A Method for Studying Semantic Construal in Grammatical Constructions with Interpretable Contextual Embedding Spaces

Gabriella Chronis and **Kyle Mahowald** and **Katrin Erk**
The University of Texas at Austin
{gabriellachronis,kyle,katrin.erk}@utexas.edu

Abstract

We study semantic construal in grammatical constructions using large language models. First, we project contextual word embeddings into three interpretable semantic spaces, each defined by a different set of psycholinguistic feature norms. We validate these interpretable spaces and then use them to automatically derive semantic characterizations of lexical items in two grammatical constructions: nouns in subject or object position within the same sentence, and the AANN construction (e.g., 'a beautiful three days'). We show that a word in subject position is interpreted as more agentive than the very same word in object position, and that the nouns in the AANN construction are interpreted as more measurement-like than when in the canonical alternation. Our method can probe the distributional meaning of syntactic constructions at a templatic level, abstracted away from specific lexemes.

1 Introduction

There are now several paradigms for the linguistically oriented exploration of large neural language models. Major paradigms include treating the model as a linguistic test subject by measuring model output on test sentences (e.g., Linzen et al., 2016; Wilcox et al., 2018; Futrell et al., 2019) and building (often lightweight) probing classifiers on top of embeddings, to test whether the embeddings are sensitive to certain properties like dependency structure (Tenney et al., 2019; Hewitt and Manning, 2019; Rogers et al., 2020; Belinkov, 2022; Manning et al., 2020). [1]

Here, we consider another approach: projecting contextual, token-level embeddings into interpretable feature spaces defined by psycholinguistic feature norms (Binder et al., 2016; Buchanan et al., 2019; McRae et al., 2005). By learning a mapping to these spaces, as illustrated in Figure 1, we attain context-sensitive, interpretable, real-valued lexical-semantic features.

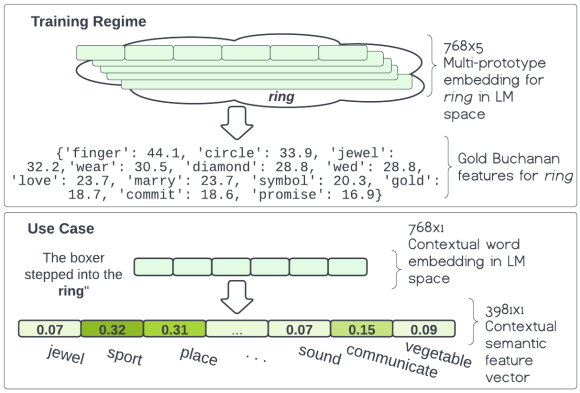

Figure 1: **(top)** Models are trained by using multi-prototype embeddings in LLM space to predict gold feature vectors derived from psycholinguistic feature norms. **(bottom)** These same models are used to project contextual word embeddings to interpretable contextual feature space (model=BUCHANAN-PLSR-MIL).

After experimenting to determine best practices for contextual-feature projection, we use these features to explore whether contextual embeddings are sensitive to subtle semantic *construals* in different grammatical constructions. Specifically, we observe how even seemingly similar constructions can impart a different semantics on their component parts or 'slot fillers' (Trott et al., 2020; Goldberg, 2019). Consider the Article + Adjective + Numeral + Noun (AANN) construction: e.g., "a beautiful three days in London," where the normally singular "a" precedes a plural noun and the adjective precedes the numeral (Solt, 2007; Dalrymple and King, 2019; Keenan, 2013). This construction often occurs with units or measure phrases (e.g., *days*, *feet*), but can also occur with non-measure nouns (e.g., "a lucky three students").

While it is tempting to think of "a lucky three students" as semantically equivalent to "three lucky

[1] Code and data for all experiments in this paper are available at https://github.com/gchronis/features_in_context.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 242–261
July 9-14, 2023

students," it has a different *construal*. Specifically, the AANN construction is acceptable only when the noun behaves as a single collective unit and is, in effect, more semantically similar to a unit of measurement than it would be in the unmarked construction. Evidence for a difference in meaning between the two variants is seen in their divergent distributions. For example, the AANN construction is unavailable in contexts like (1) and (2) (#-ed cases; adapted from Solt, 2007).

(1) The essay consisted of (a few eloquent paragraphs / # an eloquent few paragraphs) separated by pages of gibberish.

(2) He played (five boring songs / # a boring five songs), but in between he played one really good one.

The AANN construction cannot occur in contexts where the referent of the noun is split into non-contiguous parts. This distributional pattern is taken as evidence that the AANN construction construes its argument as a single, measure-like unit.

In this paper, we study distributional evidence on a larger scale, using a contextualized large language model as a 'compressed corpus' that captures observed statistical regularities over utterances of many speakers. We analyze this compressed corpus by mapping embeddings to interpretable feature spaces based on psycholinguistic feature norms. When we do this for the embedding of the noun *days* in "I spent a beautiful three days in London," we find the most salient difference with the "I spent three beautiful *days* in London" to be **a higher value for features like *measure* and *unit* when it is in an AANN construction.** We argue that this is because human speakers construe the AANN construction as being "measure-ish", and that this construal is reflected in their language use in a way that the contextual language model can pick up.

We conduct two case studies, one about AANNs and the other about grammatical subjecthood. Specifically, **we show that a word in subject position is interpreted as more agentive than the very same word in object position** (consistent with findings from psycholinguistics, e.g., Kako, 2006), and that **a noun in the AANN construction is interpreted as more measurement-like than when in the canonical alternation.** Our results demonstrate that construals can be inferred from statistical usage patterns. While we here use constructions with known construals, our positive results indicate that we may be able to analyze constructions where the construal is less clear in the theoretical literature.

While feature norms have been used to *interpret* distributional semantic models (Baroni and Lenci, 2010; Herbelot and Vecchi, 2015; Fagarasan et al., 2015; Rosenfeld and Erk, 2023), we emphasize the *linguistic* value of reliable, reusable, interpretable semantic spaces, which we use to interrogate the semantic properties of language in use. The ability of our method to characterize subtle semantic differences using language models offers a point of connection between linguistically oriented deep neural network analysis (Baroni, 2021) and topics in formal linguistics. In particular, this work empirically demonstrates the potential alignment between LMs and feature-based theories of lexical semantics (as illustrated by Petersen and Potts, 2023).

Our main goal is to use interpretable feature spaces for understanding the semantic construal of words in context, specifically the AANN construction and the transitive construction.

In Section 2, we lay out our method for constructing interpretable feature spaces for tokens in context. Then, in Section 3, we evaluate the success of our method on a sense differentiation task, a homonym feature prediction task, and a qualitative analysis. The idea is that, if the method for mapping from embedding space to context-sensitive feature space is successful, we will predict unique semantic features for different senses. Having established and validated our method, we then turn to our key constructions in Section 4.

2 Methods

The task is to learn a mapping from contextual word embedding space to an interpretable space defined by feature norms (Section 2.1), where every dimension corresponds to a semantic feature. We construct the training data by pairing feature norms with embeddings derived from contextual word vectors. We train models at the type-level, e.g., to map the embedding vectors for the word *ring* to the set of feature norms for *ring*, as shown in the top half of Figure 1. But ultimately, we use the model to predict semantic features for individual tokens. That is, we project the token vector of a single occurrence of the word "ring" into the feature space learned at the type-level, as shown in the bottom half of Figure 1.

2.1 Psycholinguistic feature norms

We construct three semantic spaces, trained from three datasets of psycholinguistic feature norms.

The McRae et al. (2005) feature norms comprise 541 concrete English nouns and 2,526 features. Participants were asked to list definitional properties of cue words. The features are full predicates; for example, a *brush* `'has_bristles'` and is `'used_on_hair'`.

The Buchanan et al. (2019) feature norms consist of over 4000 English words and 3,981 distinct features, from all open-class parts of speech, and include abstract words. The authors collect new norms and collate them with McRae norms and the Vinson and Vigliocco (2008) verb feature norms. The features are tokenized and lemmatized. If a participant said 'found in kitchens,' this yields the features `'found'` and `'kitchen'`.

The Binder et al. (2016) data consists of 535 English words rated for the relevance of 65 predefined features. The features were chosen to correspond to known neural activation regions in the human brain, and to domains of cognition and perception; they are more coarse grained than the other norms. The word *song* might have a high rating for 'Audition' but a lower rating for 'Vision'.

Feature norms as feature spaces Feature norms can be interpreted as vectors, with a real-valued dimension for each feature in the dataset. The differences between the feature norm data sets lead to differences in the feature inference problems. For MCRAE and BUCHANAN, values along each feature-dimension correspond to the number of participants who named that feature—zero in the majority of cases. These spaces are thus sparse and high-dimensional. For these two spaces, we treat the output as a ranked list of features, where the lower ranks are not relevant. The BINDER space is dense and low-dimensional, and the goal is to predict the value of each feature. Here, a low value on a feature does not indicate lack of relevance.

The norms differ in what they say about a word. The McRae and Buchanan norms are fine-grained, and represent salient or prototypical meanings. McRae norms are limited in their applicability because they only cover concrete nouns. Buchanan norms have a coverage that is wider but still somewhat ad-hoc. The Binder norms are high-level and were designed to be comprehensive.

Past and concurrent work on feature prediction has explored the utility of McRae (Fagarasan et al., 2015; Herbelot and Vecchi, 2015; Rosenfeld and Erk, 2023) and Binder (Utsumi, 2020; Turton et al., 2021) norms for probing distributional models and language models.

2.2 Embeddings

The feature norms serve as our gold feature labels that we map our type-level embeddings onto. For these type-level embeddings, we use embeddings derived from BERT (Devlin et al., 2019), either in a *vanilla* variety (one vector representation per word) or using *multi-prototype embeddings*, which have multiple embedding clusters per word (roughly corresponding to distinct usages). Specifically, we use the embeddings from Chronis and Erk (2020), which are generated by performing K-means clustering on BERT embeddings of tokens from the British National Corpus (BNC). This procedure collects up to 200 occurrences of each cue word in the British National Corpus, and generates token vectors for each occurrence with the HuggingFace `bert-base-uncased` model. For multi-prototype embeddings, these representations are clustered using K-means, using their best-performing setting of K=5 clusters per word at Layer 8. For vanilla embeddings, we generate BERT vectors through the same procedure, but simply average the token vectors together (K=1) to get one vector per word. See Appendix A for more detail on the multi-prototype vectors.

Though the mapping is *trained* from type-level (or sense-level) embeddings, contextual word vectors at the token level can be *projected* into the interpretable space using the resulting model.

2.3 Mapping from embeddings to feature norms

Though feature prediction is well explored for static embeddings (Baroni and Lenci, 2010; Herbelot and Vecchi, 2015; Fagarasan et al., 2015; Rosenfeld and Erk, 2023; Utsumi, 2020) and gaining popularity as a method to probe contextual embeddings (Chersoni et al., 2021; Turton et al., 2021; Apidianaki and Garí Soler, 2021; Proietti et al., 2022), there is no consensus as to which models work best for which datasets. We experiment with several mapping methods used previously for feature prediction. The first is a feed forward neural network (FFNN, with a single hidden layer, tanh activation, and dropout applied after the final output layer; Turton et al., 2020). The dropout parameter, hidden layer size, learning rate, and number of epochs

were grid-searched, as described in Appendix B (which also includes implementation details for the other models described). The second is partial least squares regression (PLSR, using the scikit-learn implementation; Herbelot and Vecchi, 2015; Fagarasan et al., 2015; Utsumi, 2020), whereby we run a partial least squares regression that predicts the feature space from the (potentially multi-prototype) embeddings. The third is label propagation (PROP; Rosenfeld and Erk, 2023), which percolates labels through a graph from labels to unlabeled nodes.

In all cases, the goal is to predict a real-valued semantic feature vector. Thus, the task is formulated as a multi-output regression problem. In the vanilla setting, the above methods can straightforwardly map from a particular word embedding into feature space. But, in order to map from a *multi-prototype* embedding into feature space, the problem is trickier—especially since the multi-prototype embeddings may capture meanings that are entirely absent in interpretable feature space.

Therefore, we test versions of each model using techniques inspired by multi-instance learning (MIL; Dietterich et al., 1997). The implementation of these MIL-inspired models is different for each of the three methods. For the FFNN, we use an attention mechanism that allows the model to learn a weighted average over instances, as in Ilse et al. (2018). For PLSR and Label Propagation, we simply construct a separate training example for each prototype drawn from the multi-prototype embedding That is, for a 5-prototype vector, we construct 5 training examples, where each of the 5 examples consists of a (unique) single prototype vector paired with the same type-level feature vector. See Appendix C for more detail on adaptations for the multi-prototype setting.

3 Evaluating Contextual Feature Norms for Interpreting Semantic Space

We first evaluated the models on their ability to fit the *type-level* feature norms they are trained on. We do not go into detail here, as it is context-dependent meanings we are most interested in. See Appendix D for full results. Overall, BERT-derived models were comparable to those we trained with static GloVe (Pennington et al., 2014) embeddings, and to the best static models in the literature. This initial evaluation established that models using BERT-derived embeddings are just as good as static embeddings for predicting semantic features.

	McRae		**Buchanan**		**Binder**	
	MIL	Vanilla	MIL	Vanilla	MIL	Vanilla
PLSR	.41	.39	.41	.42	.28	.26
FFNN	.36	.36	.42	.40	.30	.30
PROP	-.03	-.03	.10	.10	-.03	-.03

Table 1: Results of Sense Differentiation experiment. Pearson correlation of cosine similarities of predicted features vectors with Wu-Palmer similarity between senses. Data: pairs of tokens of the same noun lemma in SemCor. # Lemmas = 8021, # Token-pairs = 1,045,966, $p < 0.0001$ in all cases.

To evaluate our models on *in-context* feature prediction, we conduct two quantitative experiments: one on a sense differentiation task, one on a homonym disambiguation task, as well as a qualitative analysis for a representative word (*fire*). The goal of this section is to explore whether the contextual feature norm method successfully captures contextual modulation of word meaning. For these experiments, we select the hyperparameters for each model that performed the best at type-level feature prediction under 10-fold cross-validation (Appendix D).

3.1 Exp. 1: Sense Differentiation

Token-level evaluation is tricky because there are no existing datasets for in-context feature norms. Noting this obstacle, others utilize indirect methods like word-sense disambiguation and qualitative analysis, (Turton et al., 2020), or forego in-context evaluation (Chersoni et al., 2021).

Turton et al. (2020) evaluate the Binder feature prediction model using the Words in Context Dataset (Pilehvar and Camacho-Collados, 2019), which only labels token pairs as 'same meaning' or 'different meaning'. We devise a sense differentiation experiment using the SemCor corpus, (Miller et al., 1994), which lets us do a more fine-grained analysis in terms of close and distant polysemy.

The logic of this experiment is that, if two senses of a word are semantically *distant*, we expect the feature vectors in projected space to also be distant. We test the quality of our predicted feature vectors by testing how well the cosine distance between vectors for polysemous words corresponds to the distance between their senses in WordNet (Fellbaum, 2010).

To build this dataset, we collect examples of noun lemmas in the SemCor corpus, which is an-

notated with WordNet senses for words in context. In SemCor, "Water is a human right," is labeled `right.n.02`, *an abstract idea due to a person*, while "He walked with a heavy list to the right," is labeled `right.n.01`, *the side to the south when facing east*. To counteract data imbalance, we collect only up to 30 instances of a particular word from any one WordNet sense. We determine degrees of similarity between WordNet senses using Wu-Palmer similarity (Wu and Palmer, 1994), which measures the degrees of separation between them. Then, each token in the dataset is projected into interpretable semantic space. We compute the cosine similarity between pairs of tokens and compare them to the Wu-Palmer similarity of their word senses. The key hypothesis is that we should see highly similar predicted features for tokens of the same sense, somewhat divergent features when the senses are different but related, and very different features for distant senses.

Table 1 shows the results. Regardless of whether we use Multi-Instance Learning, both PLSR and FFNN models show a significant correlation between the sense similarity and similarity of predicted features. We interpret this to mean that PLSR and FFNN reflect *degree* differences of similarity between word senses.

Comparison to frozen BERT embeddings The results in Table 1 suggest that, at least to some extent, the projected semantic features capture information about different word senses. But to what extent? We take it as a given that the hidden layer embeddings of `bert-base`, because they are sensitive to context, reflect differences in word senses. Therefore, we run an additional baseline where we run the same correlational analysis using the frozen weights of `bert-base`, instead of the projected semantic feature. That is, we compute a correlation between the cosine distance between `bert-base` vectors from Layer 8 and the WordNet-derived Wu-Palmer similarity metric. The correlation between cosine distance and WordNet distance for plain BERT vectors is as high as our best models (Pearson's $r = 0.41$, $p < .0001$), which suggests that, even though the feature projection method is trained on word types, our training procedure does not lead to catastrophic information loss about word *tokens*. More precisely, for McRae and Buchanan datasets, PLSR learns a projection that is *as contextual* as the original BERT space. Our best Binder space (FFNN) is less contextual than the original BERT space, though it still differentiates senses. This evaluation also demonstrates that Label Propagation, which is good at fitting norms at the type level (as shown in Appendix D and Rosenfeld and Erk, 2023) is not an effective method for generating contextual features.

	McRae		**Buchanan**	
	MIL	Vanilla	MIL	Vanilla
PLSR	.50	.50	.42	.42
FFNN	.50	.50	.33	.25
PROP	.30	.30	.58	.25

Table 2: Results of Homonym Disambiguation Experiment. Performance on gold contextual feature prediction for homonyms (McRae and Buchanan only). Results reported are MAP@k. (n = 1093)

Performance varies across words Performance on this task is not necessarily uniform across all words. For instance, as discussed in Appendix E, performance on the sense differentiation task (using our interpretable feature projections *or* the original BERT embeddings) is better for concrete words, relative to abstract words. We leave it to future work to further explore this, as well as other sources of heterogeneity in performance.

3.2 Exp. 2: Homonym Disambiguation

The previous experiment considered many lemmas, with widely distinct as well as closely related senses. However, it is an indirect evaluation: it does not let us directly compare our projected context-dependent features to *known* context-dependent feature norms. But the MCRAE dataset offers a natural experiment, since it contains 20 homonymous words in disambiguated format. That is, separate norms exist in the MCRAE dataset (and per force the BUCHANAN dataset, which is a superset) for 'hose (water)' and 'hose (leggings)'. We treat these disambiguated norms as gold contextual features for tokens of these senses. That is, we treat the MCRAE features for 'hose (water)' as a gold label for the token "hose" in a sentence like "I watered my flowers with the hose." As SemCor only contains a few sense-annotated tokens for each of the relevant homonyms, we use CoCA (Davies, 2018), a large corpus that of largely American English news text, to collect a dataset of tokens for each homonym. See Appendix G for details. Models were re-trained on all words in the feature norm

dataset *except* the held-out homonyms.[2]

On this task, performance is measured as mean average precision (MAP@k) over the gold homonym features from McRae and Buchanan, where k is the number of gold features specific to each concept (Derby et al., 2019; Rosenfeld and Erk, 2023). Table 2 shows results. For both sets of norms, we see strong performance. The best-performing models achieve a precision of 0.50 (on McRae) and 0.42 (on Buchanan). Though we cannot directly compare performance, feature prediction is generally understood to be a very hard task, with SOTA performance for static McRae feature prediction at 0.36 (Rosenfeld and Erk, 2023). This is because models will often predict plausible features that aren't in the gold feature set, like `has_teeth` for *cat* (Fagarasan et al., 2015).

3.3 Qualitative Analysis

In order to get a better sense of our in-context predictions, we now explore predicted features for clusters of token embeddings, extracted using the clustering procedure described in Erk and Chronis (2023) (which use the same kind of multi-prototype embeddings as described in Section 2.2), for the representative word *fire*. Focusing on a single, highly polysemous word allows us to build fine-grained intuition as to the kind of information each of our feature norms can offer. In addition, characterizing token embedding clusters may be useful in itself: Giulianelli et al. (2020) use the term *usage types* (UTs) for clusters of token embeddings, and note that they reflect word senses and other regularities such as grammatical constructions. UTs have proven useful for the study of semantic change. However, while UTs are created automatically by clustering, researchers usually manually design labels for UTs to make their interpretation clear. An automatic labeling of token clusters with projected semantic features, as we demonstrate here, could hence be useful for studying UTs.

Our goal in this section is to take 5 UTs for the word *fire* from Erk and Chronis (2023) and project them into our interpretable semantic spaces (BINDER, MCRAE, and BUCHANAN). These UTs are: *destructive* fire (e.g., "There was a fire at Mr's store and they called it arson."), *cooking/cozy* fire (e.g., "They all went over to the fire for plates of meat and bread."), *artillery* fire (e.g., "a brief burst of machine-gun fire"), and *noun compounds* (e.g., "fire brigade," "fire hydrant"). These UTs are represented as the centroids of K-means clusters of token vectors for the word *fire*.

Then, we project these usage type vectors into interpretable semantic spaces, using PLSR+MIL for McRae and Buchanan, and FFNN+MIL for Binder. Predictably, the models predict similar features values in many cases, as the senses of *fire* have a lot in common. For example, in BUCHANAN space, all UTs except *artillery* have a high rating for 'hot' (Appendix F). To avoid this issue and get at how the usage types *differ*, for each UT we average over the features predicted for the other four embedding centroids and select the features with the greatest positive difference to the target UT. Table 3 shows the features that most distinguish each UT.

The most distinctive features in Binder space are reasonable—destructive fire is indeed unpleasant,

Buchanan	
1. figurative	animal, color, light, fire, burn
2. destructive	destroy, build, cause, break, person
3. artillery	act, weapon, kill, loud, human
4. cooking	hot, food, wood, burn, heat
5. N-N compounds	person, place, work, office, law
McRae	
1. figurative	has_legs, is_hard, different_sizes, has_4_legs, is_large
2. destructive	different_colors, a_mammal, made_of_paper, made_of_cement, inbeh_-_explodes
3. artillery	a_weapon, used_for_killing, made_of_metal, is_loud, used_for_war
4. cooking	found_in_kitchens, used_for_cooking, requires_gas, an_appliance, is_hot
5. N-N compounds	has_doors, used_for_transportation, a_bird, has_feathers, beh_-_eats
Binder	
1. figurative	Color, Needs, Harm, Cognition, Temperature
2. destructive	Unpleasant, Fearful, Sad, Consequential, Harm
3. artillery	UpperLimb, Communication, Social, Audition, Head
4. cooking	Pleasant, Needs, Happy, Near, Temperature
5. N-N compounds	Biomotion, Face, Speech, Body, Unpleasant

Table 3: The most distinctive features for each prototype of *fire* multi-prototype embeddings, in each of the three interpretable semantic spaces.

[2]Because Binder norms do not contain any homonymous pairs, this evaluation is unavailable for BINDER space.

fearful, full of consequences, sad, and capable of causing harm. The MCRAE features are reasonable for the more concrete senses, which have synonyms that appear in the dataset (like 'gun' for 3 and 'oven' for 4). However, in contrast to BINDER and BUCHANAN, the distinctive MCRAE features predicted for the more abstract UTs (1, 2, and 5) have no ready interpretation.

3.4 Discussion

Mapping method Looking at both experiments, PLSR obtained the overall best results for predicting both Buchanan and McRae features. For Binder features, where the model must predict the best fit along *every* dimension, FFNN does better. Based on these experiments, we recommend using PLSR to predict definitional features like McRae and Buchanan, and FFNN to predict comprehensive features like Binder.

MIL Aside from a few instances, the multi-instance framework does not drastically improve model performance. Though the positive effect is marginal, we use MIL in the case studies below.

Choice of feature norms The experiments above also give us insight into which feature space to use when. Experiment 1 shows that different senses are very distinct in McRae ($r = 0.41$) and Buchanan ($r = 0.41$) space, but not as distinct in Binder space ($r = 0.28$).

The qualitative look at feature predictions indicates that Buchanan and Binder models produce reasonable features for the word *fire* in different contexts, including when used in a more abstract sense. Though the best McRae model scores well overall on quantitative tasks, the qualitative analysis suggests that it does not extend well to abstract senses. This conclusion aligns with expectations, given that Buchanan and Binder norms contain features for verbs and abstract nouns, whereas the McRae norms only contains concrete nouns.

Binder feature vectors are comprehensive and good for examining abstract meanings, but Buchanan feature vectors can pinpoint more precise meanings. The case studies that follow use these feature spaces according to their strengths. To get an idea of the overarching differences between two constructions, we use BINDER (4.2). To generate specific descriptions of lexical meaning in context, we use BUCHANAN (4.1).

4 Evaluating Constructions in Context

Having validated that our method works for extracting meaningful, context-dependent semantic information from large language models, we turn to two target constructions: the AANN construction (described in the Introduction) and the basic transitive construction. Crucially, in both studies, the word types are largely controlled between conditions (e.g., comparing "The family spent a beautiful three days in London." vs. "The family spent three beautiful days in London."), and so we compare context-dependent features derived from minimally different sentences. This design lets us study the effect of context in a highly controlled way, without being influenced just by the identity of the words in the sentences.

4.1 Construction 1: 'A Beautiful Three Days'

Method Using a 1,000 sentence sample from Mahowald (2023)'s dataset of sentences templatically constructed with varying nouns, adjectives, numerals, and templates from a variety of subtypes, we compared AANN head nouns to their equivalent "default" forms (e.g., "The family spent a lovely three *days* in London." vs. "The family spent three lovely *days* in London"). Crucially, these form a near minimal pair.

We extracted the embeddings for the head noun token in each sentence. We projected the token embeddings into BUCHANAN space (using PLSR – MIL) and examined the delta between each feature, for each token, in the AANN construction vs. in the default construction.

Results The top 5 features associated with the AANN construction (relative to default) were: **measure**, **one**, green, **unit**, grow. The features most associated with default (relative to AANN) were: animal, leg, child, human, please. The bolded AANN features suggest that nouns in the AANN alternation are more measure-like, and treated as more singular. These are consistent with observations in the literature. Animacy-oriented words (e.g., animal, child, human) seem to be more associated with the default construction. Though this is not proposed outright in the literature, it's been observed that AANN's are more likely to be ungrammatical when the head noun is agentive (Solt, 2007).

Focusing in on a representative sentence pair that shows a particularly sharp difference, the word *meals* in "They consumed an ugly five meals." is rated much higher on the MEASURE (.18) and UNIT

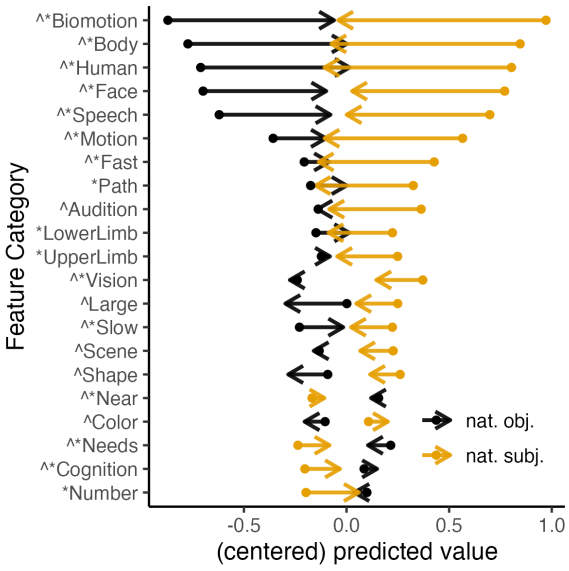

Figure 2: We plot the average predicted value of each feature for naturally occurring subjects and objects (points), and show how that probability shifts when we instead use swapped sentences (arrows). We show only those features which differ significantly for either overall subjectness vs. objectness (marked with a *), or for contextual swapping (caret). For example, Natural Objects have low values for the Biomotion feature; when swapped to subject position, their Biomotion value increases. Norms are centered but not normalized.

(.13) feature than the word *meals* in "They consumed five ugly meals." (.05 and .04, respectively). We interpret these results as evidence that projection into the Buchanan space detects a meaningful and attested semantic difference between the AANN construction and the default construction. Specifically, we can meaningfully detect that the construal associated with the AANN construction is more associated with measurement/units, compared to a non-AANN sentence matched on lexical content, even when the noun is not itself inherently a unit or measurement noun.

4.2 Construction 2: Grammatical Roles

Understanding grammatical roles like subject and object is crucial for natural language understanding. "The dog chased the cat." means something different from "The cat chased the dog." English relies largely on SVO word order for discriminating subjects vs. objects. Arguments that are animate, sentient, cause an event or a change of state in another participant, or move relative to another participant tend to be realized as subjects. Arguments that undergo a change of state, or are affected by another participant, tend to be realized as objects (Levin et al., 2005; Dowty, 1991). Most of the time, just knowing the two nouns in a transitive sentence is enough to know which is the subject and which is the object: If the nouns are "dog" and "bone", you can guess that "dog" is the subject and "bone" the object (Mahowald et al., 2022).

There is evidence that contextual language models like BERT represent subjecthood (Linzen et al., 2016; Papadimitriou et al., 2021; Hewitt and Manning, 2019). But do these models actually represent abstract grammatical subject, or do they rely on lexical information? One way to tease this apart is to study sentences where grammatical context and lexical heuristics come apart. Papadimitriou et al. (2022) showed that BERT can reliably distinguish between grammatical subject and object, even for sentences with non-prototypical arguments like, "The onion chopped the chef", but only in the higher levels of the model after more information has been shared. At lower layers, the model seems to rely on lexical information (e.g., would classify "chef" as the subject and "onion" as the object).

While prior work has explored the subject/object classification question by training bespoke probes, here we use projections into BINDER space. We focus on the set of English sentences studied in Papadimitriou et al. (2022), which are extracted from the Universal Dependencies Treebank (Nivre et al., 2016) and appear in two forms: the original form and a form in which the subject and object are swapped. For instance: compare the NATURAL, "Finally a chambermaid stuck her head around the corner" vs. the SWAPPED, "Finally a head stuck her chambermaid around the corner." The Treebank from which the sentences are sampled contains data from a number of different English corpora.

We project the subject and object in each of the 486 NATURAL sentences into BINDER space, using the FFNN-MIL method (which is best for token-level BINDER prediction), and then do the same for each of their SWAPPED counterparts. We first ask whether naturally occurring subjects tend to be more animate than objects. But we then ask whether, merely by virtue of being a subject, the lexical item takes on a more animate construal. Such a result would be consistent with psycholinguistic findings in humans: Kako (2006) shows that, even with nonce sentences like "The rom mecked the zarg," the subject word "rom" is rated as more animate.

Words that tend to appear in subject position are associated with higher animacy ratings. Given that there are known to be systematic differences between subjects and objects, will the Binder features for subjects and objects systematically differ in the NATURAL sentences? As can be seen in Figure 2, the answer is clearly yes. Animacy-associated features like Biomotion, Body, and Human are higher for naturally occurring subjects than for objects. We ran a linear regression predicting the Binder value from the subject/object status of the word, the Binder feature, and their interaction. The interaction term is the one we care about: how does the predicted value for that feature change when we are dealing with a subject or object? After Bonferroni correction for multiple comparisons, we find several features significantly correlated with subjecthood and a few with objecthood, starred in Figure 2.

The *same token* is construed as more animate when it appears in subject position. The preceding analysis could have been done using type-level Binder features: the upshot is that word *types* that appear in subject position get animacy-associated features. The highest rated words in this data set, for the Biomotion category, are: *animals*, *reptiles*, *cat*, *dog*, and they all occur as subjects in the corpus. But merely knowing that naturally occurring subjects and objects differ in Binder features does not tell us the whole story. Using the contextual feature projections, we can explore whether two tokens of the same type are construed as differing in animacy, based on whether they appear as a subject. We can do this in a controlled way by comparing the same word in the natural sentences and the swapped ones. For instance, in the sentence above, "chambermaid" appears as a subject but is an object in the swapped version. How does its Binder rating change? To assess that, we compare natural subjects vs. those same words moved to object position of the same verb in the same sentence. And we compare natural objects to those same words swapped to be subjects. Figure 2 shows that subject-oriented features like Biomotion, Body, and Human lose their large values and become more neutral. The careted features in the figure show significant effects of being swapped, after Bonferroni correction.

To assess whether our contextual feature predictions are sufficient for predicting whether a noun is a subject, no matter if natural or swapped, we run a forward-stepwise logistic regression on a portion of the data (300 sentences) to predict whether a particular token is a subject or an object based on its Binder ratings. The forward-stepwise part picks the set of Binder features that give the best prediction. We then test its k-fold cross-validation accuracy on the held-out test set. For NATURAL sentences, this method achieves 80% accuracy, compared to 73% accuracy for SWAPPED sentences. Thus, while natural sentences are easier, even the swapped sentences can be categorized better than chance using the feature norms—despite the fact that the words in question naturally occurred in the opposite roles.

We then performed the same procedure, but instead predicted whether a particular token was from a NATURAL or SWAPPED sentence. We did this separately for subjects and objects. Performance was above chance, at 70% and 71% respectively.

So a model can, with better than chance accuracy, use projected Binder features to identify which nouns are subjects in swapped sentences. But we can also predict which nouns are from swapped sentences. This result suggests that the predicted Binder features reflect contextual information, but also retain type-level information.

The results of our study align with Lebani and Lenci (2021) who investigate semantic proto-roles using distributional models and with Proietti et al. (2022), who investigate semantic proto-roles by projecting BERT into an interpretable space (similar to our method). Both show that transitive verbs have more proto-agent properties than their intransitive counterparts. The present analysis confirms and expands on their finding that BERT captures semantic role information and that projecting into interpretable space is a fruitful way of gaining insight into grammatical and thematic roles.

5 Conclusion

In this paper, we honed techniques for predicting semantic features for token embeddings. These projections are versatile. Once created, one and the same model can be used to study a wide array of phenomena. We explored their utility for studying semantic construal in syntactic constructions. We emphasize the potential of this method to answer linguistic questions about meaning differences in constructions that are less well-understood and well-theorized than the ones studied here. As such, we hope it will be possible to use this method to generate linguistic insight.

Limitations

One limitation of our study is that interpretable feature spaces are at times only semi-interpretable. We infer from patterns of model behavior that Buchanan features such as 'human', 'child', and 'animal' can be signals for animacy more broadly construed. The need to conjecture about what a feature means points to a weakness in our approach. Some interpretation will always be necessary, and with a more heavy-handed probing method like ours, it can't be certain what effects are coming from the model and which are coming from the probe.

One way to get around this need for subjective interpretation is to train a separate classifier for animacy more broadly understood, and then use the feature prediction model to examine what features are most relevant to the classifier (Chersoni et al., 2021). However, this method is not foolproof either. The classification distinction is wholly determined by the labeled data used to train the animacy probe, and the judgments are subjective. Even for a seemingly straightforward feature, the correct label is not always clear. Is a clock that *sings* the hour animate? What about a *stony face*?

Subjective interpretation is an important and unavoidable component of both linguistic and neural language model analysis. The goal of data-driven research is to extend the sphere of concern beyond self-reflexive subjective judgments of the researcher to the shared subjectivities of a language community. Information about animacy reflected in an annotated dataset still reflects subjectivities, but shared ones. It is important to always be clear about where interpretation is happening, whose interpretations are taken into account, and how they affect what conclusions may be drawn.

On that note, there are a few places where design decisions affect our analysis of lexical variation. Linguistic data enters the modeling pipeline in at least four places: BooksCorpus and Wikipedia data used to pre-train BERT, the BNC corpus which we use to derive multi-prototype embeddings, the feature norm datasets which tend to capture the subjectivities of American college students, and the texts we analyze in our case studies (both natural language text and constructed examples). These resources all cover English, but necessarily reflect different varieties of English, given that they were collected in different places at different times. For example, usage types in the BNC often differ from those derived from Wikipedia data.

Not only do the corpora we use represent potentially disjoint varieties (English spoken by college students in Vermont, English in newswire and fiction genres, English in reference texts). They also all represent the semantics of the unmarked, *normative varieties* of English. Normative English dominates all data collection contexts upon which our study rests. Consequently, to the extent that our model is a proxy for English semantic judgments, it is a proxy for dominant semantic associations among the composers of these texts and participants in the feature norm studies.

Though it is interesting and useful to study the English language as a whole, care must be taken to ensure that the sample is representative of all speakers; and ideally, our approach supports linguistic approaches which aim to describe and explain the semantics of smaller language communities. This would require language models trained on corpora at the level of communities of practice, as well as feature norms specific to these communities. We are hopeful that the future of statistical methods in lexical semantic analysis moves in this direction.

Ethics Statement

Our models are developed and published in order to encourage academic research in descriptive linguistics. In the future, we plan to use our method to study the inherent non-neutrality of language models by examining the influence of training corpus composition on the semantic representation of social meanings, as represented by cultural keywords. Because they are built on top of an unpredictable language model, the feature prediction methods, as well as the models we publish, are recommended for descriptive research only. Researchers should take into account the potential for language models, like language, to reflect of harmful ideologies such as sexism, racism, homophobia, and other forms of bigotry.

Acknowledgements

This work was made possible through funding from an NSF GRFP Grant to GC, NSF Grant 2139005 to KM. Thank you to the UT Austin Linguistics Computational Linguistics group for helpful comments and the SynSem group for their enthusiasm in considering how language modeling might inform their questions in semantics. For helpful discussions, thanks to Adele Goldberg and the Prince-

ton language group, Richard Futrell, and Isabel Papadimitriou.

References

Marianna Apidianaki and Aina Garí Soler. 2021. ALL dolphins are intelligent and SOME are friendly: Probing BERT for nouns' semantic properties and their prototypicality. In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 79–94, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Marco Baroni. 2021. On the proper role of linguistically-oriented deep net analysis in linguistic theorizing. *arXiv preprint arXiv:2106.08694*.

Marco Baroni and Alessandro Lenci. 2010. Distributional Memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Jeffrey R. Binder, Lisa L. Conant, Colin J. Humphries, Leonardo Fernandino, Stephen B. Simons, Mario Aguilar, and Rutvik H. Desai. 2016. Toward a brain-based componential semantic representation. *Cognitive Neuropsychology*, 33(3-4):130–174.

Marc Brysbaert, AB Warriner, and V Kuperman. 2014. Concreteness ratings for 40 thousand generally known English word lemmas. *BEHAVIOR RESEARCH METHODS*, 46(3):904–911.

Erin M. Buchanan, K. D. Valentine, and Nicholas P. Maxwell. 2019. English semantic feature production norms: An extended database of 4436 concepts. *Behavior Research Methods*, 51(4):1849–1863.

Emmanuele Chersoni, Enrico Santus, Chu-Ren Huang, and Alessandro Lenci. 2021. Decoding word embeddings with brain-based semantic features. *Computational Linguistics*, 47(3):663–698.

Gabriella Chronis and Katrin Erk. 2020. When is a bishop not like a rook? When it's like a rabbi! Multi-prototype BERT embeddings for estimating semantic relationships. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 227–244, Online. Association for Computational Linguistics.

Mary Dalrymple and Tracy Holloway King. 2019. An amazing four doctoral dissertations. *Argumentum*, 15(2019). Publisher: Debreceni Egyetemi Kiado.

Mark Davies. 2018. The 14 Billion Word iWeb Corpus. https://www.english-corpora.org/iWeb/.

Steven Derby, Paul Miller, and Barry Devereux. 2019. Feature2Vec: Distributional semantic modelling of human property knowledge. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5853–5859, Hong Kong, China. Association for Computational Linguistics.

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.

Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. 1997. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1-2):31–71.

David Dowty. 1991. Thematic proto-roles and argument selection. *Language*, 67(3):547–619.

Katrin Erk and Gabriella Chronis. 2023. Katrin Erk and Gabriella Chronis. Word embeddings are word story embeddings (and that's fine). In Shalom Lappin and Bernardy Jean-Philippe, editors, *Algebraic Structures in Natural Language*. Taylor and Francis, Oxford.

Luana Fagarasan, Eva Maria Vecchi, and Stephen Clark. 2015. From distributional semantics to feature norms: Grounding semantic models in human perceptual data. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 52–57, London, UK. Association for Computational Linguistics.

C. Fellbaum. 2010. WordNet. *Theory and Applications of Ontology: Computer Applications*, pages 231–243.

Richard Futrell, Ethan Wilcox, Takashi Morita, Peng Qian, Miguel Ballesteros, and Roger Levy. 2019. Neural language models as psycholinguistic subjects: Representations of syntactic state. 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 32–42, Minneapolis, Minnesota. Association for Computational Linguistics.

Mario Giulianelli, Marco Del Tredici, and Raquel Fernández. 2020. Analysing Lexical Semantic Change with Contextualised Word Representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3960–3973, Online. Association for Computational Linguistics.

Adele E Goldberg. 2019. *Explain me this: Creativity, competition, and the partial productivity of constructions*. Princeton University Press.

Aurélie Herbelot and Eva Maria Vecchi. 2015. Building a shared world: Mapping distributional to model-theoretic semantic spaces. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 22–32, Lisbon, Portugal. Association for Computational Linguistics.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. 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 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.

M Ilse, JM Tomczak, M Welling, et al. 2018. Attention-based deep multiple instance learning. *Proceedings of Machine Learning Research*, 80.

Brendan T. Johns and Michael N. Jones. 2012. Perceptual Inference Through Global Lexical Similarity. *Topics in Cognitive Science*, 4(1):103–120.

Edward Kako. 2006. Thematic role properties of subjects and objects. *Cognition*, 101(1):1–42.

Caitlin Keenan. 2013. A pleasant three days in Philadelphia: Arguments for a pseudopartitive analysis. *University of Pennsylvania Working Papers in Linguistics*, 19(1):11.

Gianluca E. Lebani and Alessandro Lenci. 2021. Investigating Dowty's proto-roles with embeddings. *Lingue e linguaggio*, 2:165–197.

Beth Levin, Malka Rappaport Hovav, et al. 2005. *Argument realization*. Cambridge University Press Cambridge.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

Kyle Mahowald. 2023. A discerning several thousand judgments: GPT-3 rates the article + adjective + numeral + noun construction. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 265–273, Dubrovnik, Croatia. Association for Computational Linguistics.

Kyle Mahowald, Evgeniia Diachek, Edward Gibson, Evelina Fedorenko, and Richard Futrell. 2022. Grammatical cues are largely, but not completely, redundant with word meanings in natural language. *arXiv preprint arXiv:2201.12911*.

Christopher D Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, and Omer Levy. 2020. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences*, 117(48):30046–30054.

Ken McRae, George S. Cree, Mark S. Seidenberg, and Chris Mcnorgan. 2005. Semantic feature production norms for a large set of living and nonliving things. *Behavior Research Methods*, 37(4):547–559.

Timothee Mickus, Denis Paperno, Mathieu Constant, and Kees van Deemter. 2020. What do you mean, BERT? In *Proceedings of the Society for Computation in Linguistics 2020*, pages 279–290, New York, New York. Association for Computational Linguistics.

George A. Miller, Martin Chodorow, Shari Landes, Claudia Leacock, and Robert G. Thomas. 1994. Using a Semantic Concordance for Sense Identification. In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 8-11, 1994*.

Joakim Nivre, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666.

Isabel Papadimitriou, Ethan A. Chi, Richard Futrell, and Kyle Mahowald. 2021. Deep subjecthood: Higher-order grammatical features in multilingual BERT. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2522–2532, Online. Association for Computational Linguistics.

Isabel Papadimitriou, Richard Futrell, and Kyle Mahowald. 2022. When classifying grammatical role, BERT doesn't care about word order... except when it matters. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 636–643, Dublin, Ireland. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Erika Petersen and Christopher Potts. 2023. Lexical semantics with large language models: A case study of English "break". In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 490–511, Dubrovnik, Croatia. Association for Computational Linguistics.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: The Word-in-Context Dataset for Evaluating Context-Sensitive Meaning Representations. 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 1267–1273, Minneapolis, Minnesota. Association for Computational Linguistics.

Mattia Proietti, Gianluca Lebani, and Alessandro Lenci. 2022. Does BERT recognize an agent? modeling Dowty's proto-roles with contextual embeddings. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4101–4112, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Alex Rosenfeld and Katrin Erk. 2023. An analysis of property inference methods. *Natural Language Engineering*, 29(2):201–227.

Stephanie Solt. 2007. Two types of modified cardinals. In *International Conference on Adjectives. Lille*.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Sean Trott, Tiago Timponi Torrent, Nancy Chang, and Nathan Schneider. 2020. (Re)construing meaning in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5170–5184, Online. Association for Computational Linguistics.

Jacob Turton, Robert Elliott Smith, and David Vinson. 2021. Deriving contextualised semantic features from BERT (and other transformer model) embeddings. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 248–262, Online. Association for Computational Linguistics.

Jacob Turton, David Vinson, and Robert Smith. 2020. Extrapolating Binder Style Word Embeddings to New Words. In *Proceedings of the Second Workshop on Linguistic and Neurocognitive Resources*, pages 1–8, Marseille, France. European Language Resources Association.

Akira Utsumi. 2020. Exploring What Is Encoded in Distributional Word Vectors: A Neurobiologically Motivated Analysis. *Cognitive Science*, 44(6):e12844.

David P. Vinson and Gabriella Vigliocco. 2008. Semantic feature production norms for a large set of objects and events. *Behavior Research Methods*, 40(1):183–190.

Ethan Wilcox, Roger Levy, Takashi Morita, and Richard Futrell. 2018. What do RNN language models learn about filler–gap dependencies? In *Proceedings of BlackboxNLP*, Brussels.

Zhibiao Wu and Martha Palmer. 1994. Verb Semantics and Lexical Selection. In *32nd Annual Meeting of the Association for Computational Linguistics*, pages 133–138, Las Cruces, New Mexico, USA. Association for Computational Linguistics.

A Embedding Details

For training, we use the multi-prototype embeddings of Chronis and Erk (2020). They are generated by performing k-means clustering on BERT embeddings of tokens from the British National Corpus (BNC). This procedure collects up to 200 occurrences of each cue word in the BNC and generates tokens vectors for each occurrence with the HuggingFace `bert-base-uncased` model. These representations are then clustered using K-means, using the authors' best-performing setting of K=5 clusters per word at layer 8. These multi-prototype vectors are unordered, 'bag-of-senses' representations.

For the static embedding baseline, we use the pretrained Wikipedia 2014 + Gigaword 5 pretrained GloVe with 300 dimensions, which is trained on 6B tokens with 400K vocabulary word (Pennington et al., 2014).

For token-level evaluations in Section 3 above, it does not make sense to compare to GloVe because GloVe embedding space is not contextual. Instead, we compare the multi-prototype, MIL models to single prototype (vanilla) versions of each model. Embeddings for the vanilla models are generated using the same procedure described above for multi-prototype, but all tokens are averaged into a single vector representation (K=1) rather than clustering them into prototypes.

B Model Implementation Details

For all models, we train using ten-fold cross-validation with an 80-10-10 train-dev-test split. For the MIL models, no prototypes of the same word are repeated between train and test sets. For each prediction task, we tune model hyperparameters using a sampled grid search (see uploaded code and data for details). The chosen hyperparameter settings are the ones with the best average performance on the dev set across folds.

The FFNN model is implemented in PyTorch and trained using the Adam optimizer with stochastic gradient descent. We search over number of epochs (30, 50); dropout (0, .2, .5), learning rates (1e-5, 1e-4, 1e-3), and hidden layer size (50, 100, 300).

Partial Least Squares regression is a statistical method to find the fundamental relations between

two matrices (semantic spaces). PLSR is useful in this case because it allows for correlations among the independent variables (embedding dimensions). We use the PLSR implementation from scikit-learn. We grid search over the number of PLSR dimensionality components (50, 100, 300).

Label propagation uses code from Rosenfeld and Erk (2023). Models were trained on a 2.3 GHz 8-Core Intel Core i9 processor with 16 GB of RAM. In label propagation, each labeled training example is embedded as a node in a graph along with unlabeled training data. Training takes place iteratively; in each iteration, labels spread through the graph. In this method, word embeddings are labeled with their corresponding features, withholding labels from the test set. Unlabeled nodes receive features of labeled nodes which are nearby in embedding space. Johns and Jones (2012) first applied this method to feature prediction from distributional models. In their model, the features of an unlabeled word are calculated as a weighted sum of the feature values that labeled words have—the weights are determined by cosine distance in distributional semantic space. Rosenfeld and Erk (2023) evaluate more sophisticated approaches to label propagation, called modified absorption. With modified absorption, labels do not propagate under certain conditions. For instance, features won't propagate to words that are very unfamiliar, or to words which are already well-labeled with properties.

C Predicting with Multi-Prototype Embeddings

The classic MIL problem is a classification task. The input is an unordered bag of instances, and the output is a binary classification label. The label of the whole bag is 1 if at least one of the instances in the bag has the label 1. However, the labels of the individual instances are unknown—only the bag labels are available. We take this as inspiration for our scenario, where we have a multi-prototype representation, along with a feature vector that may reflect only one of the prototypes (as in the *ring* example above).

To make the FFNN suitable for MIL, the FFNN is extended by an attention mechanism without ordering, as in Ilse et al. (2018). This method computes a weighted average over the instances. Code for the attention module was adapted from their implementation, and can be found at `https://github.com/AMLab-Amsterdam/AttentionDeepMIL`. It was used in combination with the attention module defined in this blog post: `https://medium.com/swlh/multiple-instance-learning-c49bd21f5620`.

To adapt PLSR for MIL, we construct one training example for each prototype. That is, for a 5-prototype vector, we construct 5 training examples, one for each vector, labeled with the type-level features. Thus, we conduct PLSR on a dataset with noisy labels. No prototypes of the same word are repeated between train and test sets.

Similar to PLSR, to adapt Label Propagation for multi-prototype embedding inputs, we represent each prototype as an independent node that maps to a type-level feature vector.

D Type-level Evaluation Results

Results are reported on the type-level training task. These evaluations show how well the different models are able to fit the different feature norms. We find that all models are on par with the performance reported in the existing literature on inferring static semantic features (Fagarasan et al., 2015; Herbelot and Vecchi, 2015; Derby et al., 2019).

Our goal is to predict semantic feature norms from words in context. We define a mapping problem from contextual-language-model-derived embeddings to an interpretable semantic space defined by psycholinguistic feature norms. The training data are experimentally collected semantic features for word *types*. Each consists of a cue word and a feature vector. We compare MIL and vanilla versions of FFNN, PLSR, and Label Propagation models.

The literature on feature prediction uses different evaluation methods. For McRae and Buchanan prediction, where the goal is to produce the most important features, we report Mean Average Precision at K (MAP@K), where K is the number of gold features for a concept (Derby et al., 2019). For Binder vectors, every feature is valued for every word, MAP@k is always equal to 1. For Binder, where the goal is to capture the relative importance of each feature, precision is not an appropriate metric. In this case, we use mean squared error (MSE) to measure the best overall fit.

Performance overall matched the best results in the literature for static feature prediction, and models that used the BERT embeddings performed as well or better compared to training on static GloVe embeddings (Table 4). On the McRae prediction

Model	McRae MAP@k (↑)	Buchanan MAP@k (↑)	Binder MSE (↓)
PLSR			
BERT MIL	0.33	**0.37**	2.32
BERT Vanilla	**0.34**	0.29	2.37
GloVe	0.33	0.23	2.37
FFNN			
BERT MIL	0.32	0.26	0.82
BERT Vanilla	0.32	0.26	0.88
GLoVe	0.30	0.26	1.14
PROP			
BERT MIL	0.31	0.32	0.96
BERT Vanilla	0.32	0.30	**0.10**
GloVe	0.30	0.26	0.89

Table 4: Type-level performance of models trained with BERT-derived and GloVe embeddings on McRae Buchanan and Binder feature norm prediction tasks. Bolded cells indicate the highest-performing models for each feature prediction task.

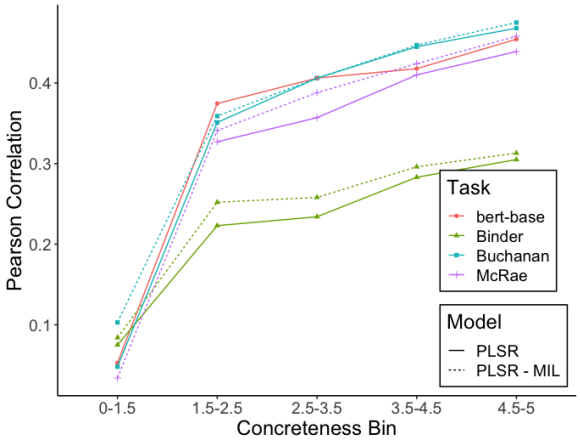

Figure 3: Pearson correlation of cosine similarities between predicted feature vectors with Wu-Palmer similarity between senses. Data: pairs of tokens of the same lemma in SemCor, broken down by lemma concreteness according to Brysbaert.

task, PLSR and label propagation perform the best, but the scores are more or less similar across the board. The best performance was within range of the best MAP@k scores reported in the literature (MAP@k = .36 on McRae, per Rosenfeld and Erk, 2023). BERT embeddings produce features comparable in performance to GloVe vectors. For Buchanan, BERT models do not improve over GloVe vectors. MIL did not fare any better than single-instance learning at the type level, with the exception of PLSR for Buchanan which led to a large performance gain.

These results confirm the finding of Rosenfeld and Erk (2023) that Label Propagation with modified absorption does very well at the task of feature prediction (or property inference, as they call it).

However, as described in the main text, our implementation of Label Propagation is not good at modeling context-sensitive lexical-semantic phenomena unless it is supplied with unlabeled nodes for different senses at training time. Label Propagation under the MIL condition did a particularly good job at disambiguating homonyms (Table 2), provided that the different senses were given as unlabeled nodes during training. However, Label Propagation does very poorly on the sense differentiation task (Table 1), showing that this model does not predict different features for different senses when it is not exposed to unlabeled nodes for these senses during training. We believe this is a consequence of the number of nodes in our graph. At test time, PROP is limited to a fixed number of potential features—given any context vector, it retrieves the closest vector in the graph and gives those labels. Unless there are very many nodes in the graph for each word, PROP will often return the same features for different senses, because there is a high pairwise similarity in BERT space among tokens of the same type (Mickus et al., 2020).

Performance for Label Propagation should improve with the number of unlabeled nodes included during training, but this increases runtime and is not feasible for large datasets or convenient for *ad hoc* linguistic analyses like those we wish to apply the feature prediction model to.

E Concreteness Analysis

In all spaces, concrete polysemous senses are more clearly separated than abstract senses. This is shown in Figure 3, which breaks down sense differentiation results by their concreteness ratings according to Brysbaert et al. (2014). The problem is worst for McRae, and least pronounced for Binder. This may be due to even more variation in meaning for abstract words, which tend to be highly polysemous. Indeed, the same pattern is observed in the frozen BERT space: for concrete words, cosine similarity of token vectors is not strongly correlated with WordNet distance.

Qualitative examination of predicted features reveals that the models are not bad at abstract meanings. For example, consider the sentence "People travel many miles to gaze upon this nat-

Usage Type 1 (transformative)	Usage Type 2 (destructive)	Usage Type 3 (artillery)	Usage Type 4 (cooking)	Usage Type 5 (N-N compounds)
fire	fire	act	fire	fire
hot	hot	fire	hot	person
burn	burn	danger	burn	hot
light	light	kill	light	burn
danger	destroy	weapon	heat	light
heat	danger	human	wood	red
cook	heat	metal	cook	danger
red	cook	loud	danger	heat
wood	hurt	light	warm	cook
act	red	hurt	food	destroy

Table 5: Top 10 predicted Buchanan features for the centroids of 5-k-means clustering on BERT token embeddings from the BNC. (model = PLSR - MIL)

Homonym	Sentence	Gold Feature Norms
bat (animal)	I was particularly surprised to see a tame golden fruit bat, hanging upside down on a tree branch in the morning sunshine.	wing, fly, nocturnal, black, cave, fur, animal
bat (baseball)	I was at the plate. He threw; I swung the bat. The ball rocketed into left field.	hit, wood, ball, metal, long, sport

Table 6: Example data for the Buchanan homonym disambiguation task. Sentences from COCA containing homonyms are paired with a feature norm that targets the disambiguated sense.

ural wonder, though few are willing to approach it closely, since it is reputed to be the haunt of various demons and devils." Our Buchanan model predicts plausible features for the rather abstract 'haunt': 'one', 'face', 'dead', 'bad', 'body', 'place', 'person'. But the McRae model, which did not see abstract words in training and whose features only cover very concrete nouns, does not produce plausible features: 'is_expensive', 'is_smelly', 'made_of_wood', 'is_large'. Predicted Binder features are also plausible: 'Vision', 'Harm', 'Unpleasant', 'Sad', 'Consequential', 'Attention', 'Angry'.

This analysis does not reflect model performance on abstract words so much as it points to a potentially interesting relationship between abstract words in BERT space and in WordNet. Do contextual vectors primarily reflect different kinds of meaning for abstract words besides word sense?

F Top predicted features for sense clusters

Table 5 shows the top 10 Buchanan features for each centroid of the usage type clusters for *fire* (k=5, tokens taken from the BNC). Many of the most salient features are the same across the different usage types. Meanings specific to each sense and usage type are more evident when one focuses on the most *distinctive* features for each cluster (Table 3).

G McRae Homonym Dataset Collection Procedure

We train our contextual model at the type level because of the present lack of in-context feature norms to use for training and evaluation. To evaluate at the token level directly, as described in Section 3, we use the features that McRae et al. (2005) collected for disambiguated homonyms.

For this evaluation, we construct a test set of sentences containing these homonyms, each labeled with the feature vector for that homonym. SemCor, the sense-annotated dataset used for the sense-differentiation evaluation, does not contain enough tokens of each of the homonyms. So, we turned to the Corpus of Contemporary American English (Davies, 2018). The data were collected using the following procedure:

For each homonym, (1) Search for the target word. (2) Read through a random sample of occurrences of the word, highlighting sentences that unambiguously use the target sense. (3) The same researcher double-checks the list to filter out accidental sense mismatches.

At least 20 tokens of each homonym were collected, stopping at 50 (with an average of 40 con-

texts per sense). Table 6 shows two examples from the resulting dataset. The list of homonyms and the number of tokens for each one is given in Table 7, and the full dataset is available in the supplemental data.

Word	Sense	# Tokens
bat	animal	52
bat	baseball	51
board	black	28
board	wood	56
bow	ribbon	43
bow	weapon	52
cap	bottle	20
cap	hat	207
crane	animal	14
crane	machine	101
hose	tube	42
hose	leggings	55
mink	animal	32
mink	coat	33
mouse	animal	64
mouse	computer	78
pipe	plumbing	27
pipe	smoking	20
tank	army	35
tank	container	83

Table 7: List of cue words used in homonym disambiguation experiment along with the number of tokens of each homonym collected from CoCA for the dataset.

H Licenses

Dataset/Model	License
McRae Feature Norms	unknown
Buchanan Feature Norms	GPL 3.0
Binder Feature Norms	CC BY-NC-ND 4.0
Multi-Prototype Embeddings	CC BY-NC 4.0
BNC	http://www.natcorp.ox.ac.uk/docs/licence.html
bert-base-uncased	Apache 2.0
SemCor	Apache 2.0
Brysbaert Concreteness Norms	CC BY-NC-ND 3.0
AANN Sentences	CC BY-NC-ND 4.0

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitations

☑ A2. Did you discuss any potential risks of your work?
Ethics

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract; Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

USED
models - huggingface BERT (base-uncased)(Appendix A)
datasets - Binder feature norms (Section 2) - McRae feature norms (Section 2) - Buchanan feature norms (Section 2) - multi-prototype embeddings: (Section 2; Appendix A) - British National Corpus (Appendix A) - SemCor NLTK implementation (Section 4.1) - Brysbaert Concreteness Norms (Section 4.3) - AANN Data (Section 5.1) - Swapped Subjects dataset (Section 5.2) - Universal Dependencies Treebank (Section 5.2) - WordNet (Section 4.1) - CoCA (Appendix E)
code: - Label Propagation (Section 3 par 3) - Attention-MIL (Appendix B)
CREATED
models - feature prediction models (Section 4)
datasets - homonym disambiguation dataset (Section 4.2; Appendix E)
code - features-in-context library (Section 3)

☑ B1. Did you cite the creators of artifacts you used?
USED
models - huggingface BERT (base-uncased)(Appendix A)
datasets - Binder feature norms (Section 2) - McRae feature norms (Section 2) - Buchanan feature norms (Section 2) - multi-prototype embeddings: (Section 2; Appendix A) - British National Corpus (Appendix A) - SemCor NLTK implementation (Section 4.1) - Brysbaert Concreteness Norms (Section 4.3) - AANN Data (Section 5.1) - Swapped Subjects dataset (Section 5.2) - Universal Dependencies Treebank (Section 5.2) - WordNet (Section 4.1) - CoCA (Appendix E)
code: - Label Propagation (Section 3 par 3) - Attention-MIL (Appendix B)

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
models - huggingface BERT (Apache-2.0)
datasets - Binder feature norms (CC BY-NC-ND 4.0) - McRae feature norms (license unknown; generally used for academic research) - Buchanan feature norms (GPL 3.0) - multi-prototype embeddings: (CC BY-NC 4.0) - BNC (license: http://www.natcorp.ox.ac.uk/docs/licence.html) - SemCor NLTK implementation (Apache-2.0) - Brysbaert (2014) CC BY-NC-ND 3.0 - AANN Data (CC BY-NC) - Swapped Subjects dataset (CC BY-NC) - Universal Dependencies Treebank (CC BY-NC-ND 4.0) - WordNet (WordNet 3.0 License) - CoCA (Custom Academic License: https://www.english-corpora.org/academic_license.asp)
code: - Label Propagation (CC BY-NC-ND 4.0) - Attention-MIL (MIT)

☒ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
We did not specify that we use artifacts in the intended way due to space limitations because they are by and large very commonly used linguistic datasets and were employed in their usual manner. While feature norms (with the exception of Binder norms) were not designed specifically for use with language models, applying them to analyze distributional models is a standard technique.

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
The corpora used are already anonymized of personal information.
Multiprototype embeddings are constructed from randomly sampled sentences from the BNC. The homonym disambiguation dataset is sampled from CoCA. It is possible that offensive content makes its way into these example sentences. Given that we undertake a descriptive study subtle semantic variations in English, and those variations are influenced by social meanings, we determine it wise to not attempt to filter the results in any way. However, they should not be used in any prescriptive machine learning applications where the desired behavior is more important than uncovering statistical patterns in the training corpora.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
USED
models - huggingface BERT (Limitations)
datasets - Binder feature norms (Section 2) - McRae feature norms (Section 2) - Buchanan feature norms (Section 2) - multi-prototype embeddings: (Appendix A) - BNC (Appendix A) - SemCor (Section 4.1) - Brysbaert: to save space, demographics of participants are not listed, as this was not a central part of the work. Issues related to demographics of semantically annotated data are discussed in the Limitations section - AANN Data: Sectiopn 5.1. Templatically constructed - Swapped Subjects dataset (Section 5.2) - Universal Dependencies Treebank (Section 5.2) - WordNet (WordNet 3.0 License). Not included, to avoid redundancy. As we are using this resource to analuze an english language model, it's understood that it is an English language resource. - CoCA - Not included, to avoid redundancy. As we are using this resource to analuze an english language model, it's understood that it is an English language resource.
CREATED
models - feature prediction models (Limitations)
datasets - homonym disambiguation dataset (Section 4.2; Appendix E)
code - features-in-context library (Limitations)

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
USED
datasets - Binder feature norms (Appendix C) - McRae feature norms (Appendix C) - Buchanan feature norms (Appendix C) - multi-prototype embeddings: (Appendix C) - BNC n/a - SemCor NLTK implementation (Section 4.1, Table 1) - Brysbaert (2014) n/a - AANN Data (Section 5.1) - Swapped Subjects dataset (Section 5.2) - Universal Dependencies Treebank: n/a - WordNet: n/a - CoCA : n/a
CREATED
datasets - homonym disambiguation dataset (Section 4.2, Table 2; Appendix E)

C ☑ Did you run computational experiments?

Section 4, Section 5, Appendix D

☒ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Parameters and infrastructure details are reported in Appendix C. We did not report computational budget because runtime information was not saved during experiments. However, all model tuning experiments and data analyses were run on a quad-core personal computer, which means they are not cost-prohibitive to reproduce

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Appendix C, paragraph C. The best-found hyperparameter values for the feature prediction models are listed in the Supplemental Materials, and the best-performing models will be published along with a Colab notebook for using them.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 4, Paragraph 1

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Appendix C, Paragraph 3

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.