Testing Large Language Models on Compositionality and Inference with Phrase-Level Adjective-Noun Entailment

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

Abstract

Previous work has demonstrated that pre-trained large language models (LLM) acquire knowledge during pre-training which enables reasoning over relationships between words (e.g, hyponymy) and more complex inferences over larger units of meaning such as sentences. Here, we investigate whether lexical entailment (LE, i.e. hyponymy or the *is a* relation between words) can be generalised in a compositional manner. Accordingly, we introduce PLANE (**P**hrase-**L**evel **A**djective-**N**oun **E**ntailment), a new benchmark to test models on fine-grained compositional entailment using adjective-noun phrases. Our experiments show that knowledge extracted via In–Context and transfer learning is not enough to solve PLANE. However, a LLM trained on PLANE can generalise well to out–of–distribution sets, since the required knowledge can be stored in the representations of subwords (SW) tokens.

1 Introduction

Composition and entailment are crucial features of human language and reasoning. The first refers to the ability to combine units of meaning, like words or phrases, into larger constructs, such as sentences or paragraphs. Entailment, on the other hand, refers to the notion of inference. A linguistic element A (e.g. a word or phrase) is said to entail an element B if, assuming A is true, so is B. Word-level entailment is often referred to as lexical entailment (LE), *hypernym* detection, or the *is a* relation (Weeds et al., 2014; Vulić and Mrkšić, 2018; Kober et al., 2021), and refers to examples such as dog *entails* ($\models$) animal and gun $\not\models$ weapon. Yet entailment does not occur just between two words, and has been a long standing problem in NLP (Dagan et al., 2005; MacCartney and Manning, 2008; Marelli et al., 2014; Nie et al., 2020). When occurring between two sentences, it is usually referred to as natural language inference (NLI).

Although arguments have been made in favour of a more probabilistic interpretation of the task (see Pavlick and Callison-Burch (2016); Pavlick and Kwiatkowski (2019), inter alia), NLI benchmarks generally abandoned the rigid binary classification for a three way classification, usually involving a `neutral` or `UNK` label[1]. With a few exceptions, e.g., Baroni et al. (2012); Kartsaklis and Sadrzadeh (2016); Kober et al. (2021), NLI is still the main method adopted by the NLP community to jointly study the compositional and inferential abilities of a model. However, commonly used benchmarks frequently contain spurious statistical associations that a model can use to solve the task (Poliak et al., 2018; Dasgupta et al., 2018; McCoy et al., 2019). These cues might be as simple as the presence of negation or lexical overlap (Dasgupta et al., 2018), but can be more complex, and exploit similar syntactic substructures between premise and hypothesis (McCoy et al., 2019).

Popular alternatives to training and testing models on datasets containing significant biases are prompting (Petroni et al., 2019; Do and Pavlick, 2021; Hanna and Mareček, 2021) and In–Context learning (Brown et al., 2020). The success of these paradigms has grown in parallel with the popularity of large language models (LLMs) based on Transformers (Vaswani et al., 2017). LLMs architectures are usually pre-trained with mask or next-sentence prediction tasks, and later fine-tuned on other downstream tasks. Pre-trained LLMs have been successfully used with a prompt-based framework to extract factual information (Petroni et al., 2019) (e.g. Dante, `born_in`, Italy), LE relations (Bouraoui et al., 2020; Hanna and Mareček, 2021) (e.g. car, *is a*, vehicle) and study the more complex entailment in Winograd-style schemata (Do and Pavlick, 2021). Here, we study the impact that pre-training, NLI tuning and supervised learning have on the

[1]Usually matched with `entailment` and `non-entailment/contradiction`.

Proceedings of the 29th International Conference on Computational Linguistics, pages 4084–4100
October 12–17, 2022.

performance of a LLMs tested on compositional entailment, using adjective–noun phrases. That is, we investigate at which stage a LLMs might learn that *red car* $\models$ *vehicle*, as well as *red car* $\models$ *red vehicle*; whilst *fake gun* $\not\models$ *weapon*, even though *fake gun* $\models$ *fake weapon*.

Our main contributions are as follow. First, in Section 3, we introduce PLANE (**P**hrase–**L**evel **A**djective–**N**oun **E**ntailment), a large and automatically annotated resource to evaluate models on phrase–level compositional entailment for the English language. We then provide consistent evidence that knowledge acquired by LLMs during the pre-training phase (Section 4), and during finetuning on NLI tasks (Section 5) is weak, yielding poor and unstable performances on PLANE. In contrast, we show in Section 6 how, in a supervised setting, a model like BERT can effectively generalise to out-of-distribution test sets, and how crucial the role of subword (SW) tokens is to this ability. Finally, our work underlines how the different logical functions associated with the three macro classes of adjectives, frequently ignored or oversimplified, can pose notably different challenges to these models.

2 Related Work

Prompting Among the vast literature on prompting LLMs, the work from Hanna and Mareček (2021) is closely related to ours, and provides evidence that BERT retains information on the hyponym-hypernym relation occurring between two words. The work also shows how crucial the structure of the prompt can be. Garí Soler and Apidianaki (2020) provide evidence on the rich representations that BERT has about scalar adjectives and their intensity. Do and Pavlick (2021) propose a set of detailed entailment-based experiments, using both prompting and finetuning paradigm. Here, Winograd-like scenarios are used to carefully construct sentences that challenge LLM's internal association between two entities. Results strongly suggest that, once a model is not able to rely on those learned associations, the task becomes challenging even after finetuning.

Phrase entailment Compared to NLI, phrase-level entailment (PLE) has received significantly less attention. Baroni et al. (2012) present a set of experiments on compositional entailment considering adjective (e.g., `BIG` *dog* $\models$ *dog*) and quantifier modifications (e.g., `ALL` *dogs* $\models$ `SOME` *dogs*). However, instances were strictly limited to AN $\models$ N, and the class of the modifying adjectives was not discussed or differentiated in the results. Kartsaklis and Sadrzadeh (2016) introduced a manually annotated dataset for PLE, using subject-verb, verb-object, and subject-verb-object phrases. Negative samples were built by reversing each entailment item. In contrast, in our dataset, the label of an item can not be inferred by directional clues (i.e. hyponym-hypernym vs hypernym-hyponym) or by the absence of the hypernym relation between constituent words (e.g. *big cat* $\not\models$ *dog* because *cat* $\not\models$ *dog*). Kober et al. (2021) showed how automatically constructed compound-noun and AN compositional items can be used as a data augmentation method to enhance LE. However, this work filtered out intensional adjectives and assumed that for all other adjectives, $N \models h(N) \implies AN \models h(N)$. AN phrases were also studied within the context of fully formed sentences. The main example is the work from Pavlick and Callison-Burch (2016), that introduced the AddOne dataset. Overall, AddOne resemble the standard NLI benchmark, with sentence as premise and hypothesis, used to formulate a three way (`entailment, non-entailment, UNK`) classification task. However, in this case premise and hypothesis differ only by the presence or absence of a single adjective. Apidianaki and Garí Soler (2021) probed BERT with AddOne to study how it encodes the property of a noun. In contrast, we study the different entailment relations which are valid for different classes of adjectives.

3 PLANE

In this section, we describe the PLANE benchmark. We first outline how each of the three classes of adjectives, intersective (I), subsective (S) and intensional (O), affects the relation between a noun and its hypernym, as well as the noun itself. We then describe the sources used to gather adjectives, nouns, AN phrases, and hypernyms, and the procedure used to generate entailment items.

3.1 Adjective Classes

Adjectives can be divided into three macro classes: intersective (I), subsective (S) and intensional (O). From an entailment perspective, the distinction is based on how they modify a noun, N, with respect to itself as well as with respect to it's hypernyms ($hyps(N)$) (McCrae et al., 2014; Lalisse

	Inference Type (IT)	Intersective (I)	Subsective (S)	Intensional (O)
1	AN $\models$ N	✓	✓	✗
2	AN $\models$ h(N)	✓	✓	✗
3	AN $\models$ Ah(N)	✓	✗	✓

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

and Asudeh, 2015). We focus on three inference types, summarised in Table 1, all starting from an adjective-noun (AN) phrase.

AN phrases containing **intersective** (**I**) adjectives (e.g., *red*, *dead* and *Finnish*) describe a subset of entities subsumed by the noun itself and also a subset of entities which all have that adjective as a property. For example, a *red car* is both a *car* and a *red thing*. Thus, AN phrases containing intersective adjectives satisfy all of the forms of inference types (IT) shown in Table 1. Continuing our example, *red car* $\models$ *car* (IT 1), *red car* $\models$ *vehicle* (IT 2) and *red car* $\models$ *red vehicle* (IT 3).

Phrases with **subsective** (**S**) adjectives (e.g., *small*, *intelligent* and *strong*), describe a subset of entities subsumed by the noun but not a subset of entities which have that adjective as a property. For example, a *small elephant* is an *elephant* but it is not necessarily a *small thing*. Thus, AN phrases containing subsective adjectives satisfy IT 1 and 2 inferences but not IT 3 inferences listed in Table 1. In our example, whilst *a small elephant* $\models$ *elephant* and *small elephant* $\models$ *animal*; *small elephant* $\not\models$ *small animal*.

Intensional (**O**) adjectives (e.g. *fake*, *former*, *possible*) have the exact opposite behaviour of subsective. When an intensional adjective modifies a noun, it negates some of its core properties (e.g. *fake gun* $\not\models$ *gun*) and thus IT 1 inferences do not hold. Inferences with IT 2 also do not hold for intensional adjectives since the modification also directly applies to the hypernym of the noun (e.g., *fake gun* $\not\models$ *weapon*). However, since the adjective modification describes a subset of entities fully disjoint from the noun itself, this new set is usually contained within the subset of entities described using the hypernym of the noun modified by the adjective (e.g., *fake Glock* $\models$ *fake gun* $\models$ *fake weapon*) and thus IT 3 holds.

As in LE, we consider PLE as a binary classification task. We note that an argument on the probabilistic nature of PLE as in Pavlick and Kwiatkowski (2019) could be made. In our modelling scheme, *former president* $\models$ *politician*, and *small mouse* $\models$ *small animal* are formally false (McCrae et al., 2014); but, in the real world, might be judged to be unknown or true. We take the position that these cases require additional knowledge in order to judge them to be true. A *small mouse* $\models$ *small animal* because our knowledge suggests that *mouse* $\models$ *small animal*, and the modification of *mouse* by *small* does not change this. In this work, we assume that only LEs between unigrams are known a priori. We then consider whether LLMs contain the knowledge which will enable us to reason over necessary entailment between AN phrases. Therefore, in our binary classification task, the `negative` label covers all cases which might be judged in the real world to be false, unknown or dependent on additional knowledge.

We now present how the evaluation dataset has been constructed, starting from the source of adjectives (A), adjective-noun (AN) phrases and hypernyms ($hyps(N)$).

3.2 Sources

Adjectives Our main source is the list provided by Lalisse and Asudeh (2015), consisting of 300 items in English. Each adjective is tagged with its class, whether it is weakly or strongly polysemous, and/or context dependent[2]. Further intensional adjectives were added from the dictionary in Kennard et al. (2014). After filtering out all adjectives tagged as context-dependent, wee remained with a total of 312 unique items.

Adjective-Noun phrases To collect compositional and realistic AN phrases[3] we parsed a clean Wikipedia dump (Wilson, 2015) via Spacy[4] (Honnibal and Johnson, 2015). We then filtered out all phrases where the identified adjective was not in the adjective list previously described.

[2]The class of an adjective can vary according to the context or the noun it modifies. Deep, for example, can be intersective, as in *deep lake*, or subsective, like in *deep thinker*)

[3]See Appendix A for further analysis.

[4]We used the `en_core_web_lg` model.

Hypernyms We used Wordnet (Fellbaum, 1998) via the NLTK API to collect nouns' hypernyms. We first filtered out AN phrases that were potentially mislabelled by Spacy as containing a noun, by searching for `noun` synsets. We then queried Wordnet for hypernyms of the noun ($hyps(\texttt{N})$), up to a maximum path distance of 3 and always following the first synset. For AN phrases containing an intensional (O) adjective, this procedure was limited to direct hypernyms (i.e. hypernyms with path distance 1 from the noun). This is to mitigate the fact that IT 2 and 3 inferences might not be always false/true for this class of adjectives. As an example, consider the phrase *alleged thief*. In line with our previous discussion, *alleged thief* `is_not_a` *thief* and *alleged thief* `is_a` *alleged criminal*. However, as we move up the hypernym hierarchy, we find *alleged thief* `is_a` *person*, and *alleged thief* `is_not_a` *alleged person*.

We then filtered out any hypernyms that were already in bigram or multi-word-expressions (MWE) form. Although they present an interesting resource for future investigation, here we focus on the set of unigram hypernyms, to control more precisely the automatic construction of items and mitigate the possibility of including idiomatic phrases. Lastly, test items were further restricted to instances containing nouns occurring at least once within each adjective class. This was done to control for results determined solely by possible strong/weak noun–adjective associations.

Inference Types Once the hypernyms ($hyps(\texttt{N})$) for each `AN` were collected, we automatically constructed all possible positive (✓) and negative (✗) items following the rules presented in Table 1. This converts triplets of the IT <`A`, `N`, $h(\texttt{N})$> where $h(\texttt{N}) \in hyps(\texttt{N})$ into triplets of the IT $<$ c_1, c_2, `label`> where c_1 is the `AN` phrase, c_2 is one of `N`, $h(\texttt{N})$ or `A`$h(\texttt{N})$ and `label` indicates whether an entailment holds between c_1 and c_2.

The final PLANE dataset contains 312 unique adjective, ∼7800 unique nouns and approximately 1.9M unique inference items. The complete benchmark and code for the experiments are openly available[5]

4 In–Context Learning

In this section we investigate the ability of multiple LLMs to solve compositional entailment without any target training. To do so, we adopt an In-Context learning paradigm. With a similarly aim, Hanna and Mareček (2021) evaluated a model's performance on LE by testing if it was able to unmask a prompt *P* such as "A *x* is a `[MASK]`" with a correct hypernym of *x*. Given the phrasal nature of our investigation, we structure our prompts to ask the model whether a particular instance is a *positive* (✓) or *negative* (✗) example of an entailment pair.

Results from Hanna and Mareček (2021) and preliminary Zero-Shot experiments (See Appendix B.1) suggest the performance of a model may be largely affected by its lack of understanding of the task, or particular words in the prompt. Thus, we experiment with a Two-Shot NLI-like format, providing models with some solved examples and background knowledge about entailment, involving the lexical items in the hypothesis. More specifically, we adopt a prompt *P* consisting of two 'labelled' premises and one 'unlabelled' hypothesis, e.g.,:

p_1 : A big car is a good example of a car.

p_2 : A big car is a poor example of a big vehicle.

h : A big car is a `[MASK]` example of a vehicle.

As in the example, each of the three components of *P* (i.e. the two premises and the hypothesis) has a unique inference type (IT). We structure the prompts in this way for two reasons: i) to independently study each <`A`, `N`, $h(\texttt{N})$ > triplets generating every $< c_1, c_2,$ `label` >; ii) investigate if a context that facilitate the identification of an adjective's class, also yields better performances. In the example above, even if a model has no knowledge on the adjective *big*, but knows how subsective (S) adjectives work, it can directly infer from the premises the class of *big*, and, hence, the correct label for the hypothesis. However, if p_2 and h were inverted, the only way a model could solve the instance would be knowing how subsective adjectives work and that *big* is subsective. Lastly, to investigate potential recency effects of the two premises, we query each model with the presented prompt and one with inverted p_1 and p_2. For example, given a hypothesis with IT 3, we consider both premises with IT 1,2 and premises with IT 2,1.

Since *P* contains labelled examples, models can observe the expected label within the given sample. We hence define a set of label's verbalisers for positive (✓), and one for the negative (✗) labels.

[5] https://github.com/lorenzoscottb/PLANE

	Internal	External
$p_{1,2}$	A $\{c_1\}$ is a {verbaliser} example of a $\{c_2\}$.	A $\{c_1\}$ is a type of $\{c_2\}$:{verbaliser}.
h	A $\{c_1\}$ is a [MASK] example of a $\{c_2\}$.	A $\{c_1\}$ is a type of $\{c_2\}$:[MASK].

Table 2: Prompt templates (PT). The two premises (p_*) – hypothesis (h) structures used in the Two–Shot experiment. $c_{1,2}$ refer to the head and tail components of a given inference type (IT) (see Table 1 for reference).

We experiment with two prompt templates, and three label verbalisers, presented in Table 2 and 3 respectively. Given a prompt P, its label l, we define the task as the ability of an LLM to generate, as first prediction for the [MASK], the token t that corresponds to the correct verbaliser for l. Performance is computed via F1 score, since it is possible that t will be different from either of the correct verbalisers.

	✓	✗
GP	good	poor
TF	true	false
PN	positive	negative

Table 3: Labels' verbalisers. Tokens used to verbalise positive (✓) and negative (✗) labels in the Two–Shot experiment.

Selected Models We focus on three families of Transformer networks: BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019) and Distillation based (Sanh et al., 2019) (i.e. DistillBERT and DistillRoBERTa). For RoBERTa and BERT, we considered both base and large models (cased for BERT). Models for this and later experiments were all implemented via Hugging Face (Wolf et al., 2020). All experiments were run on a NVIDIA GeForce RTX 3090.

Results Results from individual models are presented in Table 4, divided by adjective classes. Overall, the performance is fairly poor, with all models presenting low average scores, and remarkable variances across classes. As discussed below, part of this is surely generated by the different verbalisers and prompt templates adopted in the experiment. However, a non-irrelevant part of this variance seems to be directly explained by the adjective class itself. As we can see, all models follow the trend in results associated with each class, finding intersective (I) adjective examples easiest, followed by intensional (O) and then subsective (S). Indeed, the first might not come as a surprise, especially as I adjectives are always associated with a positive label. Yet the fact that rare and anomalously–behaving adjectives such as the intensional (O) ones seem to be easier to deal with than subsective (S), the predominant class in human language, is more unexpected.

PT and verbaliser analysis Across models, prompt templates (PT) and verbalisers have remarkably differed effects on each class and IT (see Figure 6 in Appendix B.2 for summary). Most PT–verbaliser combinations yield almost flawless performances on intersective (I) adjectives, suggesting LLMs are generally keen to choose the same label appearing in both premises. In this class, the variance derives almost entirely from the PN verbaliser. As intersective adjectives are associated just with positive (✓) labels, this evidence suggest a possible association of PN with negative solutions.

Results from subsective (S) items point to similar conclusions. First of, almost all PT–verbaliser combinations struggle to solve instances where the hypothesis has IT 3. That is, when the hypothesis presents the opposite label to both premises. Moreover, PN seems to be again associated with a tendency towards negative labels, especially if combined with the External PT. Such combination is the only one improving the performance on IT 3, but severely damages all other inferences.

In intensional (O) adjectives, where most IT have negative (✗) labels, this association partially affects the TF verbaliser too. However, most models still fail where the hypothesis has opposite label to both premises (IT 3). Overall, this suggests that, whenever presented with premises sharing the same label, regardless of which, models tend to overcome possible internal associations, and opt to repeat the presented label.

Concluding, we note conflicting observations on the recency effect, expected to emerge when p_2 and h share the same label (see Figure 7 in Appendix B.2). The effect has a mostly positive impact on the GP verbaliser (in S and O classes), but contradictory effects on the others, especially PN.

Adj. Class	BERT-base	BERT-large	DistillBERT	DistillRoBERTa	RoBERTa-base	RoBERTa-large
I	69.9 ± 38	78.2 ± 34	70.9 ± 36	83.79 ± 32	97.6 ± 5	99.4 ± 1
S	40.1 ± 36	48 ± 36	34.4 ± 40	19.6 ± 27	38.3 ± 32	41.4 ± 42
O	59.3 ± 40	54. 4 ± 37	52.6 ± 46	61.6 ± 40	48.1 ± 34	44.8 ± 38
Average	56.5 ± 40	60.2 ± 43	52.6 ± 43	55 ± 42	61.3 ± 37	61.9 ± 42

Table 4: Two–Shot learning results. Mean F1 scores (± standard deviation, obtained collapsing prompts' and verbalisers' results) of individual models on the Two–Shot learning experiment, divided by adjective class.

5 Transfer Learning

Evidence from Section 4 suggest In–Context learning is too susceptible to internal correlations and biases to be reliable. Since models trained to classify text for entailment are very popular, we next investigate whether tuning a LLM for sentence level entailment can provide enough information to reliably solve phrase-level entailments from PLANE. For comparison, we re-use the same test from the Two–Shot experiment, re-framing the task as a standard NLI text classification. We replace the standard premises-hypothesis input sentences with a $< c_1,c_2 >$ pair, and evaluate a model's performance in classifying each scenario as presenting an entailment or not. We adopt F1 scores, since, in contrast to PLANE's binary classification, NLI also has a third label (2). This label, often referred to as neutral or `UNK`, usually denotes instances where annotators could not agree on the presence or absence of entailment (1).

Selected Models In the experiment, we use Liu et al. (2019) and Nie et al. (2020) RoBERTa models, both fine-tuned to run NLI-like tasks, and a RoBERTa-base model we tuned on the AddOne benchmark from Pavlick and Callison-Burch (2016). As mentioned, AddOne was designed to study AN composition in the context of full sentences, using premises and hypothesis that differ by a single adjective.

Results Table 5 summarises the results, which appear contrasting. Nie et al. (2020)'s performance is fairly in line with average results of RoBERTa models in the Two–Shot setting (see Table 4). However, in this setting, subsective (S) items seem to obtain a far better performance, especially with respect to intensional (O), suggesting a strong shift towards positive solutions. On the other hand, it appears NLI tuning had a negative impact on Liu et al. (2019)' model. The very high performance observed for intensional adjective strongly suggest a strong preference for contradiction label, as suggested by the error analysis (see Figure 9 in C.1 for visual summary. As for the model tuned on AddOne, the same analysis confirmed that the poor performance across the board depends on a strong preference for neutral labelling. Interestingly, we found that all models share a pattern of predictions for neutral (2) labels (see Figure 9 in C.1 for visual summary). When presented with subsective (S) adjectives, neutral mislabelling is more frequent with positive items, whilst the opposite is true for intensional (O) ones.

Adj. Class	Liu et al. 19	Nie et al. 20	AddOne
I	17.1	90.3	35.9
S	24.1	58.8	32.2
O	57.4	31.1	25.5
Average	32.8	60	31.2

Table 5: Testing NLI models results. F1 scores, divided by adjective class, of RoBERTa models tuned on different NLI benchmarks, and tested on phrase-level entailment. The test set consists of PLANE items used in the Two–Shot experiment.

Variance analysis As mentioned in the introduction, multiple work (e.g. Dasgupta et al. (2018); McCoy et al. (2019)) have shown how biases can arise from syntactic structures. To investigate if the structure of an instance (i.e., the IT) has an impact on each model's performance, we investigate the results divided by adjective class and ITs. The results are summarised in Figure 1.

First off, the image clearly shows the preference of Liu et al. (2019) for negative labels and Nie et al. (2020) for positive ones. Interestingly, we can also see how, at least for these two models, these preferences are strongly accentuated under inference type 1. This effect could be related to the lexical overlap heuristic described in McCoy et al. (2019). This heuristic refers to those instances where the hypothesis (h) contains multiple words from the premise (p), especially within its first tokens. Inference type 1 (i.e. AN $\models$ N) could elicit this bias since h is simply a partial repetition of p.

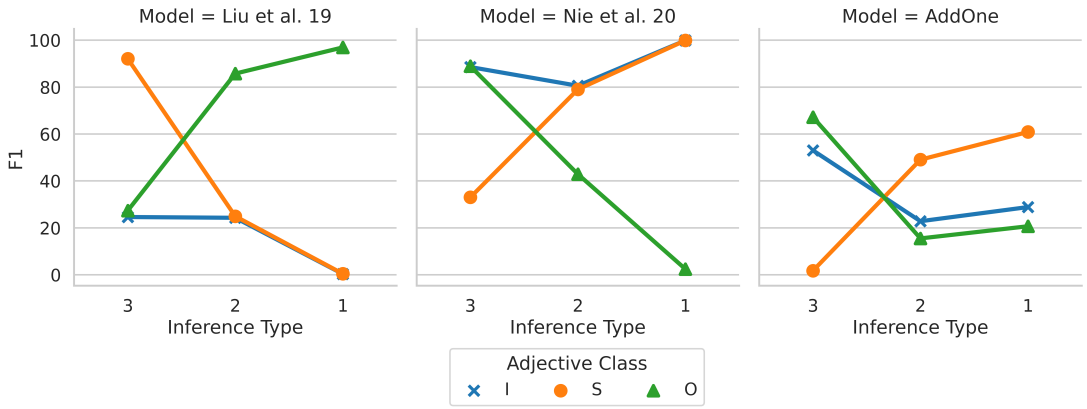


Figure 1: Transfer–Learning variance analysis. Visualisation of the variance observed in different models tuned on NLI-based datasts (column), with respect to each adjective class (hue) and inference type (x axis).

However, McCoy et al. (2019) found that in MNLI (Williams et al., 2018) – Liu et al. (2019)'s training data – such heuristic was mainly associated with a positive label, which is in contrast with our results. It would however partially explain why this behaviour is not expressed by the model trained on AddOne, where the lexical overlap is close to 100% by design, so that a model can not use the heuristic at all. Lastly, it is worth noting how in the model trained on AddOne, subsective (S) adjective display an almost specular pattern to intersective (I) and intensional (O). Observing opposite patterns between S and O is not surprising, as they have opposed labels with respect to each IT (see Table 1). What is unexpected is that I adjectives, always associated with positive labels, produce results almost identical to those of O, where only IT 3 presents a positive label.

6 Supervised Learning

As suggested in McCoy et al. (2019), and supported by preliminary experiments (see Appendix D), drawing the test set from the same distribution of the train set likely over–simplifies the task for LLMs. Hence, to study the performance of a model in a supervised setting, we focus on it's ability to generalise out of distribution (GOoD). Furthermore, we conducted an experiment using a setting where structural cues as the inference types (IT) have been removed (One–IT). As LLMs' vocabularies contain a significant amount of subwords (SW) tokens, together with word tokens, we provide an analysis on the impact of SWs on the model's performance. Following the work of Hanna and Mareček (2021), Do and Pavlick (2021), Apidianaki and Garí Soler (2021), we focus the supervised experiments on a BERT-base model.

6.1 Generalise Out of Distribution (GOoD)

We use PLANE to generate splits where the vocabularies (i.e. adjective, noun, and hypernyms) used in the training and test set do not overlap. That is, each adjective, noun, and hypernym is unique to either the train or the test set. We frame the task as a sequence classification. Following preliminary experiment (see Appendix D), input length is set to 12. We collect 5 different (and openly available) train–test splits, and train the model for 1 epoch. Results are displayed in the left column of Table 6.

Compared to previous results, the performance is strong, remarkably more stable, and is well above chance. The training regime still contains potential structural biases (the ITs), that can facilitate the solution. Yet those cues are useless if not correctly combined with the class of an adjective. Given that single word memorisation is excluded by design, one could assume an effect of pre-training. However, this seems unlikely, given earlier results. Another possibility is that inferences are being made which rely in some way on the constituent subword (SW) tokens of otherwise unseen lexical items.

Training Setting	GOoD	One–IT
Accuracy	.85 ± .05	.86 ± .01

Table 6: Fientuning results. Accuracy (mean ± standard deviations) obtained by BERT, when finetuned on different PLANE–generated splits, in the full generalise out of distribution (GOoD) and One–IT GOoD setting.

Subwords analysis To study the impact of subwords, we compute the accuracies obtained in each test split by BERT, divided by adjective class, and

compare them against the percentage of test instances containing SWs. The results are displayed in Figure 2.

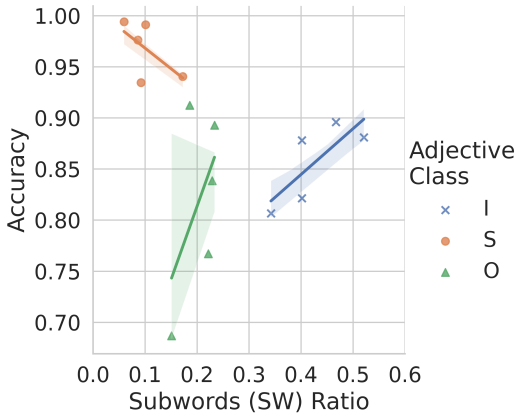


Figure 2: Subword (SW) analysis in GOoD training setting. Analysis of the relation between the amount of sequences containing WP and the accuracy obtained by BERT in each of the five GOoD test splits, divided by adjective class.

We begin noticing that each class seems to cluster around fairly specific SW ratios, which might already facilitate the correct classification of a given input. In sequences with subsective (S) adjectives, SWs are actually all related to nouns and/or hypernyms. This seems to create strong biases that, in the absence of SWs in the adjective position, would suggest to the model that the adjective is subsective, and, hence, the solution. The negative impact that subwords have on S instances might be further explained by the fact that up to 60% of the N/h(N) SWs set overlaps with SWs used in I and 0 adjectives.

A similar overlap also affects intensional (O) adjectives. Up to 65% of adjective subwords overlap with the subwords (SW) used by nouns and hypernyms, and circa 28% also overlap with SWs used for I adjectives. This suggests that, although minimal, an increase in subwords could help the model to identify the correct class of an instance.

Intersective (I) adjectives present the highest ratio of subwrds. Despite the set of adjectives and nouns/hypernyms SWs have similar length, the overlap is very low – between 10 and 7%. This would allow the model to directly exploit SWs to deduct the correct class of an adjective.

6.2 One–IT

The test sets from previous experiment still contained structural cues (ITs) that could assist the model. To study the impact of those cues, we collect new training and test sets, using solely IT 3. We focus on IT 3 as it is the only subset of PLANE where intersective and subsective adjectives, the two largest classes, present opposite labels. Similarly to previous experiment, we balance the number of positive and negative labels, and assure that nouns and hypernyms do not act as cues. We sample five train-test splits, and train with same settings of previous experiment. Results are presented in the right column of Table 6.

The absence of structural cues yields very similar results to the ones from previous experiment, with lower standard deviation and seemingly more stable. Results divided by single split and classes are displayed in Figure 3. SW analysis is also carried out for this training regime, adopting the same setting as in Section 6.1.

Subwords analysis In this setting, Intersective (I) adjectives reached a lower performance, and present a weaker correlation between accuracy and SW ratio. A possible explanation involves the large overlap – circa 50% – in the set of SWs used for adjectives and nouns/hypernyms. Furthermore, the number of instances with and without SWs are remarkably similar, making it potentially difficult to use subwords' presence as cue.

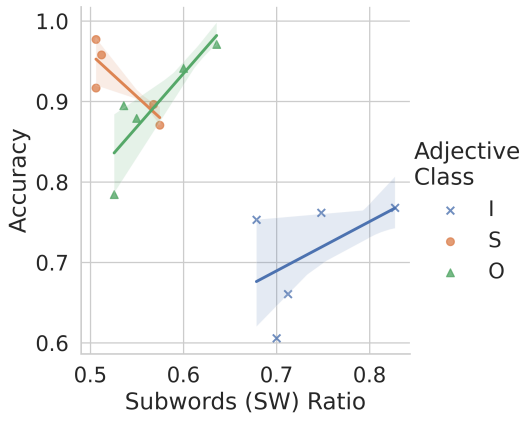


Figure 3: Subword (SW) analysis in One–IT training setting. Analysis of the relation between the amount of sequences containing WP and the accuracy obtained by BERT in each of the five One–IT GOoD test splits, divided by adjective class.

In this experiment, we did found a set of subsec-

tive (S) adjectives containing subwords. However, this set is very small, so the absence of SWs in the adjective position could bias the decision towards negative (✗) labels. Such minimal increase could however act as distractor, explaining the steeper slope of the regression (orange) line.

Once again, O class has the most marked interaction between accuracy and SW ratio. However, in this case, instances with a SW ratio similar to S items do not seem affected. A possible explanation is the very restricted set of SWs (33) used for these adjectives. This smalls set could facilitate an adjective's classification, hence producing a correct solution.

7 Discussion and Conclusion

Adjectives can be grouped in three macro classes. From a logical and linguistic perspective, these classes shape the truth value of a lexical entailment (LE) pair as *dog* $\models$ *animal* in multiple ways, depending on the class and the structure of said inference, as presented in Table 1. This versatility provides a valuable resource to study composition and inference with great detail and control, but was often oversimplified. As previous evidence suggest large language models (LLM) are able to retain word-level entailment information (Petroni et al., 2019; Hanna and Mareček, 2021), we designed a resource to study if LLMs can tackle fine–grained compositional inference, with AN entailment.

Results based on In–Context learning suggest that LLMs' performance is too unstable, and frequently relying on pre-existing word associations or labelling patterns. Conclusions are not so different with models tuned to classify text for entailment. As Section 5 strongly suggests, after tuning a model for sentence–level inference, the knowledge is hardly transferable to the same task at phrase level. These evidence are likely connected to how AN phrases behave within the context of fully formed sentences (Pavlick and Callison-Burch, 2016). From a logical stand, *Japanese economy* $\models$ *economy*. Yet, given a sentence as "*Bush travels Monday to Michigan to make remarks on the Japanese economy.*"[6], potential annotators might say it does not entail "*Bush travels Monday to Michigan to make remarks on the economy.*". Of course this and similar scenarios are influenced by complex commonsense and pragmatical knowledge. Yet this opens interesting questions on how AN phrases in and out of context are related to each other, whether a model should be able to correctly reason over both, and, most importantly, what can we do to make that happen.

Experiments with supervised learning and out–of–distribution test sets suggest that a LLM such as BERT can become robustly efficient, even in absence of structural cues. Our results strongly suggest that the solution is aided by subwords (SW) tokens. Aside from leaking some information to the test set, SW might create biases related to how they distribute in different adjective classes. This solution is computationally efficient and effective, but might pose some limits. This solution is simple, computationally efficient and effective. However, it is unlikely that it provides a theoretically sound model of natural language from the perspective of composition, especially since SW are rarely morphologically grounded (Hofmann et al., 2021, 2022). From a practical perspective, it also poses questions as to how we should define out-of-distribution sets when working with LLMs.

To conclude, we introduced PLANE, an extensive annotated resource to train and test models on compositional phrase-level entailment, using adjective-noun phrases. We provided evidence that knowledge learnt via pre-training or NLI tuning is insufficient to solve the task, and showed how, in a supervised setting, a model like BERT can learn to generalise out of distribution examples, adopting strategies connected to SW tokens. Future work will focus on extending In–Context learning to autoregressive LLMs, using PLANE to evaluate LE models on composition, and investigate a three-way or probabilistic labelling system.

Ethical and Broader Impact Statement

As the work has a mainly theoretical focus, authors do not foresee a significant ethical issue related to the set of experiment. However, we note that a number of intersective (I) adjectives refer to nationality (e.g. English, Italian, Japanese) and religious faith (e.g. Christian, Jewish). It is possible that phrases containing biases and/or stereotypes contained in the WikiDump we adopted might have accidentally ended up in the final version of PLANE. As for the broader impact, we believe our work makes two key contributions: i) offers a tool to investigate in grater detail adjective-noun phrases with respect to inference; ii) provides analyses and evidences in support of the need of taking into account the

[6] Example from Pavlick and Callison-Burch (2016)

distinction between adjective classes, as they pose clearly different challenges to the tested models.

Acknowledgements

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

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

Zied Bouraoui, José Camacho-Collados, and Steven Schockaert. 2020. Inducing relational knowledge from BERT. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7456–7463. AAAI Press.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, T. J. Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *ArXiv*, abs/2005.14165.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *MLCW*.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

Aina Garí Soler and Marianna Apidianaki. 2020. BERT knows Punta Cana is not just beautiful, it's gorgeous: Ranking scalar adjectives with contextualised representations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7371–7385, Online. Association for Computational Linguistics.

Michael Hanna and David Mareček. 2021. Analyzing BERT's knowledge of hypernymy via prompting. In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 275–282, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Valentin Hofmann, Janet Pierrehumbert, and Hinrich Schütze. 2021. Superbizarre is not superb: Derivational morphology improves BERT's interpretation of complex words. 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)*, pages 3594–3608, Online. Association for Computational Linguistics.

Valentin Hofmann, Hinrich Schuetze, and Janet Pierrehumbert. 2022. An embarrassingly simple method to mitigate undesirable properties of pretrained language model tokenizers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 385–393, Dublin, Ireland. Association for Computational Linguistics.

Matthew Honnibal and Mark Johnson. 2015. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378, Lisbon, Portugal. Association for Computational Linguistics.

Dimitri Kartsaklis and Mehrnoosh Sadrzadeh. 2016. A compositional distributional inclusion hypothesis. In *Proceedings of the 9th International Conference on Logical Aspects of Computational Linguistics. Celebrating 20 Years of LACL 1996—2016 - Volume 10054*, LACL 2016, page 116–133, Berlin, Heidelberg. Springer-Verlag.

Neha Nayak Kennard, Mark Kowarsky, Gabor Angeli, and Christopher D. Manning. 2014. A dictionary of nonsubsective adjectives.

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

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

Mathias Lalisse and ash Asudeh. 2015. Distinguishing intersective and non-intersective adjectives in compositional distributional semantics.

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

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

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 216–223, Reykjavik, Iceland. European Language Resources Association (ELRA).

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

John P. McCrae, Francesca Quattri, Christina Unger, and Philipp Cimiano. 2014. Modelling the semantics of adjectives in the ontology-lexicon interface. In *Proceedings of the 4th Workshop on Cognitive Aspects of the Lexicon (CogALex)*, pages 198–209, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

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

Ellie Pavlick and Chris Callison-Burch. 2016. Most "babies" are "little" and most "problems" are "huge": Compositional entailment in adjective-nouns. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2164–2173, Berlin, Germany. Association for Computational Linguistics.

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

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

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

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Yulia Tsvetkov and Shuly Wintner. 2011. Identification of multi-word expressions by combining multiple linguistic information sources. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 836–845, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Aline Villavicencio, Valia Kordoni, Yi Zhang, Marco Idiart, and Carlos Ramisch. 2007. Validation and evaluation of automatically acquired multiword expressions for grammar engineering. In *Proceedings*

of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pages 1034–1043, Prague, Czech Republic. Association for Computational Linguistics.

Ivan Vulić and Nikola Mrkšić. 2018. Specialising word vectors for lexical entailment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1134–1145, New Orleans, Louisiana. Association for Computational Linguistics.

Julie Weeds, Daoud Clarke, Jeremy Reffin, David Weir, and Bill Keller. 2014. Learning to distinguish hypernyms and co-hyponyms. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2249–2259, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

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

Benjamin Wilson. 2015. The unknown perils of mining wikipedia. Https://blog.lateral.io/2015/06/the-unknown-perils-of-mining-wikipedia/.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

A PLANE: PMI analysis

To further control for non—compositional items, we performed a PMI analysis on PLANE's phrases. Tsvetkov and Wintner (2011) showed how higher values of PMI can indicate the presence of a multi–word–expression (MWE), whilst values below zero tend to refer to words that should not really co-occur. Villavicencio et al. (2007) compared the probability distributions of PMI scores from a set of MWE and non—MWE n—grams. The results showed how the distribution of MWE was significantly more skewed towards the upper bound, whilst non—MWE would distribute more normally across observed scores. The distributions of PMI scores of PLANE's phrases, divided by adjective class, are presented in Figure 4.

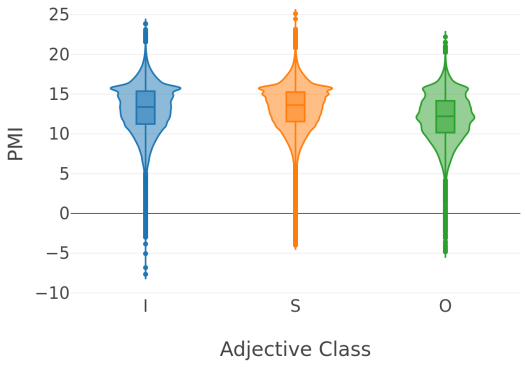


Figure 4: PMI scores by adjective class. Distribution of the PMI scores for each adjective–noun phrase in PLANE.

The median values of all three classes are notably distant from 0 and upper–bound outliers. Phrases containing intersective (I), and subsective (S) adjectives have a strikingly similar distributions, skewed towards higher values. On the contrary, phrases built with intensional (O) adjectives present a slightly lower average PMI score, and appear more evenly distributed. A manual inspection of a subset from phrases with a PMI equal to or higher than 15 didn't identify any idioms or MWE. The same observation holds for the circa 0.1% of phrases with a score equal to, or lower than, 0.

B In–Context Learning

B.1 Zero-Shot Preliminary experiment

As in Section 4, our preliminary Zero–Shot experiment focused on an unmasking problem. We adopted the same prompt templates of Table 2. However, in this case no contextual examples were included within each prompt, so models were not expose to either of the possible labels' verbalisers. We hence built a conversion table V by manually collecting sensible tokens from the set of commonly retrieved ones. Table 7, present the collected conversion table V, mapping potential verbalisers to the `positive` (✓) and `negative` (✗) labels.

✓	✗
good	poor
true	false
positive	negative
great	bad
possible	impossible
plausible	implausible
acceptable	unacceptable
strong	weak

Table 7: Verbalisers adopted for positive (✓) and negative (✗) samples in the Zero–Shot experiment.

Results Results divided by adjective class and model are presented in Table 8. RoBERTa models appear to perform the best, showing also the least amount of variance between the base and large variation of the model. BERT models are the second best performing family. Interestingly, BERT-base seems to outperform its large counterpart. Distillation based models clearly produce the worse results across the all board. Ignoring DistilBERT, results appear to follow the same pattern: performance is the highest on the intersective (I) class, followed by subsective (S) and then intentional (O).

Prompt template analysis Results in Figure 5 provide the mean F1 performance divided by adjective class (coloured lines), forms (x axis), and prompt template (PT, column). Error bars refers to standard deviations, and illustrate the variance produced by collapsing each model's performance.

As it can be clearly appreciated by the Figure, the vast majority of the variance can be attributed to the prompt template (PT). Whilst under the `Internal` PT models where partially able to interpret the given task, the `External` PT made it almost impossible to produce a correct prediction. Lastly, the fact that, regardless of PT and adjective class, ITs associated with negative labels shows a performance close to zero strongly suggest that, without contextual information, most models strongly prefer positive solutions.

Adj. Class	BERT-base	BERT-large	DistilBERT	DistilRoBERTa	RoBERTa-base	RoBERTa-large
I	35.8 ± 25.2	24.1 ± 16.9	7.8 ± 5.4	5.3 ± 3.7	42.5 ± 29.9	42.2 ± 29.8
S	21.9 ± 14.8	13.9 ± 9.2	8.3 ± 5.6	5.0 ± 3.5	23.1 ± 16.2	23.1 ± 16.3
O	5.9 ± 2.2	6.4 ± 1.5	3.3 ± 1.3	1.8 ± 0.3	8.6 ± 4.9	7.9 ± 5.2
Average	21.2 ± 14.1	14.8 ± 9.2	6.5 ± 4.1	4.0 ± 2.5	24.7 ± 17.0	24.4 ± 17.1

Table 8: Zero–Shot learning results. Individual model's performances (mean F1 ± standard deviation from prompt template (PT)), divided by adjective class.

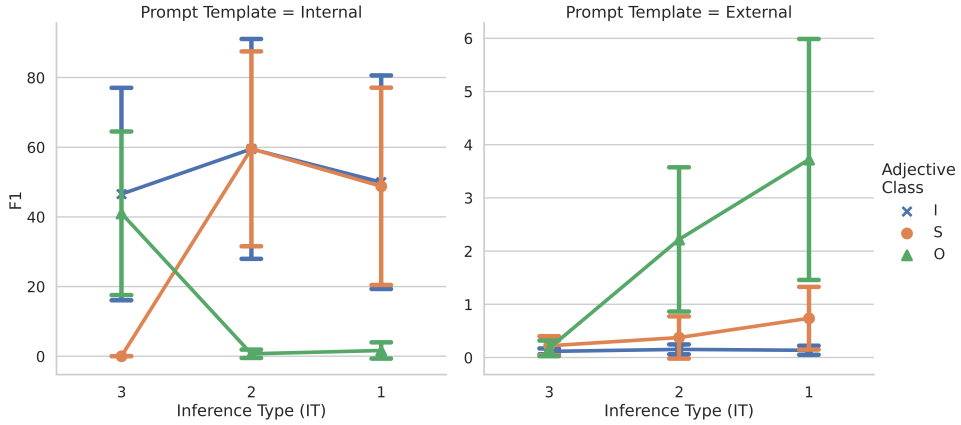


Figure 5: Variance analysis of the Zero–Shot experiment. The graph displays the mean F1 and standard deviation (bars, obtained by collapsing models' performance) obtained on different adjective classes. X-axis refers to the inference type (IT) of the test-items.

B.2 Two-Shot: visualising variance and recency effect

The analysis on the variance generated by prompt templates (PT) and verbalisers is presented in Figure 6. Mean F1 performance is provided, collapsed by models' and premises' permutations [7]. Each column represent an adjective class (I, S and O, respectively), whilst the rows identify the two PT: `Internal` and `External`, respectively (see Table 2).

Figure 7 present the results in further detail, divided by single permutations of a sequence's premises to visualise possible recency effect.

C Transfer Learning

C.1 visualisation of error analysis

The section provides a visual summary of the error analysis in Section 5 via Figure 9.

D In–Distribution Compositional Generalisation

The In–Distribution generalisation experiment presents the same setting as Section 6, with fundamental difference that test set do not contain out–of–distribution items. Following Keysers et al. (2020)'s notation, given a dataset, we identify a set of *atoms*, single words (i.e., adjectives, nouns, hypernyms) and inference types (IT), and a set of *compounds*, which are combination of these three elements. Hence, for this experiment, we generated training and test splits with overlapping atoms, and disjoint compound distributions. Results in term of accuracy against maximum sequence length are presented in Figure 8.

First, cutting the maximum length of the input sequence to 6 tokens produces chance–level performance. As two–third of test items are composed of only 6 words, this suggested that: i) a consistent portion of the input sequences gets split into WordPieces (WP); ii) our splitting algorithm successfully generated sets without biases or c_1–`label` association the model could use to solve the task. As soon as input sequence length reaches 12, the task becomes, as predictable, trivial. WP might still play a minor role – accuracy is still not 1 with sequence length of 8). However, since the atoms, and the adjectives especially, are shared between train and test, the model technically has all the information need to infer the correct label: combining the adjective with inference type (IT)

[7] That is, when testing items with hypothesis of IT 3, we combine results for premise with IT sequences 1,2 2,1.

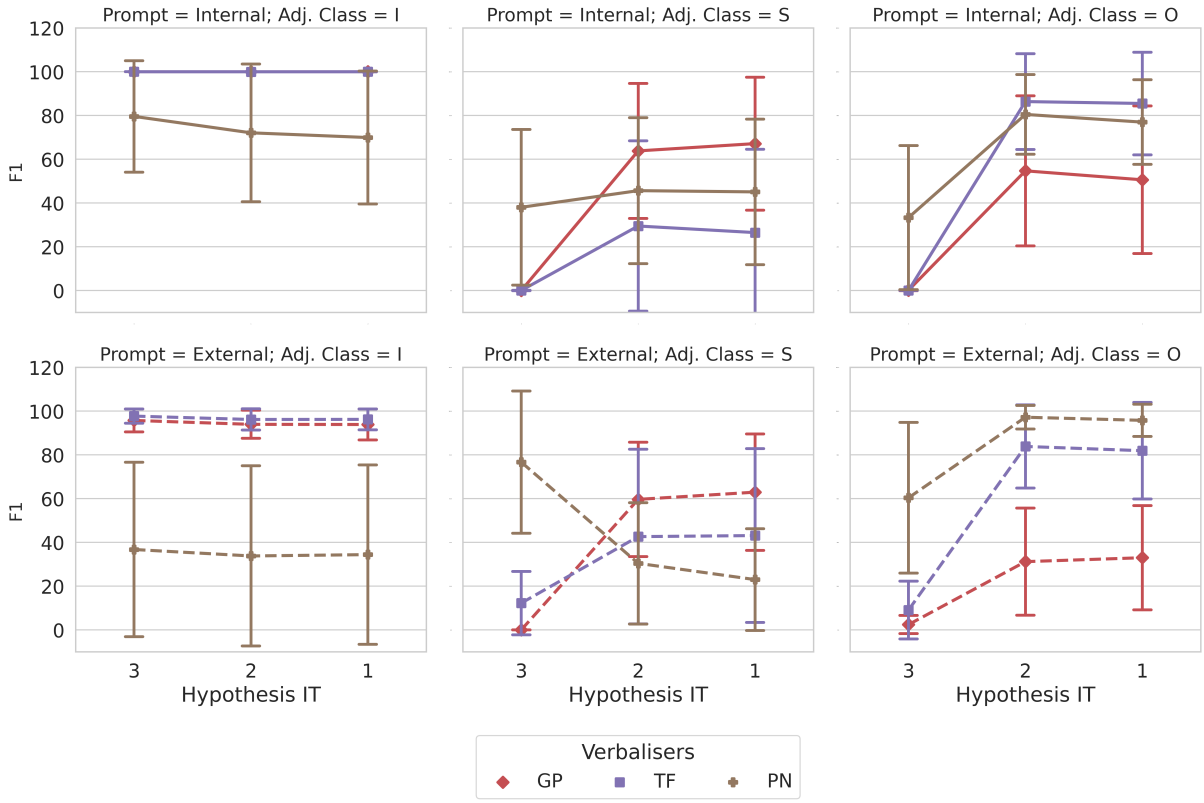


Figure 6: Variance analysis of the Two–Shot experiment. Each graph displays the mean F1 and standard deviation (shown via error bars, generate by collapsing models' performance) obtained by different verbalisers, on a specific combination of prompt template (rows) and adjective class (columns). X-axis refers to the inference type (IT) presented in the hypothesis of the test-items.

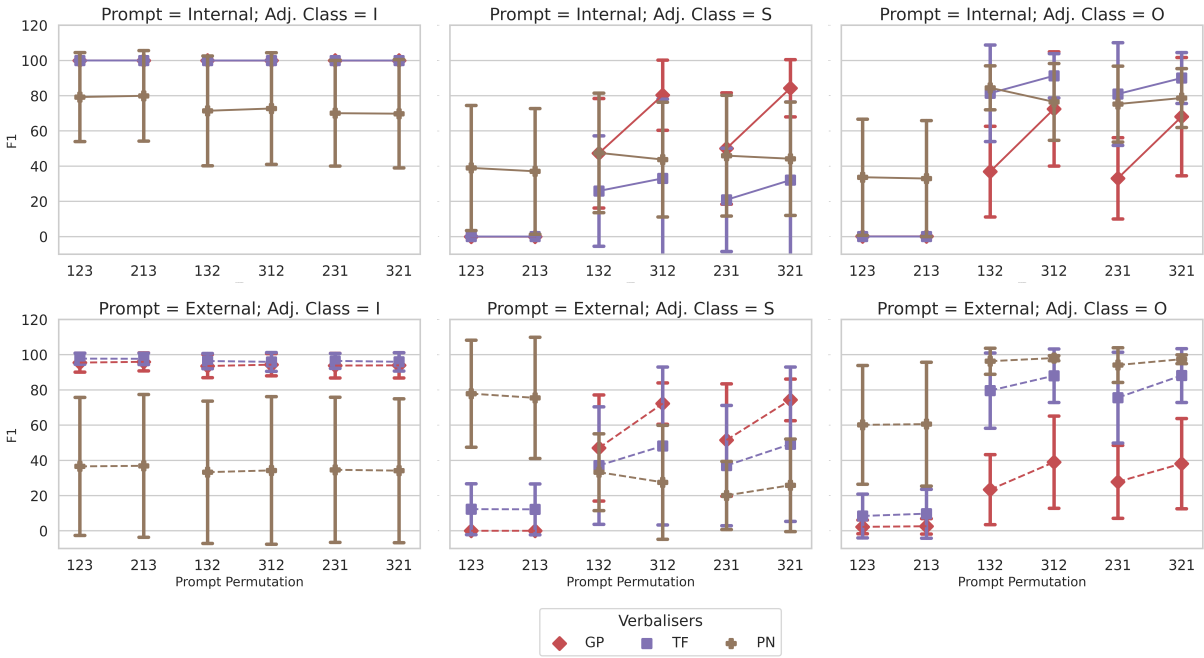


Figure 7: Variance and recency effect analysis of the Two–Shot experiment. Each graph displays the mean F1 and standard deviation (shown via error bars, generate by collapsing models' performance) obtained by different verbalisers, on a specific combination of prompt template (rows) and adjective class (columns). X-axis refers to the inference type (IT) sequence presented in the test-items.

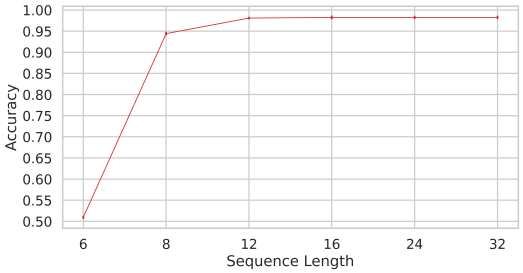
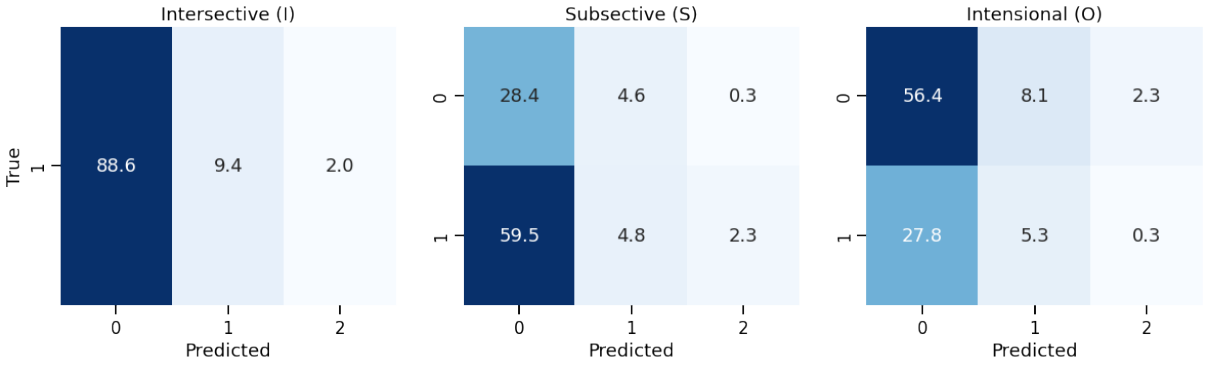
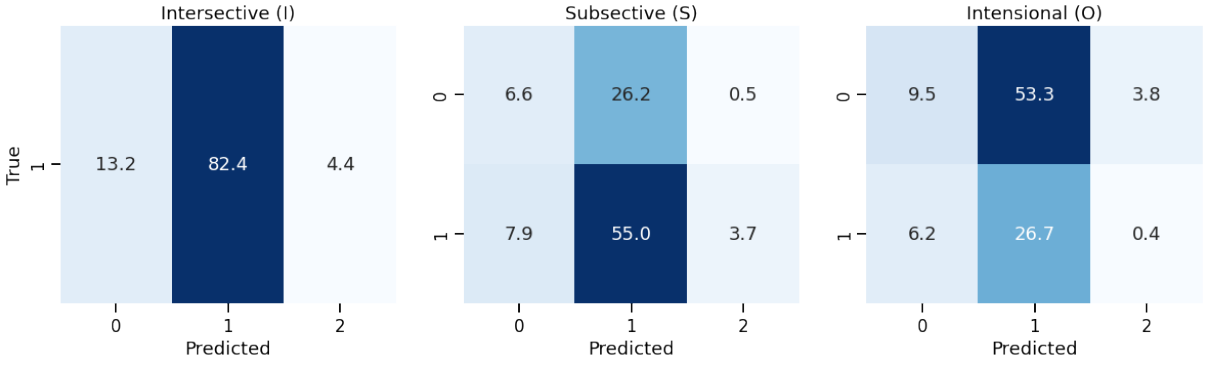


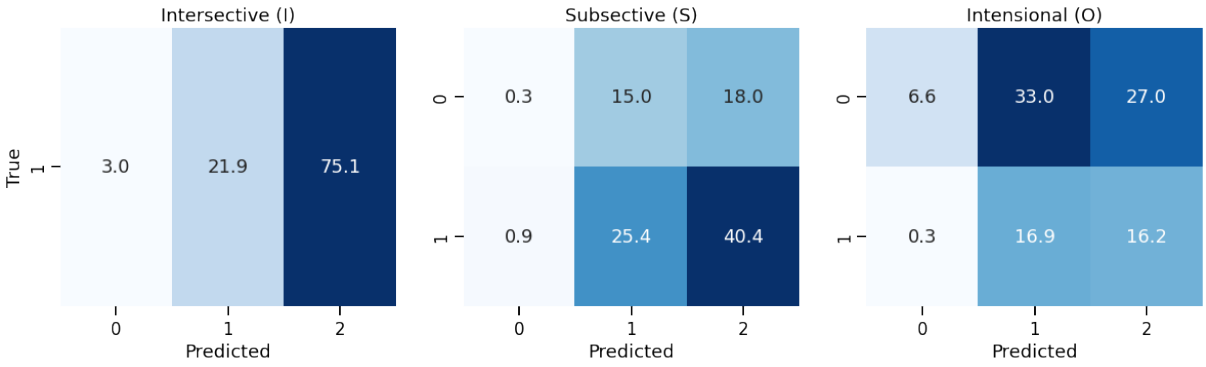
Figure 8: In–Distribution generalisation results. Impact of the input sequence length on accuracy in the compositional generalisation experiment.



(a) Liu et al. (2019)



(b) Nie et al. (2020)



(c) AddOne

Figure 9: Testing NLI models error analysis'. Confusion matrices of the three RoBERTa models tuned on different NLI benchmarks, and tested on PLANE instances from the Two-Shot experiment.