. We named our novel neural network model as **Analogical Neural Network for Embeddings** (**ANNE**).

In detail, we implemented a feed-forward neural network architecture with one hidden layer: the model is trained to learn a function $f : R^{2D} \rightarrow R^{D}$ that maps an input vector *x* to a generated embedding *y* of dimension D (where *D*=300), preserving the structural properties of the selected base. The input vector x should incorporate the analogical base (e.g., *drink water*), and the new argument (e.g., *cider*). We tested two possible combinations: i) the input vector is the concatenation between the analogical base and the new argument (CONCAT, Figure 1a); ii) we compute the difference between the analogical base and the argument in the same relation; the resulting vector is concatenated to the new argument vector (DIFF; Figure 1b). The intuition below the DIFF input representation is that we apply some aspects of Mikolov's analogical operation with the nonlinearities and supervision offered by a neural network.

We developed several variants of this network, each with a distinct objective function. The basic architecture (SIMPLE) is trained to maximize the cosine similarity between the original and predicted embedding. However, ANNE should not simply create a vector similar to the actual instance in the DSM but also learn the relational structure

[3] It is, however, doubtful that linguistic analogies are computed in this way at the brain level (Bybee, 2010).

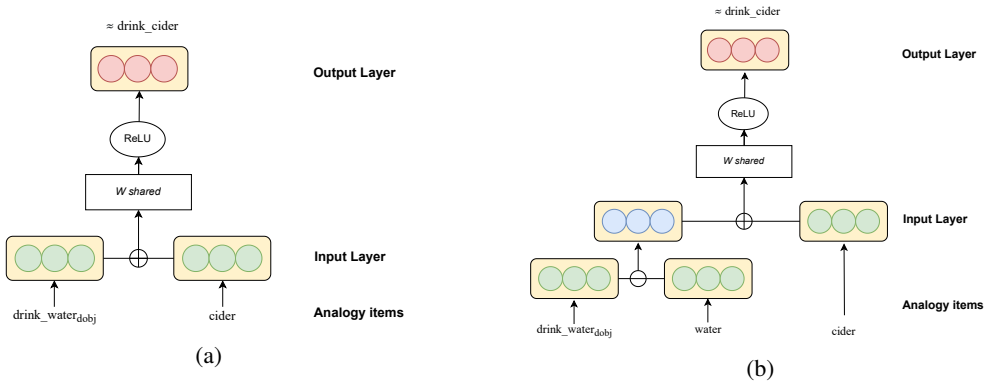


Figure 1: ANNE architecture with CONCAT (a) and DIFF (b) input. The $\bigoplus$ indicates a vector concatenation, while $\ominus$ indicates vector subtraction.

of the base and transpose it to the generated embedding. To this end, we implemented a multiple losses function, which combines the SIMPLE loss with a new loss aimed at minimizing the difference of the similarity between the relational vector and the embeddings of its words computed for the base and the target. For instance, if the similarity between *drink water* and *drink* is 0.60 and the similarity between *drink water* and *water* is 0.49, similar scores should be obtained by computing the similarity of the output vector with the vectors of *drink* and *cider*. Therefore, the network is trained to generate RelEmbs that preserve the same relations with its components as the ones in the analogical base. To compare the similarity scores we tested three functions (cf. Appendix B): the Mean Squared Error (MULTIMSE), the mean of the scores difference (MULTIAVG), or the hinge loss function (MULTIHINGE).

Training the network We selected analogical base-target pairs from the Relational Embeddings attested in space to train the neural network. We assembled the dataset as follows: given a relation <*head, dependent, syntactic relation*>, we selected all RelEmbs with the same head and syntactic relation and calculated the similarity between the arguments of each pair, keeping only those pairs with the cosine similarity between arguments ≥ 0.4 and the cosine similarity between RelEmbs ≥ 0.6. We chose the filter on similarity heuristically: the idea is that the candidate targets should be somehow similar with respect to their analogical bases but not the exact synonyms. The final dataset consisted of 350,404 items and was divided into **Train** and **Test** parts (respectively, 95% and 5%).

To verify the analogy-solving capability of the network, i.e., its ability to generalize from the base, we kept some analogical pairs out of the training step. The resulting data (named **Test-unseen**), comprises 3,201 pairs (cf. Appendix C). This dataset should verify the network's performance when encountering new relations, which is to say, evaluate the model's generalization ability. The training setup configurations are reported in Appendix D.

3.2 RSA Evaluation

A preliminary evaluation of ANNE consisted of computing the similarity between relational embeddings attested in the DSM and embeddings analogically generated from ANNE attested in the Test and the Test-unseen datasets. We applied the Representational Similarity Analysis (RSA; Kriegeskorte et al. (2008); Kriegeskorte and Kievit (2013)), a computational technique that allows us to compare heterogeneous representations in higher-order spaces. The core idea is simple: instead of directly correlating representations of stimuli in different representation spaces, we compute how similar representations are between pairs of stimuli in each space, and the resulting similarity matrices are then compared. As we are interested in understanding how similar the original and generated embeddings are, we created a pair of matrices where rows are the vectors representing the analogical targets from a test set and columns correspond to a subset of the RelEmbs vocabulary.[4] Following Lenci et al. (2022), we randomly sampled 100 disjoint sets of 1,000 lexemes, ran RSA analyses on each sample, and then computed the average score.

Table 3 reports Spearman's ρ between the similarity matrix computed with the original RelEmbs

[4] A matrix with 301,235 columns would be computationally too expensive.

and the matrix with vectors generated with ANNE. We can observe that the models reach similar results for the two test data, even if the Test-unseen scores are always slightly lower than those for the Test set. Overall, the SIMPLE model reaches the best scores (0.851 for Test and 0.835 for Test-unseen), while MULTIAVG performs the worst (reaching just 0.739 for Test-unseen with DIFF input). However, the average correlation of all models is significantly high. As a baseline, we also performed the vector offset method (Mikolov et al., 2013a). RSA correlation scores significantly drop (0.734 and 0.71 for the Test and Test-unseen, respectively). The worst architecture (MULTIAVG diff) is still better than the baseline for Test ($p < 0.1$), but not for Test-unseen. The best architecture (SIMPLE$_{concat}$) is different from the baseline with $p < 0.001$.[5] This result corroborates our assumption that the ANNE architecture is better at generating analogical vectors than a simple vector operation.

	TEST	TEST-UNSEEN
simple$_{concat}$	0.851	0.835
simple$_{diff}$	0.848	0.834
multiHinge$_{concat}$	0.819	0.805
multiHinge$_{diff}$	0.806	0.788
multiAVG$_{concat}$	0.782	0.754
multiAVG$_{diff}$	0.77	0.739
multiMSE$_{concat}$	0.835	0.82
multiMSE$_{diff}$	0.824	0.804
baseline	0.734	0.71

Table 3: Average Spearman's correlation between original and analogically generated semantic spaces computed with RSA on 100 random samples of 1, 000 words for Test and Test-unseen datasets.

4 Compositionality vs. Idiomaticity

Finally, we present a series of analyses to evaluate the meaning encoded in analogically-generated embeddings. We hypothesize that the best-generated embedding should keep the same relationship among components as the base (*systematicity*). As a counterproof, we also generated embeddings from idiomatic expressions. In this case, we expect analogies with idiomatic bases to give odd results in the semantic space because of their reduced compositionality and systematicity. The results should answer the following questions: What are the characteristics of analogically-generated embeddings? How does the type of input (concatenation or difference) affect the final representation? What loss functions are better at retaining the same structural relation of the base, while at the same time generalizing from the original embedding?

Data The analogical bases employed are 44 verbal phrases (22 idioms from Libben and Titone (2008) + 22 compositional manually picked from frequent relations) and 24 nominal compounds (12 idiomatic + 12 compositional) selected from the Noun Compound Senses dataset (Cordeiro et al., 2019) and the dataset by Reddy et al. (2011).

For each phrase, we manually chose a relation similar to the base but not attested in the vocabulary space, with the same head and syntactic role. For example, given the relation <*market*$_N$, *fish*$_N$, *compound*> ("a fish market"), we replaced the noun *fish* (i.e., the dependent) with the noun *shrimp*; expressly, the relation <*market*$_N$, *shrimp*$_N$, *compound*> ("a shrimp market") is not attested in RelEmbs vocabulary. The final dataset consists of 68 analogical pairs, half with an idiomatic base and half with the compositional counterpart.

	Idiomatic → Target	Compositional → Target
VN	*break ice → break chunk*	*break bone → break finger*
NC	*loan shark → credit shark*	*reef shark → atol shark*

Table 4: Examples of analogical pairs (the idiomatic/compositional base on the left, the target on the right of the arrow).

4.1 Analysis 1: Correlation of the Similarities with the Components

To evaluate if and how the ANNE configurations are generating embeddings systematically, we observed if the similarities between the relational embedding and those of the component words are similar for both the analogical base and the generated target. The assumption is that the embedding generated by ANNE should have the same internal structure as the base from which it is inferred: that is, the relationship between the phrase meaning and the meaning of its components should be systematically retained in the generated distributional vector. This idea can be approximated by the similarities between the RelEmbs and its parts: if the similarity between *break (a) bone* and *break* is 0.4 (*simHead* score) and the similarity with the dependent *bone* is 0.42 (*simDep* score), comparable scores should

[5] p-values for Fisher's r-to-z transformation, one-tailed test.

be obtained computing the similarities of *break (a) finger* with *break* and *finger*, respectively.

We computed the cosine similarity scores for all ANNE implementations. We assume that the best architecture (i.e., the one that best fits our theoretical hypothesis) should be the one that has i) comparable similarity distributions for compositional bases and derived targets (for both word components), and ii) different (or incoherent) similarity distributions for targets generated from idiomatic bases. By looking closely at the plots in Appendix E, we observe that each architecture produces different outputs. Among all models, MULTIAVG is the one performing worse (plots in (c) and (d)): the generated embeddings have high similarities with the dependent component in both idiomatic and compositional cases, possibly because they retain too much distributional information from dependent words used to generate the new embedding. The MULTIMSE (plots in (e) and (f)) and the SIMPLE (plots in (a) and (b)) losses show a similar behavior: they give a high *simHead* and *simDep* to vector generated from idiomatic targets. This result shows that, when deriving a new literal phrase meaning from a figurative one should be impossible, the models largely rely on attribute similarity instead of truly learning a relation. In this sense, the MULTIMSE$_{diff}$ (plots in (f)) model is the only one that perfectly respects our hypothesis (distributions should be the same for targets from compositional bases but different for targets with idiomatic bases). Conversely, the MULTIHINGE model (plots in (g) and (h)) reduces the impact of the dependent word, as proved by the fact that the mean similarity of *simDep* is lower for the target (orange) than for the base (blue).

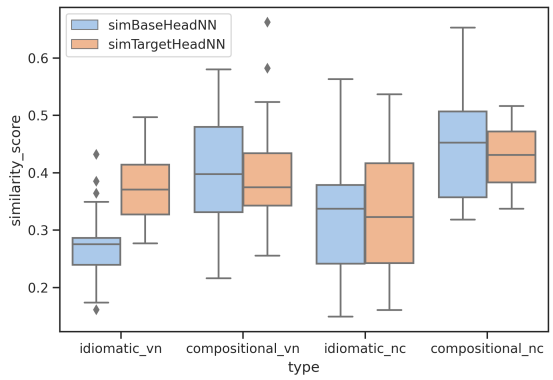


Figure 2: Distribution of the similarities between RelEmbs and their head for MULTIMSE$_{diff}$. Blue boxplots refer to the base embedding, orange to the analogically-generated ones.

4.2 Analysis 2: Intersection of Neighbors

As a complementary measure to cosine similarity, we computed the intersection between the 50-nearest neighbors of i) the base and the generated target, and ii) the generated target and the respective head/dependent.[6] The first measure tells us how much information the analogical embedding retains from its base: the higher the value, the higher their similarity, so it could be that the network did not generalize from the input. The second measure should say how much the analogy moved the distribution towards the component meanings. Appendix F reports the results as a series of heatmaps.

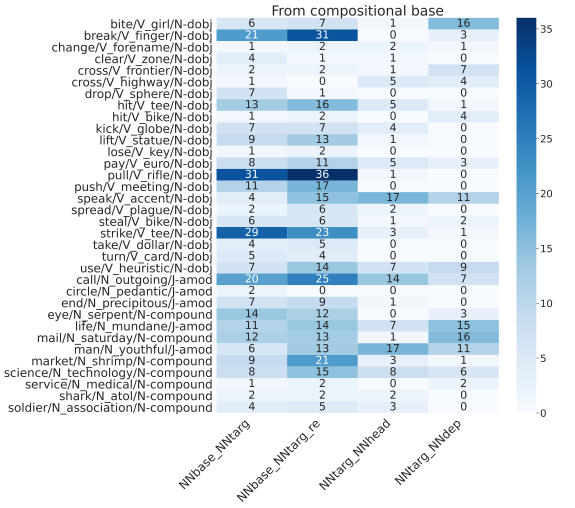


Figure 3: Heatmap for MULTIMSE$_{concat}$ shows the intersection between the neighbors of the analogically-generated embedding and the base ($NN_{base}NN_{targ}$, **re* only RelEmbs)), the head ($NN_{targ}NN_{head}$), and the dependent ($NN_{targ}NN_{dep}$).

Embeddings generated from idiomatic bases have no shared neighbors with the vectors of their heads or dependents: as the network was not trained on this type of analogies (which are impossible), we expected the neural model to fail. What we can add, however, is that sometimes it generates a new embedding that has no common neighbors with either its base or components, sometimes it resolves the analogy by copying the distributional signature of the base. The only exception to this trend is MULTIAVG (subfigures (c) and (d)): we notice that some items, such as *shark credit*, *cockroach market*, and *gastropod mail*, partake many neighbors with their dependent, revealing that this

[6] For head and dependent embeddings, we only considered neighbors that are RelEmbs to limit the variability.

model is not generalizing correctly.

A more complex scenario appears if we consider the targets generated from compositional bases. As noticed above, ANNE with CONCAT input has more shared neighbors between the target and the base (first two columns), while this is not the case for the ANNE with DIFF input (see heatmaps in (a) and (b) as example). This finding is further proof that a neural network that takes as input the concatenation of vectors for the base and the target argument attempts to generate an embedding as close as possible to the input relational embedding. In other words, this type of input could negatively impacts ANNE in learning the correct inference.

4.3 Architectures' comparison

Previous analyses reveal that some parameter configurations are better than others. ANNE models that take as input the SIMPLE concatenation of the RelEmb base and the word embedding generate vectors too similar to the base, while modifying the base with an operation similar to Mikolov's vector offset produces better results. Overall, it seems that ANNE trained with MULTIHINGE and MULTIMSE losses (with DIFF input) induce more consistent and explainable results, while MULTIAVG is suboptimal for its tendency to generate embeddings similar to the target's dependent.

5 General Discussion

An open issue in DSMs is how distributional representations can be projected from the lexical level to the sentence or even discourse level. Most previous approaches have tried to solve this issue by explicitly relying on the classic principle of compositionality. Given the Fregean assumption that phrase meaning is a function of the meanings of its constituents, different computational strategies have been proposed to derive vectors for phrases by taking word embeddings as inputs.

In this paper, we proposed a new methodology grounded on a usage-based perspective: we tried to generate new distributional representations by implementing an analogical function in the form of a neural network. Word analogies have been used as a standard intrinsic evaluation task for measuring the quality of word (Mikolov et al., 2013c; Levy and Goldberg, 2014; Linzen, 2016) and sentence embeddings (Zhu and de Melo, 2020; Wang et al., 2021; Ushio et al., 2021b). However, the task is usually defined as a candidate retrieval: given an analogical proportion, find the correct completion from a list of candidates to solve the analogy. On the contrary, our aim is to generate a completely new embedding, similarly to what is done in reasoning and computer vision (Reed et al., 2015; Sadeghi et al., 2015; Upchurch et al., 2016; Ichien et al., 2021): the task consists in training deep learning models to recognize a relationship among two images and generate a transformed query representation (in this case, an image) accordingly. We believe that future investigations in linguistic analogies should pick up from this literature, and ANNE is a first attempt along this direction.

Our ANNE approach is not without limitations. One controversial aspect of ANNE is the choice of building the target by simply changing the argument in the relation. While it is not too problematic for verbal phrases, it raises questions for adjective-noun phrases and noun compounds. Consider the expressions *blue car* and *fast car*. Many things can be blue and not be a car, but not everything can be fast (e.g., **fast carrot*) because *fast* constrains the possible realizations of its head. A similar observation could be shown for noun compounds: in some cases, their meaning is related to both components (e.g., *bank account*), but sometimes their meaning retains aspects of one component (e.g., *head teacher*). To take into account the specificities of each type of phrases, we could train different ANNE architectures for each type of phrases.

The main difficulty is to balance relational and attributional similarity. Indeed, the use of a new item in a construction requires a great deal of relational knowledge (Gentner and Markman, 1997); nonetheless, the importance of similarity or shared attributes to linguistic analogy is not less vital (Bybee, 2010). A qualitative evaluation of analogical inferred embeddings reveals that analogy is easier to compute if the similarity between the entities in the syntactic relations is high. For instance, most all architectures build a good representation of *science technology* generated from *earth science*, maybe because there are lots of "topic science" expressions (cf. Table 5). Conversely, if attribute similarity is lower (i.e., the words between the base and the target are somewhat dissimilar), the analogical model is challenged. The neighbors of *pedantic circle* (derived from *literary circle*, cf. Table 6) are odd and incoherent with the expected meaning, maybe because the adjective *literary* is usually associated with a work of literature (an inanimate

	concat	diff
SIMPLE	*earth science*	*area of technology*
	apply science	*apply technology*
	marine science	*focus* ORGANIZ.
	new science	*include technology*
	area of technology	*area of engineering*
MULTIHINGE	*earth science*	*focus* ORGANIZ.
	new science	*area of technology*
	apply science	*apply technology*
	relate to technology	*electronic technology*
	area of technology	*create technology*
MULTIMSE	*earth science*	*area of technology*
	apply science	*include technology*
	area of technology	*focus* ORGANIZ.
	new science	*aspect of technology*
	area of engineering	*aspect of use*
MULTIAVG	*apply science*	*information technology*
	information technology	*apply technology*
	development in science	*area of technology*
	role of technology	*apply science*
	area of technology	*new technology*

Table 5: 5-nearest neighbors of *technology science* (compound) generated *earth science*.

	concat	diff
SIMPLE	*show (a) letter*	*guess* PERSON
	explain in letter	*extol (the) virtue*
	(a) disciple PERSON	*point_out* PERSON
	(a) letter address	*complain about* PERSON
	refer in (the) letter	*dismiss (an) idea*
MULTIHINGE	LOCATION *scholar*	*state for example*
	join on return	*extol (the) virtue*
	accompany (an) expedition	*join on return*
	await (the) return	*serve curacy*
	(a) letter address	*say in july*
MULTIMSE	*show letter*	*complain about* PERSON
	explain in letter	*guess* PERSON
	letter address	*extol (the) virtue*
	enlist aid	*point_out* PERSON
	(a) PERSON *demand*	*say according to* PERSON
MULTIAVG	*see before* PERSON	*see before* PERSON
	like (one's) style	*feel like* PERSON
	tell (a) girl	*tell (a) girl*
	tell about time	*realize* PERSON
	everyone tell(s)	*want (a) baby*

Table 6: 5-nearest neighbors of *pedantic circle* (amod) generated from *a literary circle*.

object), while *pedantic* collocates with a person. In these cases, different factors could contribute to the success or failure of the model, which should be further investigated.

The introduction of analogy as a strategy to derive meaning for novel expressions does not entail the entire suppression of compositional approaches. From a theoretical stance, not every expression can be built using analogical inference: if analogy fails, compositional operations switch over to guide interpretation. In this regard, the question should not be whether analogically-generated vectors are better than computationally-built ones, but when one mechanism is preferred to the other. Answering this question is challenging from both a psycholinguistic and computational stance. The issues related to computational models of analogy as a productive mechanism in language are theoretical before methodological. While it is true that the cognitive process of analogy represents a central mechanism in human cognition (Hofstadter, 2001), the problem in defining a linguistic theory that formalizes precisely what an analogy is and when it occurs is complex. In other words, it is hard to predict which analogies will actually be drawn and at what linguistic level (Behrens, 2017, p. 215). Ideally, future systems aiming at modeling language comprehension should be able to include this mechanism too. New benchmarks will have to be built with the aim of identifying analogical inferences. These datasets could also be valuable for behavioral analyses.

6 Conclusion and Future Works

In this paper, we presented a new approach that simulates the construction of phrasal meaning as an analogical process implemented with deep learning techniques. We proposed a distributional representation of constructional phrases and a model of generating new embeddings analogically rather than applying traditional compositional operations. We experimented with our analogical neural network to understand how it can generalize and be extendable to different scenarios. We argued that the proposed methodology could open the doors to new analyses in distributional semantics as well as in computational models of language processing.

The future research perspectives on ANNE are considerable. Firstly, we could build a more sophisticated phrasal representation using contextualized embeddings (Ethayarajh, 2019) based on Transformers (Vaswani et al., 2017; Devlin et al., 2019). Moreover, we should compare our Relembs with other phrasal representations representations (Shwartz, 2019; Alipoor and Schulte im Walde, 2020) and Relation Embeddings (Camacho-Collados et al., 2019; Ushio et al., 2021a). Moreover, while we performed analogy over preselected base-target pairs, we aim at investigating methods to automatically retrieve the best analogical candidate. Finally, we plan to evaluate ANNE's ability to model human behavior on more complex tasks regarding compositionality and language productivity.

References

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Pasca, and Aitor Soroa. 2009. A Study on Similarity and Relatedness Using Distributional and WordNet-based Approaches. In *Proceedings of NAACL-HLT*.

Pegah Alipoor and Sabine Schulte im Walde. 2020. Variants of vector space reductions for predicting the compositionality of English noun compounds. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4379–4387, Marseille, France. European Language Resources Association.

Ben Ambridge. 2020. Against Stored Abstractions: A Radical Exemplar Model of Language Acquisition. *First Language*, 40(5-6):509–559.

Marco Baroni, Raffaella Bernardi, and Roberto Zamparelli. 2014. Frege in Space: A Program of Compositional Distributional Semantics. *LiLT (Linguistic Issues in Language Technology)*, 9.

Marco Baroni and Roberto Zamparelli. 2010. Nouns Are Vectors, Adjectives Are Matrices: Representing Adjective-noun Constructions in Semantic Space. In *Proceedings of EMNLP*.

Heike Behrens. 2017. The Role of Analogy in Language Processing and Acquisition. *The Changing English Language: Psycholinguistic Perspectives*, pages 215–239.

Elia Bruni, Nam-Khanh Tran, and Marco Baroni. 2014. Multimodal Distributional Semantics. *Journal of Artificial Intelligence Research*, 49:1–47.

Joan L. Bybee. 2010. *Language, Usage and Cognition*. Cambridge University Press.

Jose Camacho-Collados, Luis Espinosa Anke, and Steven Schockaert. 2019. Relational word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3286–3296, Florence, Italy. Association for Computational Linguistics.

Jianpeng Cheng and Dimitri Kartsaklis. 2015. Syntax-aware Multi-sense Word Embeddings for Deep Compositional Models of Meaning. In *Proceedings of EMNLP*.

Emmanuele Chersoni, Enrico Santus, Alessandro Lenci, Philippe Blache, and Chu-Ren Huang. 2021. Not All Arguments Are Processed Equally: A Distributional Model of Argument Complexity. *Language Resources and Evaluation*, 55(4):873–900.

Emmanuele Chersoni, Enrico Santus, Ludovica Pannitto, Alessandro Lenci, Philippe Blache, and C-R Huang. 2019. A Structured Distributional Model of Sentence Meaning and Processing. *Natural Language Engineering*, 25(4):483–502.

Bob Coecke, Mehrnoosh Sadrzadeh, and Stephen Clark. 2010. Mathematical Foundations for a Compositional Distributional Model of Meaning. *arXiv preprint arXiv:1003.4394*.

Silvio Cordeiro, Aline Villavicencio, Marco Idiart, and Carlos Ramisch. 2019. Unsupervised Compositionality Prediction of Nominal Compounds. *Computational Linguistics*, 45:1–57.

William Croft. 1991. *Syntactic Categories and Grammatical Relations: The Cognitive Organization of Information*. University of Chicago Press.

William Croft. 2001. *Radical Construction Grammar: Syntactic Theory in Typological Perspective*. Oxford University Press.

Matthew Crump, Randall Jamieson, Brendan T Johns, and Michael N Jones. 2020. Controlling the Retrieval of General vs Specific Semantic Knowledge in the Instance Theory of Semantic Memory. In *CogSci*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL*.

Kawin Ethayarajh. 2019. How Contextual Are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings. In *Proceedings of EMNLP*.

Stefan Evert and Gabriella Lapesa. 2021. FAST: A Carefully Sampled and Cognitively Motivated Dataset for Distributional Semantic Evaluation. In *Proceedings of CONLL*.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing Search in Context: The Concept Revisited. In *Proceedings of International Conference on World Wide Web*, pages 406–414.

Dedre Gentner and Arthur B Markman. 1997. Structure Mapping in Analogy and Similarity. *American Psychologist*, 52(1):45.

Talmy Givón. 1995. *Functionalism and Grammar*. John Benjamins Publishing.

Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuoka. 2016. Analogy-based Detection of Morphological and Semantic Relations with Word Rmbeddings: What Works and What doesn't. In *Proceedings of the NAACL Student Research Workshop*.

Adele E. Goldberg. 2006. *Constructions at Work: The Nature of Generalization in Language*. Oxford University Press on Demand.

Thomas Hoffman and Graeme Trousdale, editors. 2013. *The Oxford Handbook of Construction Grammar*. Oxford University Press, Oxford.

Douglas R Hofstadter. 2001. Analogy as the Core of Cognition. *The Analogical Mind: Perspectives from Cognitive Science*, pages 499–538.

Paul Ibbotson. 2013. The Scope of Usage-based Theory. *Frontiers in Psychology*, 4:255.

Nicholas Ichien, Qing Liu, Shuhao Fu, Keith J Holyoak, Alan Yuille, and Hongjing Lu. 2021. Visual Analogy: Deep Learning versus Compositional Models. *arXiv preprint arXiv:2105.07065*.

Randall K Jamieson, Johnathan E Avery, Brendan T Johns, and Michael N Jones. 2018. An Instance Theory of Semantic Memory. *Computational Brain & Behavior*, 1(2):119–136.

Michael N Jones and Douglas JK Mewhort. 2007. Representing Word Meaning and Order Information in a Composite Holographic Lexicon. *Psychological Review*, 114(1):1.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*.

Nikolaus Kriegeskorte and Rogier A Kievit. 2013. Representational Geometry: Integrating Cognition, Computation, and the Brain. *Trends in Cognitive Sciences*, 17(8):401–412.

Nikolaus Kriegeskorte, Marieke Mur, and Peter A Bandettini. 2008. Representational Similarity Analysis-connecting the Branches of Systems Neuroscience. *Frontiers in Systems Neuroscience*, page 4.

Ronald W Langacker. 1987. *Foundations of Cognitive Grammar: Theoretical Prerequisites*, volume 1. Stanford University Press.

Alessandro Lenci. 2018. Distributional Models of Word Meaning. *Annual Review of Linguistics*, 4(1):151–171.

Alessandro Lenci, Magnus Sahlgren, Patrick Jeuniaux, Amaru Cuba Gyllensten, and Martina Miliani. 2022. A Comprehensive Comparative Evaluation and Analysis of Distributional Semantic Models. *Language, Resources and Evaluation*.

Omer Levy and Yoav Goldberg. 2014. Dependency-based Word Embeddings. In *Proceedings of ACL*.

Maya R Libben and Debra A Titone. 2008. The Multidetermined Nature of Idiom Processing. *Memory & Cognition*, 36(6):1103–1121.

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.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of ACL: System Demo*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013a. Distributed Representations of Words and Phrases and their Compositionality. *Advances in Neural Information Processing Systems*, 26.

Tomáš Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic Regularities in Continuous Space Word Representations. In *Proceedings of NAACL-HLT*.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013c. 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.

Jeff Mitchell and Mirella Lapata. 2010. Composition in Distributional Models of Semantics. *Cognitive Science*, 34(8):1388–1429.

Jeff Mitchell, Mirella Lapata, Vera Demberg, and Frank Keller. 2010. Syntactic and Semantic Factors in Processing Difficulty: An Integrated Measure. In *Proceedings of ACL*.

Denis Paperno, Nghia The Pham, and Marco Baroni. 2014. A Practical and Linguistically-motivated Approach to Compositional Distributional Semantics. In *Proceedings of ACL*.

Barbara H. Partee. 2004. *Compositionality in Formal Semantics: Selected Papers*. Explorations in Semantics. Blackwell Publishing Ltd.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An Imperative Style, High-performance Deep Learning Library. *Advances in Neural Information Processing Systems*, 32.

Siva Reddy, Diana McCarthy, and Suresh Manandhar. 2011. An Empirical Study on Compositionality in Compound Nouns. In *Proceedings of IJCNLP*.

Scott E Reed, Yi Zhang, Yuting Zhang, and Honglak Lee. 2015. Deep Visual Analogy-making. *Advances in Neural Information Processing Systems*, 28.

Fereshteh Sadeghi, C Lawrence Zitnick, and Ali Farhadi. 2015. Visalogy: Answering Visual Analogy Questions. *Advances in Neural Information Processing Systems*, 28.

Vered Shwartz. 2019. A systematic comparison of English noun compound representations. In *Proceedings of the Joint Workshop on Multiword Expressions and WordNet (MWE-WN 2019)*, pages 92–103, Florence, Italy. Association for Computational Linguistics.

Richard Socher, Brody Huval, Christopher D Manning, and Andrew Y Ng. 2012. Semantic Compositionality through Recursive Matrix-vector Spaces. In *Proceedings of EMNLP-CONLL*.

Michael Tomasello. 2009. The Usage-based Theory of Language Acquisition. In *The Cambridge Handbook of Child Language*, pages 69–87. Cambridge University Press.

Peter D Turney. 2012. Domain and Function: A Dual-space Model of Semantic Relations and Compositions. *Journal of Artificial Intelligence Research*, 44:533–585.

Paul Upchurch, Noah Snavely, and Kavita Bala. 2016. From A to Z: Supervised Transfer of Style and Content Using Deep Neural Network Generators. *arXiv preprint arXiv:1603.02003*.

Asahi Ushio, Jose Camacho-Collados, and Steven Schockaert. 2021a. Distilling relation embeddings from pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

Asahi Ushio, Luis Espinosa Anke, Steven Schockaert, and Jose Camacho-Collados. 2021b. BERT is to NLP what AlexNet is to CV: Can pre-trained language models identify analogies? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Online. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *Advances in Neural Information Processing Systems*, 30.

Yizhe Wang, Béatrice Daille, and Nabil Hathout. 2021. Caractérisation des relations sémantiques entre termes multi-mots fondée sur l'analogie (semantic relations recognition between multi-word terms by means of analogy). In *Actes de la 28e Conférence sur le Traitement Automatique des Langues Naturelles. Volume 1 : conférence principale*, pages 115–124. ATALA.

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.

A RelEmbs parameters

We use the skip-gram algorithm adopting the default configuration: no hierarchical softmax, 15 negative samples (how many negative contexts to sample for every correct one), and 300 as the vector dimension.

B ANNE losses

The basic architecture (SIMPLE) uses the cosine similarity between original and predicted vectors to make backpropagation. The CosineEmbeddingLoss[7] criterion from PyTorch library (Paszke et al., 2019) measures whether two inputs t and t' are similar or dissimilar using the cosine distance (*cos*):

$$\text{CEloss}(t, t', y) = \begin{cases} 1 - cos(t, t') & \text{if } y = 1 \\ max(0, cos(t, t')) & \text{if } y = -1 \end{cases} \quad (1)$$

The loss function takes as inputs t, t', and a label tensor y containing values (1 or -1). For our purposes, we set y=1, so the loss is *1 - cos(t, t')*: The closer the cosine value to 1, the more the two inputs are similar, and then the loss is closer to 0. The optimization strategy is to minimize the cost function, that is, obtaining a loss value near 0 for all items in the training set.

The MULTI-criterion loss function is defined by the general formula:

$$\begin{aligned} loss_{multi} = & \, CEloss(t, t') \\ & + g(CEloss(b, b_{head}), \\ & \quad CEloss(t', t_{head})) \\ & + g(CEloss(b, b_{dep}), \\ & \quad CEloss(t', t_{dep})) \end{aligned} \quad (2)$$

where t stands for the vector originally attested in RelEmbs space and t' corresponds to the output vector generated by the network; b represents the analogical base vector, $b_{head/dep}$ represents the vectors for the head and the dependent of the base (the same applies for $t_{head/dep}$). Finally, g(·) represents the function used to compare the phrase-argument similarity scores, which can be either the Mean Squared Error (equation 3), the mean of the scores difference (equation 4), or the hinge loss function (equation 5).

$$\text{MSE}(x, x') = (x - x')^2 \quad (3)$$

$$\text{AVG}(x, x') = mean(x - x') \quad (4)$$

$$\text{HINGE}(x, x') = max(0, x - x') \quad (5)$$

For each loss function, the cost derivative for the model's parameters (weight matrices W_1, bias vector b_1) is computed, and the appropriate parameters are updated through backpropagation.

[7] https://pytorch.org/docs/stable/generated/torch.nn.CosineEmbeddingLoss.html

C ANNE Test unseen preparation

We randomly selected 15 verbs, 15 nouns, and 15 adjectives attested in the RelEmbs vocabulary and we picked out from the original list all pairs in which one of these lemmas appeared. For instance, given the verb *study*, we saved in a separate file all pairs in which the verb occurs, such as $<study_V, aspect_N, dobj>$ ("to study the aspect") $\rightarrow <study_V, development_N, dobj>$ ("to study the development").

D ANNE Training Setup

Given the possible combinations of input type (CONCAT and DIFF) and losses functions (SIMPLE, MULTIMSE, MULTIAVG, and MULTIHINGE), we trained eight different versions of ANNE. All models were trained using 5-cross validation for 10 epochs with the Adam (Kingma and Ba, 2014) gradient descent, using a batch size of 25. Hyperparameter values equal for all models. The training was performed on a TITAN Xp GPU (12gb).

E Task1-Correlation of the component similarities of the base and the generated target

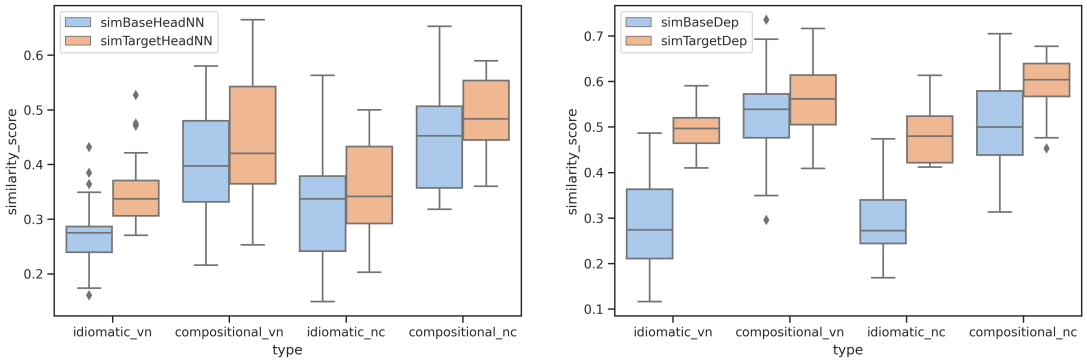
In order to visualize how these measures differ among architectures, we plotted the similarity scores using boxplots (Figure 4). Each subfigure represents the similarities computed over embeddings generated from a specific model architecture. The plot on the left refers to the RelEmb-head similarities; the plot on the right illustrates the RelEmb-dependent similarities. In each plot, we grouped boxplots for the type of base (idiomatic or compositional) and the syntactic type of phrase (verbal —VN —or nominal —NC). Finally, similarities are computed for both the base embedding (blue) and target embedding (orange).

F Task 2-Intersection of neighbors

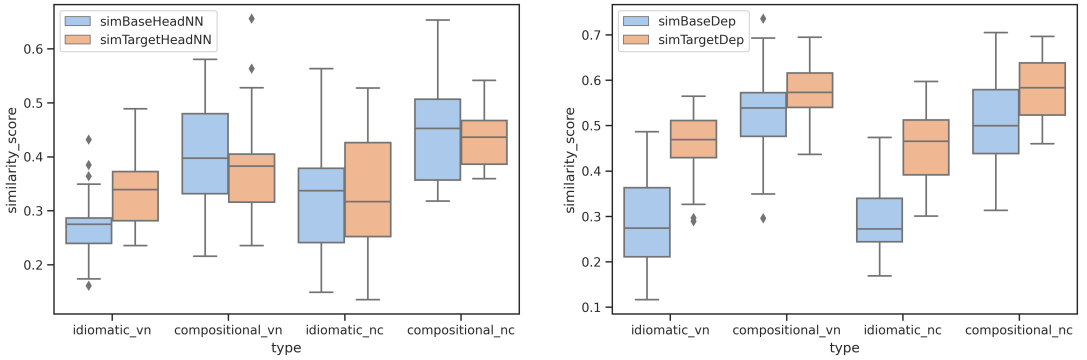
We propose here a visual aid to investigate ANNE behavior. Figure 5 groups a series of heatmaps. In each heatmap, rows correspond to a specific item from the dataset, while columns represent the intersection between the neighbors of:

- The base and the generated target (*NNbase_NNtarg*; $NNbase\_NNtarg_{re}$ considers only RelEmbs)
- The generated target and the respective head (*NNtarg_NNhead*) or dependent (*NNtarg_NNdep*) —for these, we consider only RelEmbs neighbors.

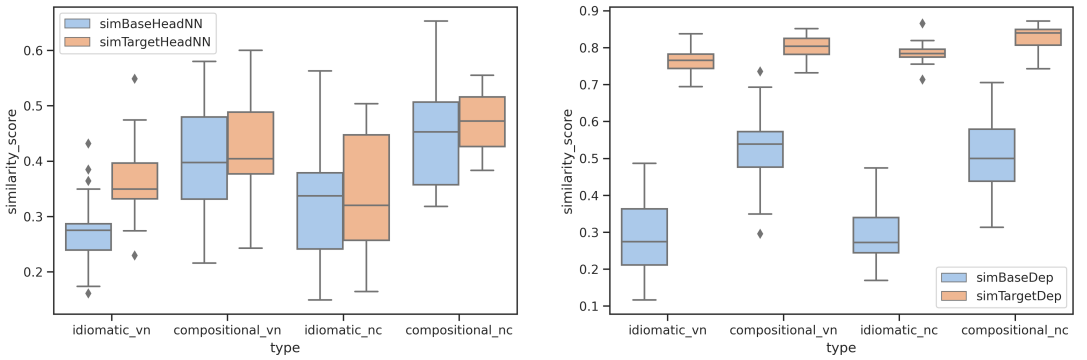
Numbers in the cells correspond to the number of neighbors retrieved. We present the results of analogical targets generated from a compositional (on the left) or idiomatic (on the right) base separately. Each subplot shows the results obtained for a specific model architecture.



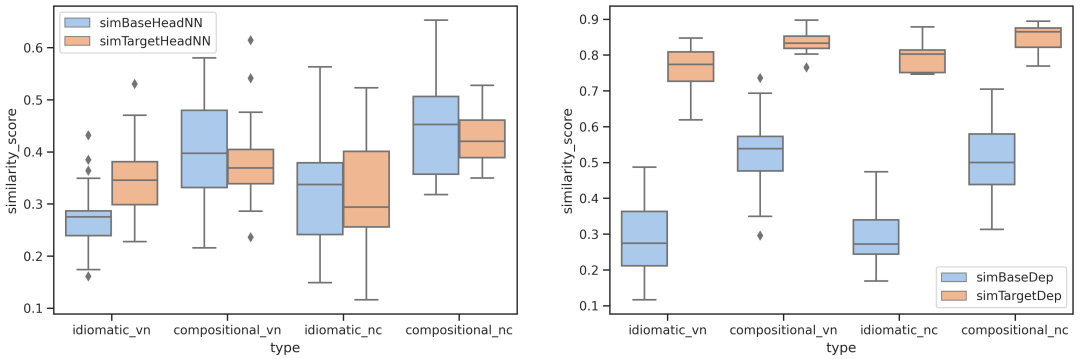
(a) SIMPLE with CONCAT input.



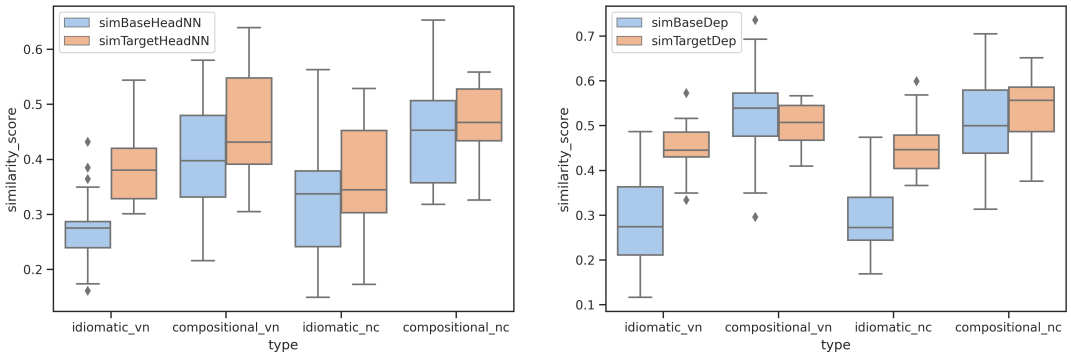
(b) SIMPLE with DIFF input



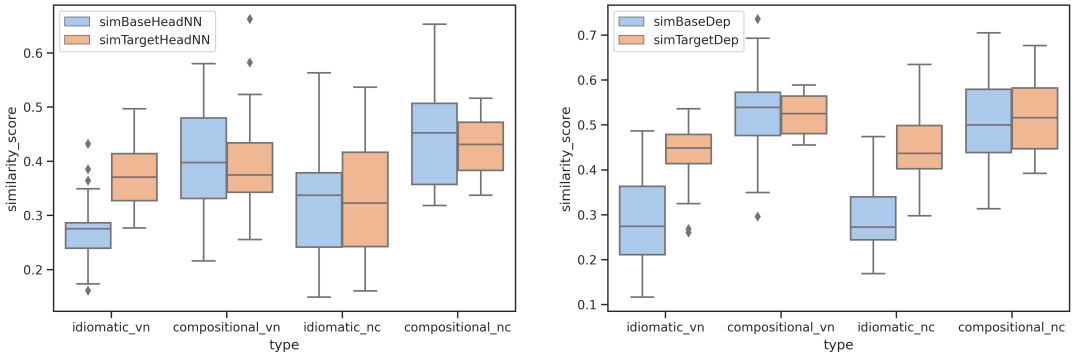
(c) MULTIAVG with CONCAT input



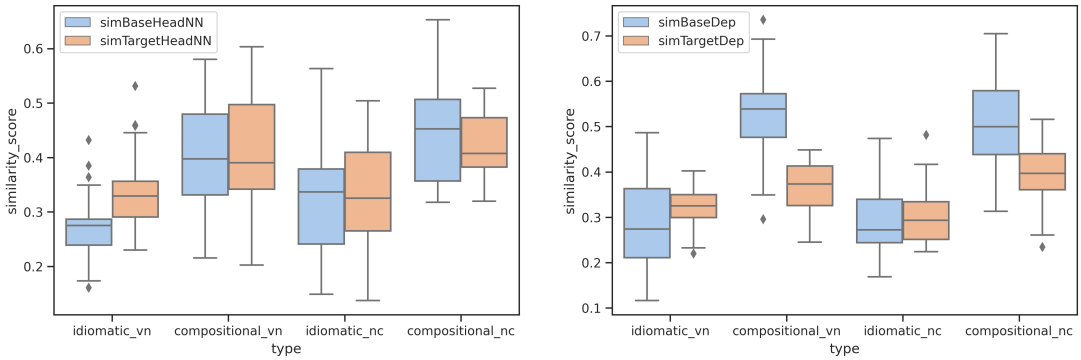
(d) MULTIAVG with DIFF input



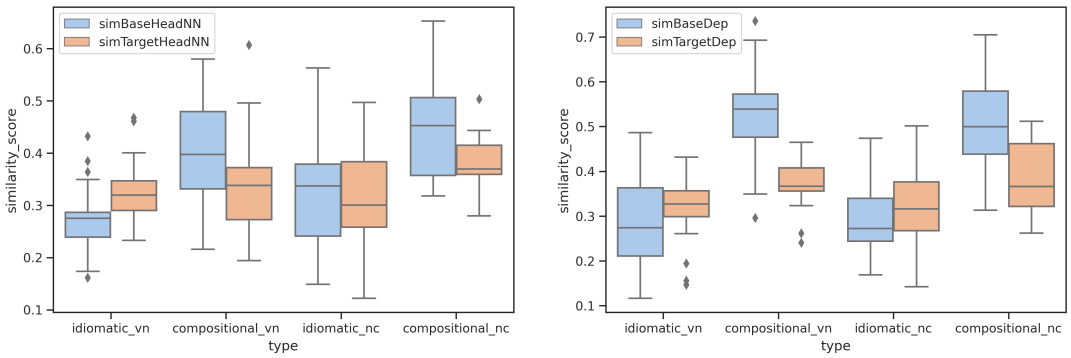
(e) MULTIMSE with CONCAT input



(f) MULTIMSE with DIFF input

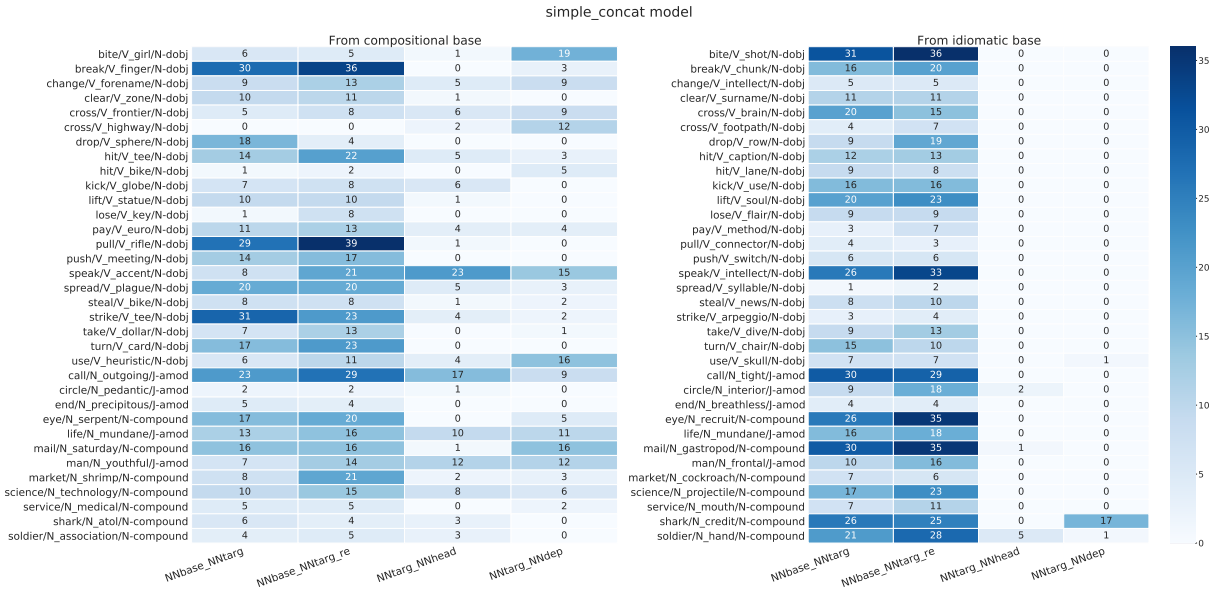


(g) HINGE with CONCAT input

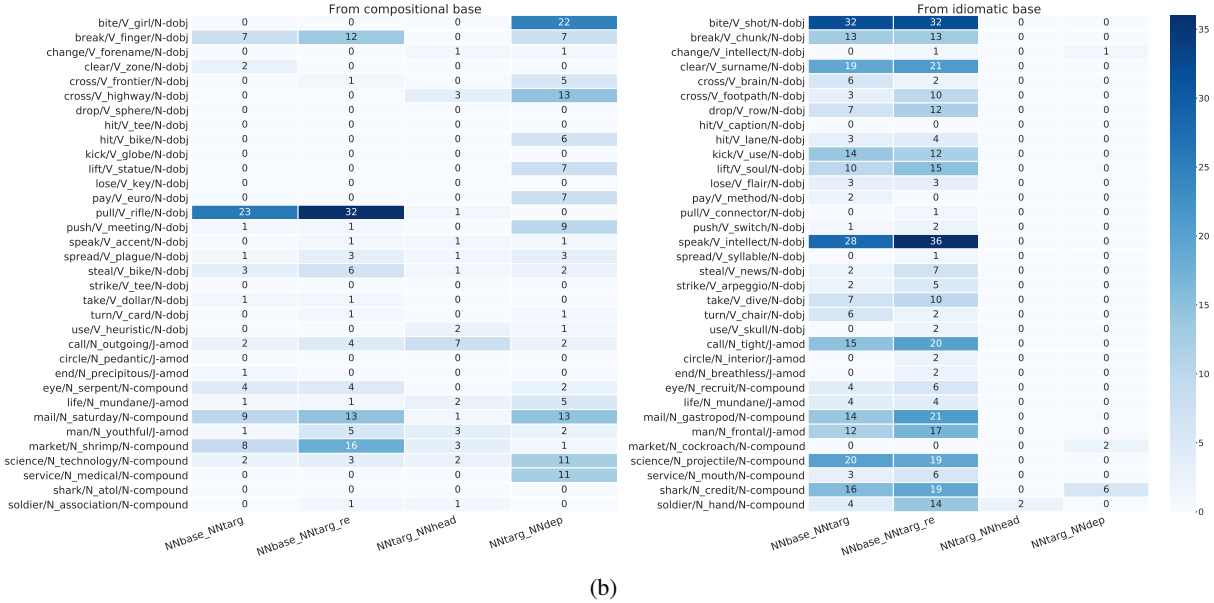


(h) HINGE with DIFF input

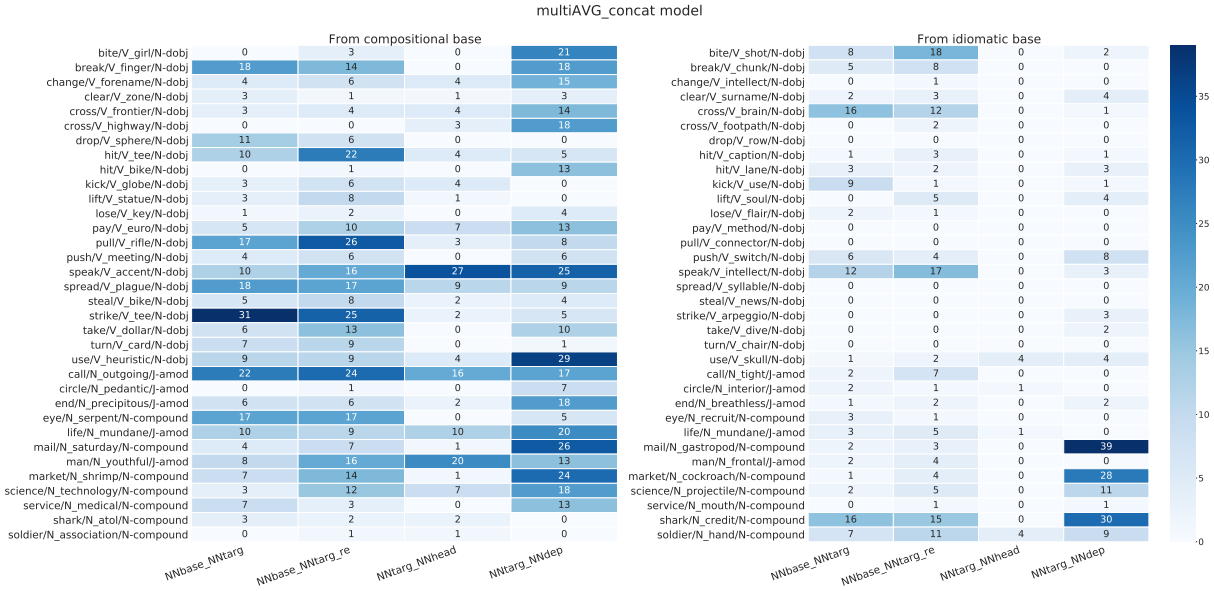
Figure 4: Distribution of the similarities between the RelEmb and its head (left), between the RelEmb and its argument (right). Data are grouped for syntactic type (nominal, NC, or verbal, VN) and if it is compositional (*compos*) or idiomatic (*idiom*). Similarities are computed for both the base embedding (blue) and target embedding (orange).


simple_concat model
From compositional base
bite/V_girl/N-dobj 6 5 1 19
break/V_finger/N-dobj 30 36 0 3
change/V_forename/N-dobj 9 13 5 9
clear/V_zone/N-dobj 10 11 1 0
cross/V_frontier/N-dobj 5 8 6 9
cross/V_highway/N-dobj 0 0 2 12
drop/V_sphere/N-dobj 18 4 0 0
hit/V_tee/N-dobj 14 22 5 3
hit/V_bike/N-dobj 1 2 0 5
kick/V_globe/N-dobj 7 8 6 0
lift/V_statue/N-dobj 10 10 1 0
lose/V_key/N-dobj 1 8 0 0
pay/V_euro/N-dobj 11 13 4 4
pull/V_rifle/N-dobj 29 39 1 0
push/V_meeting/N-dobj 14 17 0 0
speak/V_accent/N-dobj 8 21 23 15
spread/V_plague/N-dobj 20 20 5 3
steal/V_bike/N-dobj 8 8 1 2
strike/V_tee/N-dobj 31 23 4 2
take/V_dollar/N-dobj 7 13 0 1
turn/V_card/N-dobj 17 23 0 0
use/V_heuristic/N-dobj 6 11 4 16
call/N_outgoing/J-amod 23 29 17 9
circle/N_pedantic/J-amod 2 2 1 0
end/N_precipitous/J-amod 5 4 0 0
eye/N_serpent/N-compound 17 20 0 5
life/N_mundane/J-amod 13 16 10 11
mail/N_saturday/N-compound 16 16 1 16
man/N_youthful/J-amod 7 14 12 12
market/N_shrimp/N-compound 8 21 2 3
science/N_technology/N-compound 10 15 8 6
service/N_medical/N-compound 5 5 0 2
shark/N_atol/N-compound 6 4 3 0
soldier/N_association/N-compound 4 5 3 0
NNbase_NNtarg
NNbase_NNtarg_re
NNtarg_NNhead
NNtarg_NNdep
From idiomatic base
bite/V_shot/N-dobj 31 36 0 0
break/V_chunk/N-dobj 16 20 0 0
change/V_intellect/N-dobj 5 5 0 0
clear/V_surname/N-dobj 11 11 0 0
cross/V_brain/N-dobj 20 15 0 0
cross/V_footpath/N-dobj 4 7 0 0
drop/V_row/N-dobj 9 19 0 0
hit/V_caption/N-dobj 12 13 0 0
hit/V_lane/N-dobj 9 8 0 0
kick/V_use/N-dobj 16 16 0 0
lift/V_soul/N-dobj 20 23 0 0
lose/V_flair/N-dobj 9 9 0 0
pay/V_method/N-dobj 3 7 0 0
pull/V_connector/N-dobj 4 3 0 0
push/V_switch/N-dobj 6 6 0 0
speak/V_intellect/N-dobj 26 33 0 0
spread/V_syllable/N-dobj 1 2 0 0
steal/V_news/N-dobj 8 10 0 0
strike/V_arpeggio/N-dobj 3 4 0 0
take/V_dive/N-dobj 9 13 0 0
turn/V_chair/N-dobj 15 10 0 0
use/V_skull/N-dobj 7 7 0 1
call/N_tight/J-amod 30 29 0 0
circle/N_interior/J-amod 9 18 2 0
end/N_breathless/J-amod 4 4 0 0
eye/N_recruit/N-compound 26 35 0 0
life/N_mundane/J-amod 16 18 0 0
mail/N_gastropod/N-compound 30 35 1 0
man/N_frontal/J-amod 10 16 0 0
market/N_cockroach/N-compound 7 6 0 0
science/N_projectile/N-compound 17 23 0 0
service/N_mouth/N-compound 7 11 0 0
shark/N_credit/N-compound 26 25 0 17
soldier/N_hand/N-compound 21 28 5 1
NNbase_NNtarg
NNbase_NNtarg_re
NNtarg_NNhead
NNtarg_NNdep
35
30
25
20
15
10
5
0

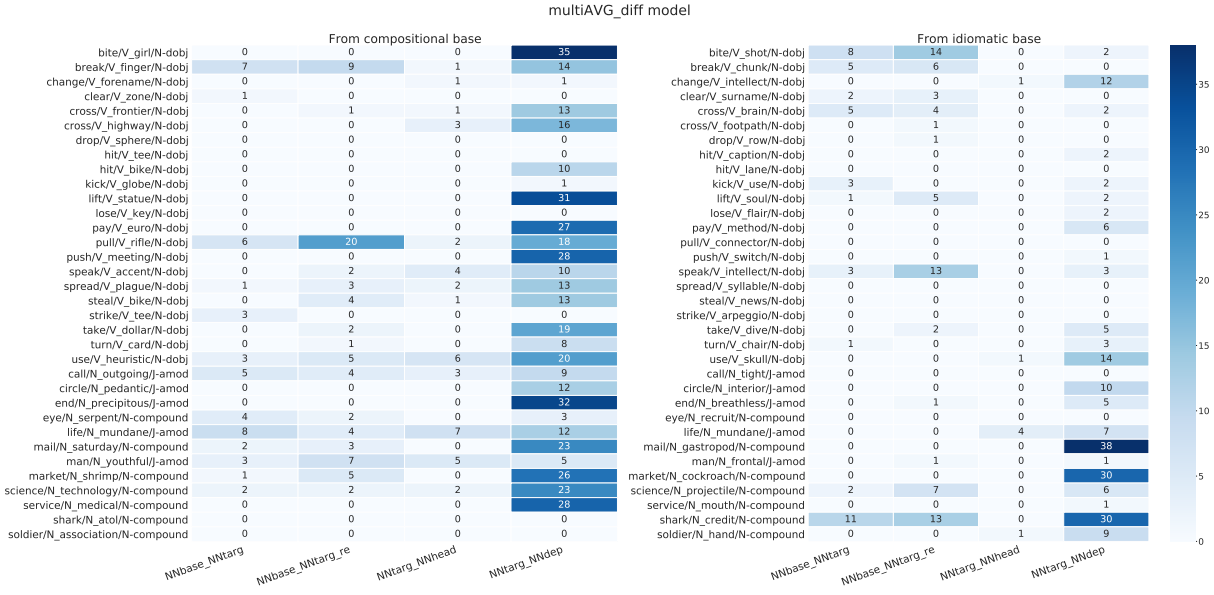
(a)


simple_diff model
From compositional base
bite/V_girl/N-dobj 0 0 0 22
break/V_finger/N-dobj 7 12 0 7
change/V_forename/N-dobj 0 0 1 1
clear/V_zone/N-dobj 2 0 0 0
cross/V_frontier/N-dobj 0 1 0 5
cross/V_highway/N-dobj 0 0 3 13
drop/V_sphere/N-dobj 0 0 0 0
hit/V_tee/N-dobj 0 0 0 0
hit/V_bike/N-dobj 0 0 0 6
kick/V_globe/N-dobj 0 0 0 0
lift/V_statue/N-dobj 0 0 0 7
lose/V_key/N-dobj 0 0 0 0
pay/V_euro/N-dobj 0 0 0 7
pull/V_rifle/N-dobj 23 32 1 0
push/V_meeting/N-dobj 1 1 0 9
speak/V_accent/N-dobj 0 1 1 1
spread/V_plague/N-dobj 1 3 1 3
steal/V_bike/N-dobj 3 6 1 2
strike/V_tee/N-dobj 0 0 0 0
take/V_dollar/N-dobj 1 1 0 0
turn/V_card/N-dobj 0 1 0 1
use/V_heuristic/N-dobj 0 0 2 1
call/N_outgoing/J-amod 2 4 7 2
circle/N_pedantic/J-amod 0 0 0 0
end/N_precipitous/J-amod 1 0 0 0
eye/N_serpent/N-compound 4 4 0 2
life/N_mundane/J-amod 1 1 2 5
mail/N_saturday/N-compound 9 13 1 13
man/N_youthful/J-amod 1 5 3 2
market/N_shrimp/N-compound 8 16 3 1
science/N_technology/N-compound 2 3 2 11
service/N_medical/N-compound 0 0 0 11
shark/N_atol/N-compound 0 0 0 0
soldier/N_association/N-compound 0 1 1 0
NNbase_NNtarg
NNbase_NNtarg_re
NNtarg_NNhead
NNtarg_NNdep
From idiomatic base
bite/V_shot/N-dobj 32 32 0 0
break/V_chunk/N-dobj 13 13 0 0
change/V_intellect/N-dobj 0 1 0 1
clear/V_surname/N-dobj 19 21 0 0
cross/V_brain/N-dobj 6 2 0 0
cross/V_footpath/N-dobj 3 10 0 0
drop/V_row/N-dobj 7 12 0 0
hit/V_caption/N-dobj 0 0 0 0
hit/V_lane/N-dobj 3 4 0 0
kick/V_use/N-dobj 14 12 0 0
lift/V_soul/N-dobj 10 15 0 0
lose/V_flair/N-dobj 3 3 0 0
pay/V_method/N-dobj 2 0 0 0
pull/V_connector/N-dobj 0 1 0 0
push/V_switch/N-dobj 1 2 0 0
speak/V_intellect/N-dobj 28 36 0 0
spread/V_syllable/N-dobj 0 1 0 0
steal/V_news/N-dobj 2 7 0 0
strike/V_arpeggio/N-dobj 2 5 0 0
take/V_dive/N-dobj 7 10 0 0
turn/V_chair/N-dobj 6 2 0 0
use/V_skull/N-dobj 0 2 0 0
call/N_tight/J-amod 15 20 0 0
circle/N_interior/J-amod 0 2 0 0
end/N_breathless/J-amod 0 2 0 0
eye/N_recruit/N-compound 4 6 0 0
life/N_mundane/J-amod 4 4 0 0
mail/N_gastropod/N-compound 14 21 0 0
man/N_frontal/J-amod 12 17 0 0
market/N_cockroach/N-compound 0 0 0 2
science/N_projectile/N-compound 20 19 0 0
service/N_mouth/N-compound 3 6 0 0
shark/N_credit/N-compound 16 19 0 6
soldier/N_hand/N-compound 4 14 2 0
NNbase_NNtarg
NNbase_NNtarg_re
NNtarg_NNhead
NNtarg_NNdep
35
30
25
20
15
10
5
0

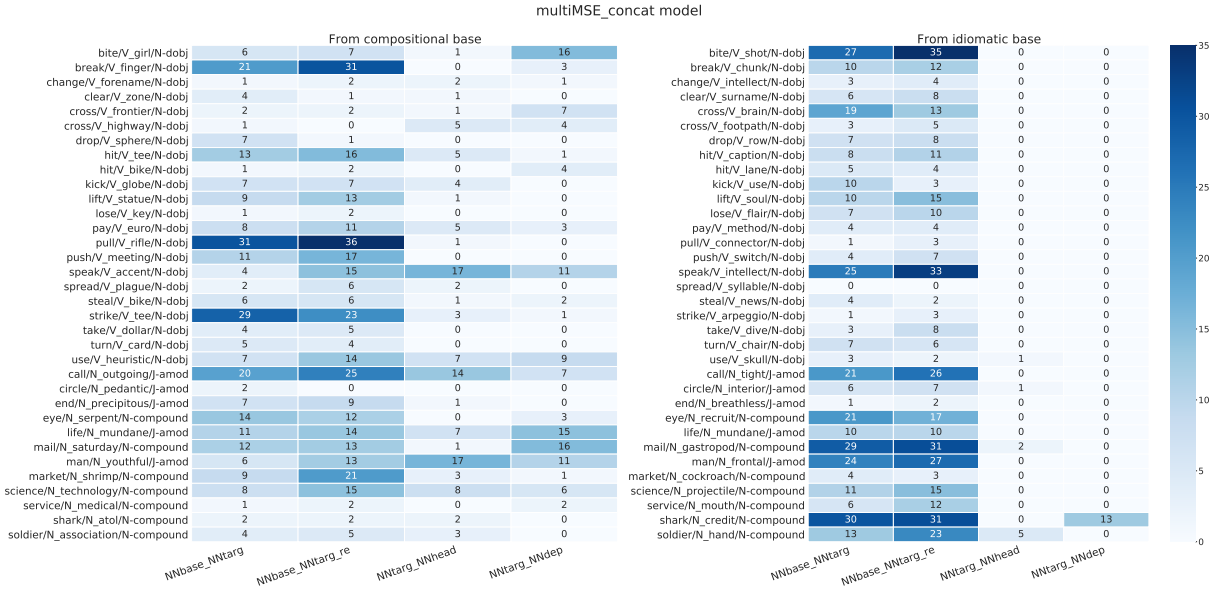
(b)


multiAVG_concat model
From compositional base
bite/V_girl/N-dobj 0 3 0 21
break/V_finger/N-dobj 18 14 0 18
change/V_forename/N-dobj 4 6 4 15
clear/V_zone/N-dobj 3 1 1 3
cross/V_frontier/N-dobj 3 4 4 14
cross/V_highway/N-dobj 0 0 3 18
drop/V_sphere/N-dobj 11 6 0 0
hit/V_tee/N-dobj 10 22 4 5
hit/V_bike/N-dobj 0 1 0 13
kick/V_globe/N-dobj 3 6 4 0
lift/V_statue/N-dobj 3 8 1 0
lose/V_key/N-dobj 1 2 0 4
pay/V_euro/N-dobj 5 10 7 13
pull/V_rifle/N-dobj 17 26 3 8
push/V_meeting/N-dobj 4 6 0 6
speak/V_accent/N-dobj 10 16 27 25
spread/V_plague/N-dobj 18 17 9 9
steal/V_bike/N-dobj 5 8 2 4
strike/V_tee/N-dobj 31 25 2 5
take/V_dollar/N-dobj 6 13 0 10
turn/V_card/N-dobj 7 9 0 1
use/V_heuristic/N-dobj 9 9 4 29
call/N_outgoing/J-amod 22 24 16 17
circle/N_pedantic/J-amod 0 1 0 7
end/N_precipitous/J-amod 6 6 2 18
eye/N_serpent/N-compound 17 17 0 5
life/N_mundane/J-amod 10 9 10 20
mail/N_saturday/N-compound 4 7 1 26
man/N_youthful/J-amod 8 16 20 13
market/N_shrimp/N-compound 7 14 1 24
science/N_technology/N-compound 3 12 7 18
service/N_medical/N-compound 7 3 0 13
shark/N_atol/N-compound 3 2 2 0
soldier/N_association/N-compound 0 1 1 0
NNbase_NNtarg
NNbase_NNtarg_re
NNtarg_NNhead
NNtarg_NNdep
From idiomatic base
bite/V_shot/N-dobj 8 18 0 2
break/V_chunk/N-dobj 5 8 0 0
change/V_intellect/N-dobj 0 1 0 0
clear/V_surname/N-dobj 2 3 0 4
cross/V_brain/N-dobj 16 12 0 1
cross/V_footpath/N-dobj 0 2 0 0
drop/V_row/N-dobj 0 0 0 0
hit/V_caption/N-dobj 1 3 0 1
hit/V_lane/N-dobj 3 2 0 3
kick/V_use/N-dobj 9 1 0 1
lift/V_soul/N-dobj 0 5 0 4
lose/V_flair/N-dobj 2 1 0 0
pay/V_method/N-dobj 0 0 0 0
pull/V_connector/N-dobj 0 0 0 0
push/V_switch/N-dobj 6 4 0 8
speak/V_intellect/N-dobj 12 17 0 3
spread/V_syllable/N-dobj 0 0 0 0
steal/V_news/N-dobj 0 0 0 0
strike/V_arpeggio/N-dobj 0 0 0 3
take/V_dive/N-dobj 0 0 0 2
turn/V_chair/N-dobj 0 0 0 0
use/V_skull/N-dobj 1 2 4 4
call/N_tight/J-amod 2 7 0 0
circle/N_interior/J-amod 2 1 1 0
end/N_breathless/J-amod 1 2 0 2
eye/N_recruit/N-compound 3 1 0 0
life/N_mundane/J-amod 3 5 1 0
mail/N_gastropod/N-compound 2 3 0 39
man/N_frontal/J-amod 2 4 0 0
market/N_cockroach/N-compound 1 4 0 28
science/N_projectile/N-compound 2 5 0 11
service/N_mouth/N-compound 0 1 0 1
shark/N_credit/N-compound 16 15 0 30
soldier/N_hand/N-compound 7 11 4 9
NNbase_NNtarg
NNbase_NNtarg_re
NNtarg_NNhead
NNtarg_NNdep
35
30
25
20
15
10
5
0

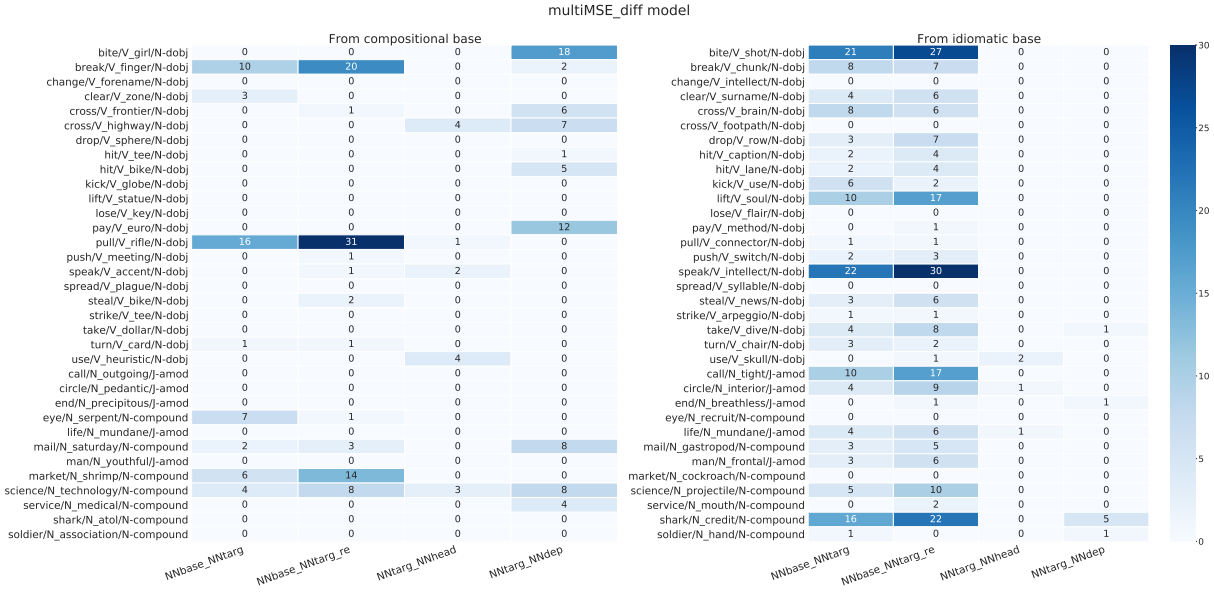
(c)


multiAVG_diff model
From compositional base
bite/V_girl/N-dobj 0 0 0 35
break/V_finger/N-dobj 7 9 1 14
change/V_forename/N-dobj 0 0 1 1
clear/V_zone/N-dobj 1 0 0 0
cross/V_frontier/N-dobj 0 1 1 13
cross/V_highway/N-dobj 0 0 3 16
drop/V_sphere/N-dobj 0 0 0 0
hit/V_tee/N-dobj 0 0 0 0
hit/V_bike/N-dobj 0 0 0 10
kick/V_globe/N-dobj 0 0 0 1
lift/V_statue/N-dobj 0 0 0 31
lose/V_key/N-dobj 0 0 0 0
pay/V_euro/N-dobj 0 0 0 27
pull/V_rifle/N-dobj 6 20 2 18
push/V_meeting/N-dobj 0 0 0 28
speak/V_accent/N-dobj 0 2 4 10
spread/V_plague/N-dobj 1 3 2 13
steal/V_bike/N-dobj 0 4 1 13
strike/V_tee/N-dobj 3 0 0 0
take/V_dollar/N-dobj 0 2 0 19
turn/V_card/N-dobj 0 1 0 8
use/V_heuristic/N-dobj 3 5 6 20
call/N_outgoing/J-amod 5 4 3 9
circle/N_pedantic/J-amod 0 0 0 12
end/N_precipitous/J-amod 0 0 0 32
eye/N_serpent/N-compound 4 2 0 3
life/N_mundane/J-amod 8 4 7 12
mail/N_saturday/N-compound 2 3 0 23
man/N_youthful/J-amod 3 7 5 5
market/N_shrimp/N-compound 1 5 0 26
science/N_technology/N-compound 2 2 2 23
service/N_medical/N-compound 0 0 0 28
shark/N_atol/N-compound 0 0 0 0
soldier/N_association/N-compound 0 0 0 0
NNbase_NNtarg
NNbase_NNtarg_re
NNtarg_NNhead
NNtarg_NNdep
From idiomatic base
bite/V_shot/N-dobj 8 14 0 2
break/V_chunk/N-dobj 5 6 0 0
change/V_intellect/N-dobj 0 0 1 12
clear/V_surname/N-dobj 2 3 0 0
cross/V_brain/N-dobj 5 4 0 2
cross/V_footpath/N-dobj 0 1 0 0
drop/V_row/N-dobj 0 1 0 0
hit/V_caption/N-dobj 0 0 0 2
hit/V_lane/N-dobj 0 0 0 0
kick/V_use/N-dobj 3 0 0 2
lift/V_soul/N-dobj 1 5 0 2
lose/V_flair/N-dobj 0 0 0 2
pay/V_method/N-dobj 0 0 0 6
pull/V_connector/N-dobj 0 0 0 0
push/V_switch/N-dobj 0 0 0 1
speak/V_intellect/N-dobj 3 13 0 3
spread/V_syllable/N-dobj 0 0 0 0
steal/V_news/N-dobj 0 0 0 0
strike/V_arpeggio/N-dobj 0 0 0 0
take/V_dive/N-dobj 0 2 0 5
turn/V_chair/N-dobj 1 0 0 3
use/V_skull/N-dobj 0 0 1 14
call/N_tight/J-amod 0 0 0 0
circle/N_interior/J-amod 0 0 0 10
end/N_breathless/J-amod 0 1 0 5
eye/N_recruit/N-compound 0 0 0 0
life/N_mundane/J-amod 0 0 4 7
mail/N_gastropod/N-compound 0 0 0 38
man/N_frontal/J-amod 0 1 0 1
market/N_cockroach/N-compound 0 0 0 30
science/N_projectile/N-compound 2 7 0 6
service/N_mouth/N-compound 0 0 0 1
shark/N_credit/N-compound 11 13 0 30
soldier/N_hand/N-compound 0 0 1 9
NNbase_NNtarg
NNbase_NNtarg_re
NNtarg_NNhead
NNtarg_NNdep
35
30
25
20
15
10
5
0

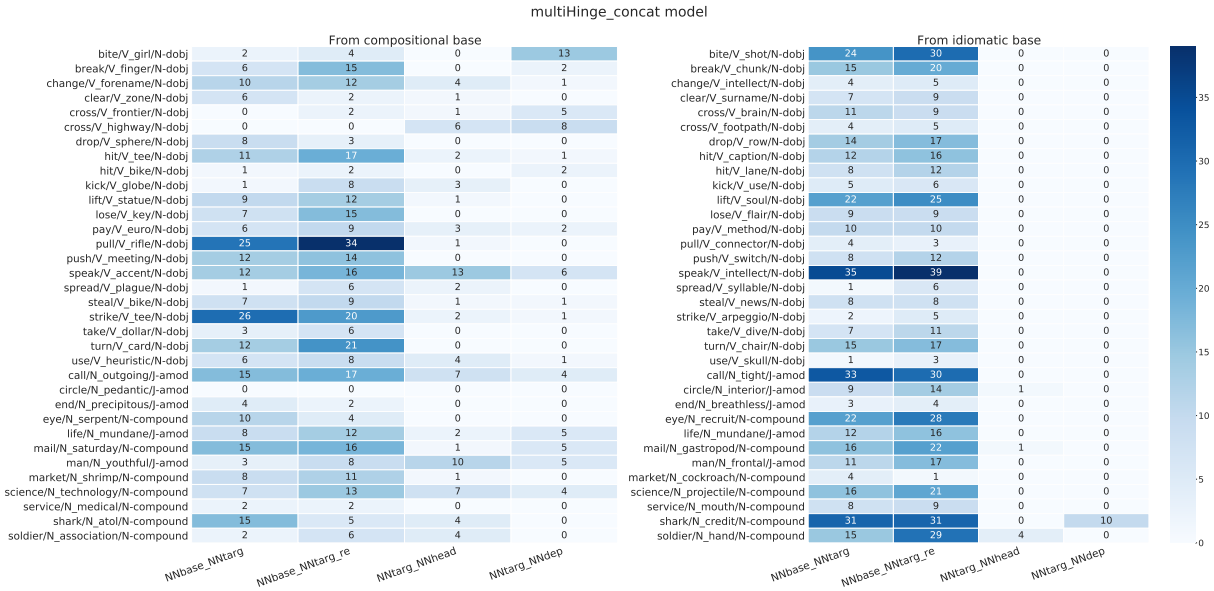
(d)


multiMSE_concat model
From compositional base
bite/V_girl/N-dobj 6 7 1 16
break/V_finger/N-dobj 21 31 0 3
change/V_forename/N-dobj 1 2 2 1
clear/V_zone/N-dobj 4 1 1 0
cross/V_frontier/N-dobj 2 2 1 7
cross/V_highway/N-dobj 1 0 5 4
drop/V_sphere/N-dobj 7 1 0 0
hit/V_tee/N-dobj 13 16 5 1
hit/V_bike/N-dobj 1 2 0 4
kick/V_globe/N-dobj 7 7 4 0
lift/V_statue/N-dobj 9 13 1 0
lose/V_key/N-dobj 1 2 0 0
pay/V_euro/N-dobj 8 11 5 3
pull/V_rifle/N-dobj 31 36 1 0
push/V_meeting/N-dobj 11 17 0 0
speak/V_accent/N-dobj 4 15 17 11
spread/V_plague/N-dobj 2 6 2 0
steal/V_bike/N-dobj 6 6 1 2
strike/V_tee/N-dobj 29 23 3 1
take/V_dollar/N-dobj 4 5 0 0
turn/V_card/N-dobj 5 4 0 0
use/V_heuristic/N-dobj 7 14 7 9
call/N_outgoing/J-amod 20 25 14 7
circle/N_pedantic/J-amod 2 0 0 0
end/N_precipitous/J-amod 7 9 1 0
eye/N_serpent/N-compound 14 12 0 3
life/N_mundane/J-amod 11 14 7 15
mail/N_saturday/N-compound 12 13 1 16
man/N_youthful/J-amod 6 13 17 11
market/N_shrimp/N-compound 9 21 3 1
science/N_technology/N-compound 8 15 8 6
service/N_medical/N-compound 1 2 0 2
shark/N_atol/N-compound 2 2 2 0
soldier/N_association/N-compound 4 5 3 0
NNbase_NNtarg
NNbase_NNtarg_re
NNtarg_NNhead
NNtarg_NNdep
From idiomatic base
bite/V_shot/N-dobj 27 35 0 0
break/V_chunk/N-dobj 10 12 0 0
change/V_intellect/N-dobj 3 4 0 0
clear/V_surname/N-dobj 6 8 0 0
cross/V_brain/N-dobj 19 13 0 0
cross/V_footpath/N-dobj 3 5 0 0
drop/V_row/N-dobj 7 8 0 0
hit/V_caption/N-dobj 8 11 0 0
hit/V_lane/N-dobj 5 4 0 0
kick/V_use/N-dobj 10 3 0 0
lift/V_soul/N-dobj 10 15 0 0
lose/V_flair/N-dobj 7 10 0 0
pay/V_method/N-dobj 4 4 0 0
pull/V_connector/N-dobj 1 3 0 0
push/V_switch/N-dobj 4 7 0 0
speak/V_intellect/N-dobj 25 33 0 0
spread/V_syllable/N-dobj 0 0 0 0
steal/V_news/N-dobj 4 2 0 0
strike/V_arpeggio/N-dobj 1 3 0 0
take/V_dive/N-dobj 3 8 0 0
turn/V_chair/N-dobj 7 6 0 0
use/V_skull/N-dobj 3 2 1 0
call/N_tight/J-amod 21 26 0 0
circle/N_interior/J-amod 6 7 1 0
end/N_breathless/J-amod 1 2 0 0
eye/N_recruit/N-compound 21 17 0 0
life/N_mundane/J-amod 10 10 0 0
mail/N_gastropod/N-compound 29 31 2 0
man/N_frontal/J-amod 24 27 0 0
market/N_cockroach/N-compound 4 3 0 0
science/N_projectile/N-compound 11 15 0 0
service/N_mouth/N-compound 6 12 0 0
shark/N_credit/N-compound 30 31 0 13
soldier/N_hand/N-compound 13 23 5 0
NNbase_NNtarg
NNbase_NNtarg_re
NNtarg_NNhead
NNtarg_NNdep
35
30
25
20
15
10
5
0

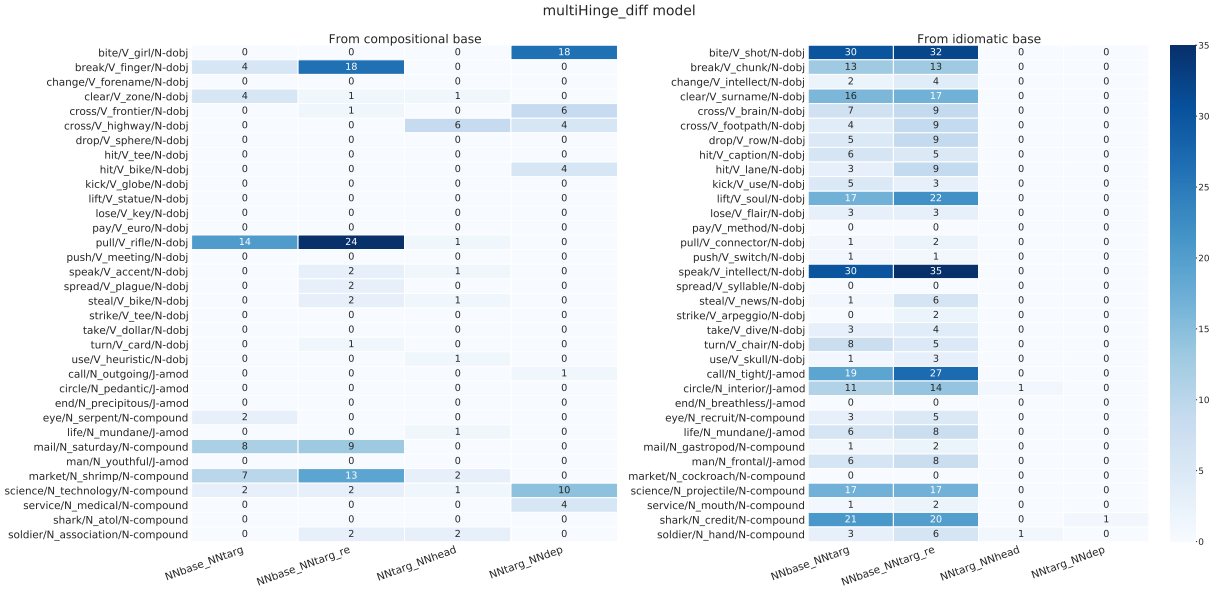
(e)


multiMSE_diff model
From compositional base
bite/V_girl/N-dobj 0 0 0 18
break/V_finger/N-dobj 10 20 0 2
change/V_forename/N-dobj 0 0 0 0
clear/V_zone/N-dobj 3 0 0 0
cross/V_frontier/N-dobj 0 1 0 6
cross/V_highway/N-dobj 0 0 4 7
drop/V_sphere/N-dobj 0 0 0 0
hit/V_tee/N-dobj 0 0 0 1
hit/V_bike/N-dobj 0 0 0 5
kick/V_globe/N-dobj 0 0 0 0
lift/V_statue/N-dobj 0 0 0 0
lose/V_key/N-dobj 0 0 0 0
pay/V_euro/N-dobj 0 0 0 12
pull/V_rifle/N-dobj 16 31 1 0
push/V_meeting/N-dobj 0 1 0 0
speak/V_accent/N-dobj 0 1 2 0
spread/V_plague/N-dobj 0 0 0 0
steal/V_bike/N-dobj 0 2 0 0
strike/V_tee/N-dobj 0 0 0 0
take/V_dollar/N-dobj 0 0 0 0
turn/V_card/N-dobj 1 1 0 0
use/V_heuristic/N-dobj 0 0 4 0
call/N_outgoing/J-amod 0 0 0 0
circle/N_pedantic/J-amod 0 0 0 0
end/N_precipitous/J-amod 0 0 0 0
eye/N_serpent/N-compound 7 1 0 0
life/N_mundane/J-amod 0 0 0 0
mail/N_saturday/N-compound 2 3 0 8
man/N_youthful/J-amod 0 0 0 0
market/N_shrimp/N-compound 6 14 0 0
science/N_technology/N-compound 4 8 3 8
service/N_medical/N-compound 0 0 0 4
shark/N_atol/N-compound 0 0 0 0
soldier/N_association/N-compound 0 0 0 0
NNbase_NNtarg
NNbase_NNtarg_re
NNtarg_NNhead
NNtarg_NNdep
From idiomatic base
bite/V_shot/N-dobj 21 27 0 0
break/V_chunk/N-dobj 8 7 0 0
change/V_intellect/N-dobj 0 0 0 0
clear/V_surname/N-dobj 4 6 0 0
cross/V_brain/N-dobj 8 6 0 0
cross/V_footpath/N-dobj 0 0 0 0
drop/V_row/N-dobj 3 7 0 0
hit/V_caption/N-dobj 2 4 0 0
hit/V_lane/N-dobj 2 4 0 0
kick/V_use/N-dobj 6 2 0 0
lift/V_soul/N-dobj 10 17 0 0
lose/V_flair/N-dobj 0 0 0 0
pay/V_method/N-dobj 0 1 0 0
pull/V_connector/N-dobj 1 1 0 0
push/V_switch/N-dobj 2 3 0 0
speak/V_intellect/N-dobj 22 30 0 0
spread/V_syllable/N-dobj 0 0 0 0
steal/V_news/N-dobj 3 6 0 0
strike/V_arpeggio/N-dobj 1 1 0 0
take/V_dive/N-dobj 4 8 0 1
turn/V_chair/N-dobj 3 2 0 0
use/V_skull/N-dobj 0 1 2 0
call/N_tight/J-amod 10 17 0 0
circle/N_interior/J-amod 4 9 1 0
end/N_breathless/J-amod 0 1 0 1
eye/N_recruit/N-compound 0 0 0 0
life/N_mundane/J-amod 4 6 1 0
mail/N_gastropod/N-compound 3 5 0 0
man/N_frontal/J-amod 3 6 0 0
market/N_cockroach/N-compound 0 0 0 0
science/N_projectile/N-compound 5 10 0 0
service/N_mouth/N-compound 0 2 0 0
shark/N_credit/N-compound 16 22 0 5
soldier/N_hand/N-compound 1 0 0 1
NNbase_NNtarg
NNbase_NNtarg_re
NNtarg_NNhead
NNtarg_NNdep
30
25
20
15
10
5
0

(f)



(g)



(h)

Figure 5: Heatmaps showing the intersection of common neighbors. Plot on the left refers to the target computed from a compositional base, plot on the right shows results for vectors generated from idiomatic base.