Temporal and Aspectual Entailment

Thomas Kober
University of Edinburgh
tkober@inf.ed.ac.uk

Sander Bijl de Vroe
University of Edinburgh
sbdv@ed.ac.uk

Mark Steedman
University of Edinburgh
steedman@inf.ed.ac.uk

Abstract

Inferences regarding *Jane's arrival in London* from predications such as *Jane is going to London* or *Jane has gone to London* depend on *tense* and *aspect* of the predications. Tense determines the temporal location of the predication in the past, present or future of the time of utterance. The aspectual auxiliaries on the other hand specify the internal constituency of the event, i.e. whether the event of *going to London* is completed and whether its consequences hold at that time or not.

While tense and aspect are among the most important factors for determining natural language inference, there has been very little work to show whether modern NLP models capture these semantic concepts. In this paper we propose a novel entailment dataset and analyse the ability of a range of recently proposed NLP models to perform inference on temporal predications. We show that the models encode a substantial amount of morphosyntactic information relating to tense and aspect, but fail to model inferences that require reasoning with these semantic properties.

1 Introduction

Tense and aspect are two of the main contributors to the semantics of a proposition, describing the temporal location of a predication and its internal constituency, thereby considerably influencing the entailment relations it licenses. For example, while *arrive in LOC* $\models$ *be in LOC* is generally considered a valid entailment rule, the case is complicated when different tenses and aspectual auxiliaries[1] of a given verb are considered as sentences (1) and (2) illustrate.

(1) Jane *has arrived* in London.
$\models$ Jane *is* in London now.

(2) Jane *will arrive* in London.
$\not\models$ Jane *is* in London now.

Understanding the difference between an event that has happened and whose consequences hold at the present moment, and an event that is currently happening or will happen in the future, is crucial for answering questions such as *Where is Jane?* or *Is Jane in London now?* Inferring the consequences of events is important for understanding the relation between entities in the world. For example, if we read that *Lady Catherine has bought Longbourn estate*, the inference that the acquisition is *completed*, and that the resulting consequence is that Lady Catherine now *owns* Longbourn estate, is paramount for keeping knowledge bases up-to-date.

In this paper we propose a novel entailment dataset that requires models to correctly determine the internal and external temporal structure of predications when performing natural language inference. To the best of our knowledge, this is the first dataset that is primarily focused on assessing natural language inference between temporally and aspectually modified predications.

[1] For brevity we will refer to predications with different tenses and aspectual auxiliaries as *temporal predications*.

As a first evaluation on our new dataset we compare to what extent five distributional embedding models, `word2vec` (Mikolov et al., 2013), Anchored Packed Trees (Weir et al., 2016), `fastText` (Bojanowski et al., 2017), ELMo (Peters et al., 2018), and BERT (Devlin et al., 2018), and two bi-directional LSTM (biLSTM) encoders, pre-trained on SNLI (Bowman et al., 2015) and DNC (Poliak et al., 2018), respectively, are able to perform natural language inference on temporal predications. In our evaluation, we refrain from fine-tuning any of the models as our goal is to assess to what extent tense and aspect are captured in these models *per se*. As a pre-requisite diagnostic task for natural language inference between temporal predications we analysed whether the models encode the morphosyntax of tense and aspect and found that they capture a considerable amount of morphosyntactic information in their respective embedding spaces. However, neither of the models outperforms a majority class baseline on our proposed dataset due to their reliance on contextual similarity for performing inference, suggesting that models based on distributional semantics struggle with the more latent nature of tense and aspect. Our contributions in this paper are as follows:

- We assess the extent to which the models in our evaluation encode information about the agreement between an inflected verb and its aspectual auxiliary, and whether a translation operation between different tenses can be learnt from the embedding spaces.
- We propose a novel entailment dataset that requires models to perform inference with temporal predications, and evaluate the five embedding models and two pre-trained biLSTM encoders.
- We analyse the performance of the models and show that their reliance on contextual similarity is problematic for correctly modelling natural language inference governed by tense and aspect.

2 Tense, Aspect and Entailment

Tense is a grammatical category which is encoded in the morphology of the verb in English (e.g. past *loved* vs. non-past *loves*). It establishes a point of reference that allows the temporal organisation of events in a discourse. In English, tense interacts with aspectual auxiliaries such as the verbs *be* or *have* that influence the internal constituency of a predication, and determine whether an event is completed or ongoing. Tense and aspect therefore control the internal and external temporal structure of an event and govern the inferences that a predication licenses (Reichenbach, 1947; Dahl, 1985; Steedman, 1997). There is evidence that such morphology is represented in distributional embeddings (Mitchell and Steedman, 2015; Vylomova et al., 2016). In this paper we are concerned with perfect and progressive aspect, but do not focus on any other types of aspect such as the *Aktionsart* of a predication (Vendler, 1957), which we leave to future work.

2.1 The Interaction between Temporality and Entailment

Perfect aspect (typically) describes events as a completed whole, and licenses inferences regarding the consequences of that event. The use of different tenses and aspects for past events influences their relevance to the present moment and thereby their entailment behaviour. For example, the consequences of an event in the present perfect hold at the time of utterance, whereas events in the simple past or the past perfect do not (Comrie, 1985; Moens and Steedman, 1988; Depraetere, 1998; Katz, 2003). This is shown in sentences (3) and (4), where only sentence (3) licenses the inference of Elizabeth being in Meryton *now*.

(3) Elizabeth *has gone* to Meryton.
$\models$ Elizabeth *is* in Meryton now.

(4) Elizabeth *went* / *had gone* to Meryton.
$\not\models$ Elizabeth *is* in Meryton now.

This property can be explained through a Reichenbachian view of the present perfect, where the point of reference coincides with the point of speech, thereby indicating its current relevance (Reichenbach,

1947). On the other hand, events in the past simple or the past perfect license inferences for consequent states in the past, as sentence (5) shows.

(5) Elizabeth *went* / *had gone* to Meryton.
$\models$ Elizabeth *was* in Meryton.

(6) Mary *is going* to Netherfield now.
$\not\models$ Mary *has arrived* / *is* in Netherfield.

Progressive aspect describes ongoing events and therefore does not license inferences regarding their consequences as sentence (6) shows. It furthermore gives rise to the imperfective paradox (Dowty, 1979), which only seems to license inferences for non-culminated processes (Moens and Steedman, 1988), as sentences (7) and (8) show.

(7) Catherine *was walking* in the woods.
$\models$ Catherine *walked* in the woods.

(8) Jane *was reaching* London.
$\not\models$ Jane *reached* / *was in* London.

The modal future introduces an event whose realisation is uncertain, therefore any inferences about its outcome are only licensed if common-sense knowledge suggests that this is almost always the course of events as sentence (9) shows.

(9) Charles *will meet* with Jane.
$\models$ Charles *will see* Jane.

The correct treatment of tense and aspect in a predication is crucial for inferring the consequences it licenses, which is important for answering questions about a given paragraph, or creating and updating knowledge bases.

3 Models

We analyse five distributional embedding models and two pre-trained biLSTM sentence encoders for their ability to perform inference on temporal predications. Our choice of models is motivated by the observation that modelling entailment between temporal predications requires a bespoke representation of the inflected verb in the context of the given aspectual auxiliary and its arguments.

`word2vec`. We evaluate the ability of `word2vec` representations for performing inference with temporal predications. Contextualisation[2] can be achieved by averaging two word vectors, which has been shown to be a strong baseline for a range of problems (Iyyer et al., 2015; Wieting et al., 2016). Notably, adding or averaging word vectors approximates the intersection of their feature spaces (Tian et al., 2017).

APTs. Anchored Packed Trees are a recently proposed vector space model that take distributional composition to be a process of lexeme contextualisation. APTs are based on a higher-order dependency-typed structure that gives rise to a weighted, directed and labelled graph. Contextualisation is achieved through distributional composition, which requires aligning two lexemes according to their syntactic relation, and then merging the aligned representations. APTs are the only count-based (i.e. non-neural) model in our evaluation.

`fastText`. The `fastText` model represents each word as a sum of bag-of-character n-grams, thereby making better use of subword information and therefore — potentially — providing a better mechanism for encoding morphosyntactic relations. Contextualisation is achieved through averaging the respective word vectors in a phrase.

ELMo. ELMo is based on a deep bidirectional LSTM language model that creates multiple layers of representations for every token. Contextualised representations are obtained from the internal states of the LSTMs, where Peters et al. (2018) showed that lower levels of the architecture capture syntactic characteristics, and higher-levels capture semantic characteristics of words.

[2]We refer to expressing the meaning of a word in its context as *contextualisation*.

BERT. BERT uses multi-headed bi-directional self-attention and is based on the Transformer architecture (Vaswani et al., 2017). Devlin et al. (2018) observed that sequential language model architectures are limited by the unidirectionality of the models. Therefore they proposed a novel training objective that jointly conditions on left and right context in all layers. They showed that their training regime results in substantial gains over serial language model-based architectures on numerous NLP tasks.

Pre-trained biLSTM. For our new entailment dataset, we pre-trained two bi-directional LSTM (Hochreiter and Schmidhuber, 1997) sentence encoders on SNLI (Bowman et al., 2015) and DNC (Poliak et al., 2018), representing two recently released large-scale entailment datasets. Our choice of biLSTMs was motivated by their strong performance in recent studies (Balazs et al., 2017; Conneau et al., 2017).

`Word2vec`, APTs and `fastText` follow the *one representation per word* paradigm (Kober et al., 2017), where every lexeme is represented by one vector, and contextualisation is typically achieved through distributional composition. ELMo, BERT and the pre-trained biLSTMs, on the other hand ,create context-sensitive representations on the token level. This results in different representations for the same word, depending on its current context.

4 Experiments

We created two experiments to assess the extent of morphosyntactic information relating to tense and aspect that is encoded in the respective embedding spaces. Subsequently we propose a novel entailment dataset and evaluate the capability of the embedding models and the pre-trained biLSTMs to perform inference on temporal predications. All our resources are available from `https://github.com/tttthomasssss/iwcs2019`.

4.1 Auxiliary-Verb Agreement

The first experiment evaluates whether the models are able to capture the agreement between an inflected verb and its corresponding aspectual auxiliary. For example, the models should be able to determine that *will visit* represents a correct combination whereas *will visiting* does not. We consider capturing the morphosyntactic interplay between an inflected verb and its aspectual auxiliary a pre-requisite for adequately modelling the semantics of tense and aspect.

We cast the problem as a classification task with the goal of distinguishing correct auxiliary-verb pairs from incorrect ones with a diagnostic classifier. This methodology is similar to the approach of Linzen et al. (2016) who assessed the ability of LSTMs to learn number agreement in English subject-verb phrases. For the dataset, we extracted verbs from the One Billion Word Benchmark (OBWB) (Chelba et al., 2013) where each inflected verb form occurred at least 50 times. We then paired the inflected verb forms with their corresponding auxiliaries to form positive pairs, and subsequently paired each of the different inflected verb forms with all incorrect auxiliaries to build the negative pairs. We filtered the negative pairs for plausible combinations such as *is eaten* by removing valid passive constructions and any invalid combination that occurred at least 5 times in the OBWB corpus. The final dataset consists of almost $36k$ auxiliary-verb combinations with a positive : negative class distribution of 38 : 62.

4.2 Translation Operation

In the second experiment we assess whether it is possible to learn a translation operation between different tenses in the embedding space. We consider learning a translation operation in two ways: firstly a simple vector offset on the basis of the averaged difference between inflected verbs with their auxiliaries and their respective lemmas. Secondly, we train a feedforward neural network to project the infinitive representation of a verb to one of its inflected forms. The goal for both approaches is then to generate an unseen inflected verb form from a given unseen lemma.

The averaged offset translation is shown in Equation 1, where the offset o_t is calculated on the basis of a set of seed verbs S of size n, and vector representations x_t and x_ℓ of the inflected form, or contextualised form if the tense requires an auxiliary, and lemma form of the verb x, respectively. At

prediction time, we are trying to create x'_t by adding the offset o_t to the lemma x'_ℓ (where $x' \notin S$). Equation 2 shows the setup where we use a neural network to learn a translation matrix from infinitive forms to inflected forms, where f is a tense-specific neural network with a single hidden layer, that takes an unseen lemma representation x'_ℓ as input and generates an inflected form x'_t, and where Θ_t represent the learnable parameters of the network.

$$o_t = \frac{1}{n}\sum_{x \in S} x_t - x_\ell \qquad (1) \qquad\qquad x'_t = f(x'_\ell; \Theta_t) \qquad (2)$$

We subsequently evaluate whether the correctly inflected verb is in the nearest neighbour list of the generated verb. The inflected verb generation setup is inspired by Bolukbasi et al. (2016) and Shoemark et al. (2017), who used a similar method in their respective works. For the dataset, we extracted verbs from the OBWB corpus where each inflected verb form occurred at least 50 times, resulting in $\approx 2.8k$ verbs per tense.

4.3 Entailment with Temporal Predications

Lastly, we propose **TEA** — the **T**emporal **E**ntailment **A**ssessment dataset. **TEA** contains pairs of short sentences with the same argument structure that differ in tense and aspect of the main verb, and follows a binary label annotation scheme (*entailment* vs. *non-entailment*). Example sentences from **TEA** are shown in Table 1. The absence and infeasibility of creating a lexical resource for consequent state

John *is visiting* London.	$\models$	John *has arrived* in London.
John *will visit* London.	$\not\models$	John *has arrived* in London.
John *is visiting* London.	$\not\models$	John *has left* London.
John *is visiting* London.	$\models$	John *will leave* London.
George *has acquired* the house.	$\models$	George *owns* the house.
George *is acquiring* the house.	$\not\models$	George *owns* the house.

Table 1: Examples from **TEA**.

inference patterns creates the necessity for NLP systems to learn these rules from data. With **TEA**, we cast the problem of determining when a new consequent state is licensed by an event as a natural language inference task, thereby providing a first evaluation set for modern NLP models.

Data Collection. We sampled candidate pairs from the before-after category of VerbOcean (Chklovski and Pantel, 2004), the WordNet verb entailment graph (Fellbaum, 1998), the entailment datasets of Weisman et al. (2012) and Vulić et al. (2017), and the relation inference dataset of Levy and Dagan (2016). Subsequently, we manually filtered the list, and discarded candidate verb pairs without any temporal relation to each other. For each pair we chose nouns as arguments to form full sentences. The arguments further served the purpose of reducing ambiguity and avoiding habitual readings.

TEA covers entailments between an all-by-all combination of the present simple, present progressive, present perfect, past simple, past progressive, past perfect and the modal future, covering perfect and progressive aspect. The dataset contains 11138 sentence pairs with a class distribution of 22 : 78 (entailment : non-entailment). More detailed dataset statistics are presented in Appendix A.

Data Annotation. We interpreted entailment as common-sense inference (Dagan et al., 2006), and considered a positive entailment relation between two temporal predications if a human annotator would decide that sentence 2 is *most likely* true given sentence 1. We decided against a crowdsourced annotation of **TEA** as our aim was to maximise the consistency of fine-grained entailment decisions. Therefore, **TEA** was labelled by two annotators[3], where the first round of annotation resulted in just under 20% disagreement across the whole dataset. The relatively high level of disagreement suggests that even for annotators who (more or less) know what they are looking for, assessing whether an entailment holds between two temporal predications is a very challenging task.

[3]The first and second author of this paper.

Disagreements in **TEA** were resolved on a case-by-case basis and all sentence pairs with an initial disagreement have been resolved and included in the dataset. We found that with temporality involved, suddenly *everything* appeared to become uncertain. Hence we approached the disagreement resolution by first discussing which of several possible readings is the strongest, and whether that reading is sufficiently more likely than any other possible reading. Subsequently we discussed whether the strong reading is above the *almost always true* threshold.

Often, disagreements resulted from different assumptions regarding the ordering of the events' nuclei. For example, even if we accept that *buys* entails *chooses*, *will buy* does not necessarily entail *will choose*. The reason is that this pair is ambiguous between two readings, a "has-just-chosen-and-now-will-buy" reading on one hand, and a "will-choose-and-then-will-buy" reading on the other, which seem to be equally likely in the absence of any further context[4].

Even when ordering was clear, however, disagreements could arise over beliefs of when an utterance becomes licensed. Saying *will graduate*, for example, can be considered reasonable at any time, or only once *graduation* is sufficiently imminent and likely. In the latter case, *is studying* can be considered sufficiently likely to be an entailment, while in the former case the entailment is less clear[5]. Overall, world knowledge and intuition played into disagreements heavily, causing cases to fall just above or below the common-sense inference threshold depending on the annotator.

We identified a possible annotation artefact in **TEA** due to our decision to annotate the dataset sequentially rather than randomly. While this greatly reduced the cognitive load, we were confronted with possible contradictions between different tenses of entailed predicates (for example, a single event cannot happen in the past *and* the future). This initially led to more conservative annotations, since some pairs when viewed independently can sound very plausible. We tried to factor out this source of bias when resolving the disagreements, and are confident that the annotations in **TEA** are robust.

An interesting avenue for future work would be adding temporal adverbials to further reduce ambiguity for annotators — and to analyse whether models can handle them correctly. The addition of temporal adverbials might alleviate the temporal ordering ambiguity, as for example reading *will buy in 5 years* might help us conclude the ordering with *will choose*, since *choosing* is probably near *buying*.

5 Results and Analysis

For our experiments we used the publicly available versions of each embedding model. For the evaluation on **TEA**, we trained two biLSTMs on SNLI and DNC in addition to the embedding models, achieving 83% and 88% accuracy on the SNLI and DNC development sets, respectively. Appendix B lists further details for all models.

5.1 Auxiliary-Verb Agreement

For assessing whether the auxiliary-verb agreement can be detected with a diagnostic classifier, we built a binary classification task, using stratified J-K-fold cross-validation (Moss et al., 2018) and report averaged accuracy. We used the scikit-learn (Pedregosa et al., 2011) logistic regression classifier with default hyperparameter settings.

The results in Table 2 show that the representations of APTs and BERT are specific enough for a linear classifier to distinguish plausible from implausible combinations. The reason for the strong performance of APTs stems from its sparsity — plausible auxiliary-verb combinations result in representations with numerous non-zero entries, whereas implausible combinations rarely contain more than a handful of non-zero elements. While `word2vec` and `fastText` seem to capture the morphosyntactic relation between an auxiliary and an inflected verb to some extent, their performance is substantially worse than APTs and BERT. Somewhat surprisingly, the results for ELMo are worse than the majority class baseline for all auxiliaries. One possible reason for the comparatively weak performance of `word2vec`,

[4]In this case we decided that if *will buy* is true, the choosing didn't happen yet, so *will buy* $\models$ *will choose*.

[5]We decided *will graduate* $\models$ *is studying*.

Auxiliary	word2vec	APT	fastText	ELMo	BERT	Majority Class
is	0.65 (+/- 0.02)	0.88 (+/- 0.01)	0.67 (+/- 0.02)	0.52 (+/- 0.01)	**0.90** (+/- 0.01)	0.53
will	0.48 (+/- 0.01)	**0.94** (+/- 0.01)	0.58 (+/- 0.01)	0.63 (+/- 0.01)	0.89 (+/- 0.01)	0.67
has	0.84 (+/- 0.01)	**0.94** (+/- 0.00)	0.77 (+/- 0.01)	0.63 (+/- 0.01)	0.91 (+/- 0.01)	0.66
had	0.84 (+/- 0.01)	**0.95** (+/- 0.00)	0.78 (+/- 0.01)	0.62 (+/- 0.01)	0.93 (+/- 0.01)	0.66
was	0.72 (+/- 0.02)	0.86 (+/- 0.01)	0.74 (+/- 0.02)	0.52 (+/- 0.01)	**0.92** (+/- 0.01)	0.53
Average	0.71 (+/- 0.01)	**0.92** (+/- 0.00)	0.71 (+/- 0.01)	0.59 (+/- 0.00)	0.91 (+/- 0.00)	0.61

Table 2: Auxiliary-verb agreeement results. Results are averaged accuracies with standard deviations in brackets.

fastText and especially ELMo in comparison to BERT is the latter's more global training objective that does not rely on sequential input. For ELMo, we also tried running it with full sentence contexts for all auxiliary-verb combinations, which, however, did not lead to improved performance (results omitted).

5.2 Translation Operation

For obtaining an averaged vector offset, we randomly sampled a seed set of verb types from our dataset to learn an offset vector, and subsequently aimed to predict the inflected form for all remaining verb types in the dataset. We sampled 10 different seed sets of size 10 for our experiments[6].

For learning a translation operation with a neural network we used a simple feedforward architecture with a single hidden layer and a tanh activation function, using Adam with a learning rate of 0.01 to optimise the mean squared error between the generated inflected verb and the true inflected verb. Due to the neural network requiring more training data than the averaged vector offset approach, we evaluated the model using 10-fold cross-validation. For APTs we projected the explicit co-occurrence space down to 100 dimensions using SVD before feeding the representations to the neural network.

Performance for both approaches is reported in terms of *Mean Reciprocal Rank (MRR)*, averaged over the 10 randomly sampled seed sets and the 10 cross-validation folds, for the averaged offset vector and neural network approaches, respectively. For calculating MRR, the query space for retrieving an inflected verb, given its lemma and the computed translation operation, is based on all contextualised auxiliary-verb combinations, and all inflected forms of all verbs.

Creating translation operations in embedding space is primarily a word-type level task and thus potentially puts BERT and ELMo at a disadvantage as they produce representations on the token level. This is reflected in Figure 1, where both ELMo and BERT perform poorly in comparison to word2vec and fastText. APTs also exhibit weak performance on this task, with this time the sparsity of its high-dimensional representations being disadvantageous. Interestingly, performance generally dropped — except for word2vec — when moving from the simple vector offset approach to a neural network based translation operation, providing evidence that the morphosyntax of tense and aspect is well represented as a linear offset in the embedding space. One of the main reasons for the poor performance of ELMo

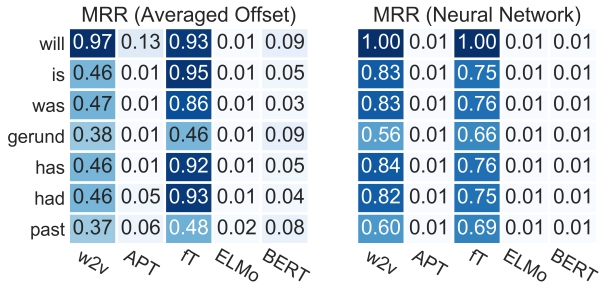

Figure 1: Translation operation results based on averaged MRR.

and BERT was that the obtained offset vectors and learnt translation matrices varied substantially across runs. Figure 2 shows the average cosine similarities (left) and average Euclidean distances (middle) between the computed offset vectors for each subtask across all 10 runs. Figure 2 furthermore shows the average Frobenius distances (right) between the learnt neural network translation matrices across all 10 folds. Figure 2 mirrors the general performance trend in Figure 1, with vector offsets obtained

[6]In preliminary experiments we found that a seed set of 5-10 verbs is sufficient.

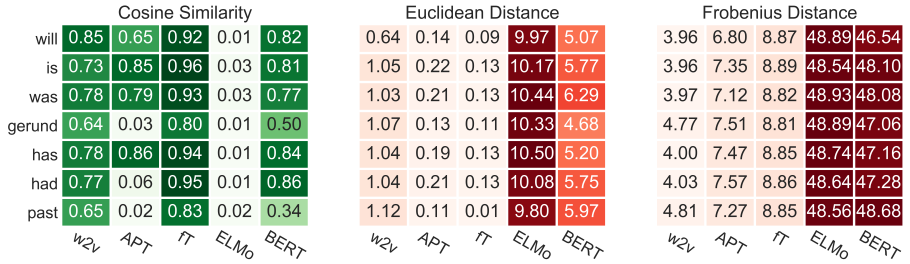

Figure 2: Average cosine similarities and Euclidean distances of averaged offset vectors and Frobenius distances of the learnt neural network weight matrices.

from `word2vec` and `fastText` having high average cosine similarity and low average Euclidean distance. Furthermore, the lower average Frobenius distance for `word2vec` is reflected in its improved performance in comparison to `fastText` whose translation matrices exhibit a larger average Frobenius distance. For ELMo in particular, the offset vectors and translation matrices differ considerably across experimental runs. The large average Frobenius distances for ELMo and BERT also suggest that the neural network struggled to find a good minimum during learning.

5.3 Entailment with Temporal Predications

The results in this section so far have shown that morphosyntactic information relating to tense and aspect is encoded in the different embedding spaces. In the following we use **TEA** to analyse whether these models are able to use that information for natural language inference. As our goal is to assess to what extent tense and aspect are captured by the models, we refrain from fine-tuning them on **TEA**.

For evaluation we measure precision and recall over varying thresholds and report performance in terms of average precision[7]. **TEA** can also serve as an additional evaluation set for sentence encoder models trained on large-scale natural language inference datasets such as SNLI or DNC, which themselves include very little temporal information in their respective test sets. We therefore additionally cast **TEA** as a binary classification task, and report accuracy and macro-averaged F1-score for the two pre-trained biLSTM models.

Table 3 shows the average precision scores for the models and the accuracy and F1-scores for the two pre-trained biLSTMs in comparison to a majority class baseline and a baseline predicting the majority class per tense pair. We used cosine as similarity measure for the embedding models and the softmax prediction scores for the biLSTMs. For APTs, we also tried the asymmetric inclusion score BInc (Szpektor and Dagan, 2008), however found cosine working better. We furthermore experimented with distributional inference (Kober et al., 2016), and found a small positive impact on recall but a slightly larger negative dip in precision, which overall led to slightly lower average precision scores. The results show

Model	Avg. Precision	Accuracy	F1-Score
`word2vec`	0.31	-	-
APT	0.28	-	-
`fastText`	0.30	-	-
ELMo	0.21	-	-
BERT	0.27	-	-
biLSTM-DNC	0.22	0.58	0.49
biLSTM-SNLI	0.21	0.51	0.47
Maj. class	0.22	0.78	0.44
Maj. class / tense pair	**0.35**	**0.80**	**0.66**

Table 3: **TEA** results. All model results are significantly worse at the $p < 0.01$ level w.r.t. the majority class / tense pair baseline, using a randomised bootstrap test (Efron and Tibshirani, 1994).

that neither of the models are able to outperform the majority class / tense baseline. This highlights that despite the use of short and simple sentences in the dataset, the latent nature of tense and aspect make **TEA** a very challenging problem.

[7] Also known as the area under the precision-recall curve.

In order to analyse the causes for the low performance across models, we calculated the false positive and false negative rates for different similarity threshold ranges for each of the models. Figure 3 shows that even for high thresholds, the neural embedding models frequently predict entailment when there isn't one, thereby producing a high rate of false positives (highlighted at the top of Figure 3). Conversely, a sparse model such as APTs, fails to predict entailment when there actually is one, resulting in a high rate of false negatives (highlighted at the bottom of Figure 3). Our results show that natural language

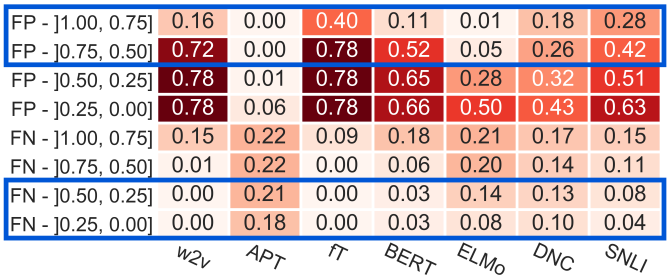

Figure 3: False Positive (FP) and False Negative (FN) rates.

inference on temporal predications is a challenging problem, especially for distributional semantic approaches. One reason is that these models are primarily governed by contextual similarity which is a bad proxy for inference in the case of a dataset such as **TEA**. For example, if *Jane has arrived in London*, then she *was going to London* at some earlier point, but it is not the case that she currently *is going to London*. Furthermore, when she *has arrived in London*, she *is visiting London* at the moment, and *will leave* again at some point in the future.

The predications in the short narrative above are very diverse in terms of tense and aspect, however the main verbs — or even the predications as a whole — typically have high distributional similarity, which inevitably leads to numerous false entailment decisions as reflected in Figure 3.

In the following we briefly analyse the impact of distributional similarity and investigate to what extent the similarity scores between two predications change when tense and aspect influence the entailment. Table 4 shows that the cosine similarity between temporally and aspectually modified predications is typically higher than for their respective lemmas. This further indicates that many false positives of the neural network based models in our results are due to high distributional similarity scores between predications. For APTs the cosine scores — even when normalised — are generally very low due to their sparsity and high dimensionality, highlighting their bias towards false negatives. However, Table 4

Predication Pair	**w2v**	**APT**	**fT**	**ELMo**	**BERT**	**DNC**	**SNLI**
visit $\models$ leave	0.36	0.09	0.53	0.59	0.69	0.69	0.28
is visiting $\models$ will leave	0.57	0.02	0.60	0.60	**0.77**	0.26	**0.26**
is visiting $\not\models$ has left	0.58	0.03	0.71	0.65	0.72	0.32	0.20
visit $\models$ arrive	0.45	0.07	0.55	0.49	0.71	0.58	0.45
is visiting $\models$ has arrived	**0.62**	**0.04**	**0.69**	**0.51**	**0.84**	0.25	**0.51**
is visiting $\not\models$ will arrive	0.57	0.01	0.60	0.50	0.81	0.32	0.25
win $\models$ play	0.52	0.14	0.54	0.59	0.73	0.39	0.32
has won $\models$ has played	**0.75**	**0.25**	**0.88**	**0.60**	**0.85**	**0.55**	0.23
has won $\not\models$ will play	0.60	0.11	0.64	0.55	0.78	0.31	0.36

Table 4: Similarity scores between the example predicates. DNC and SNLI refer to the two biLSTMs pre-trained on DNC and SNLI, respectively.

also shows that in most cases the distributional similarity between an entailed pair is higher than for a non-entailed pair (boldfaced in Table 4). This indicates that the embedding models do appear to capture *some* of the semantics of tense and aspect in their respective contextualised representations. However, their high distributional similarity overwhelms any finer distinction that the models might have extracted.

While our analysis indicates that the embedding models are able to extract knowledge about tense and aspect, the signal is not strong enough to reliably perform inference. A potential avenue for future work would therefore be the development of models that are able to better represent tense and aspect, while not being primarily governed by distributional similarity.

6 Related Work

Most previous work on inference between verbs was concerned with extracting inference rules from raw text (Lin and Pantel, 2001; Szpektor et al., 2004, 2007; Hashimoto et al., 2009; Melamud et al., 2013). As a next step, Berant et al. (2010) and Hosseini et al. (2018) leverage these rules to build entailment graphs for modelling natural language inference. However in both cases the entailment graphs are built on the basis of *verb lemmas* and do not take tense and aspect into account. One example of using tense for inference is Pavlick and Callison-Burch (2016), who leverage implicative verbs to determine that *managed to solve X* $\models$ *X is solved*. Our proposed dataset **TEA** fills a gap in the natural language inference evaluation repertoire by focusing on temporal and aspectual entailment. Recent years saw the release of a number of large-scale datasets, such as SNLI (Bowman et al., 2015), MNLI (Williams et al., 2017) or DNC (Poliak et al., 2018), but neither of these datasets focuses on, or includes a substantial proportion of, inference examples between temporal predications.

TEA is related to work on causality (Mirza et al., 2014; Mirza and Tonelli, 2014), however our dataset has been created from scratch rather than derived from TimeBank (Pustejovsky et al., 2003), as for example explicit *buys* $\models$ *owns* relations are rarely encountered in the same paragraph or connected by explicit causal links. Therefore, **TEA** captures many consequent state inferences that are missing from previous datasets. The most closely related task to **TEA** is the relation inference dataset of Levy and Dagan (2016), which however, contains only very few examples where temporality is a governing factor.

7 Future Work

In future work we plan to leverage tense- and aspect-based information for constructing temporal entailment graphs (Lewis and Steedman, 2014), where nodes represent tensed predicates (e.g. *has visited*), and edges represent entailment relations. Temporal entailment graphs, together with knowledge about the *completedness* or *current relevance* of an event, can be applied to procedural reasoning, such as tracking the state of entities through text, similar to recent work of Bosselut et al. (2017), and Henaff et al. (2017). We furthermore plan to focus on other types of aspect such as *Aktionsart*.

8 Conclusion

In this paper we highlighted that tense and aspect are two of the most important factors for performing natural language inference. We introduced a novel entailment dataset, **TEA**, that contains pairs of short sentences and focuses on entailment relations between temporally and aspectually modified verbs. We showed that distributional embedding models capture a considerable amount of the morphosyntactic information relating to tense and aspect in their embedding spaces. However, neither the embedding models, nor two pre-trained biLSTMs, were able to outperform a simple rule-based baseline on **TEA**, primarily due to their reliance on contextual similarity for inference. In this sense, tense and aspect semantically resemble logical operators like negation rather than distributional components. The challenge will be to combine logical operator semantics with distributional representations of content words.

Acknowledgements

We thank Javad Hosseini, Paola Merlo and Nate Chambers for valuable discussions and comments on this work. We also thank our anonymous reviewers for their helpful feedback which led to a substantially improved paper. This research was supported in part by ERC Advanced Fellowship GA 742137 SEMANTAX, a Google faculty award, a Bloomberg L.P. Gift award, and a University of Edinburgh/Huawei Technologies award to Mark Steedman.

References

Balazs, J., E. Marrese-Taylor, P. Loyola, and Y. Matsuo (2017). Refining raw sentence representations for textual entailment recognition via attention. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pp. 51–55. Association for Computational Linguistics.

Berant, J., I. Dagan, and J. Goldberger (2010, July). Global learning of focused entailment graphs. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, Uppsala, Sweden, pp. 1220–1229. Association for Computational Linguistics.

Bojanowski, P., E. Grave, A. Joulin, and T. Mikolov (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics 5*, 135–146.

Bolukbasi, T., K.-W. Chang, J. Zou, V. Saligrama, and A. Kalai (2016). Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, USA, pp. 4356–4364. Curran Associates Inc.

Bosselut, A., O. Levy, A. Holtzman, C. Ennis, D. Fox, and Y. Choi (2017). Simulating action dynamics with neural process networks. In *In Proceedings of ICLR*.

Bowman, S. R., G. Angeli, C. Potts, and C. D. Manning (2015, September). A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, pp. 632–642. Association for Computational Linguistics.

Chelba, C., T. Mikolov, M. Schuster, Q. Ge, T. Brants, P. Koehn, and T. Robinson (2013). One billion word benchmark for measuring progress in statistical language modeling. Technical report, Google.

Chklovski, T. and P. Pantel (2004, July). Verbocean: Mining the web for fine-grained semantic verb relations. In D. Lin and D. Wu (Eds.), *Proceedings of EMNLP 2004*, Barcelona, Spain, pp. 33–40. Association for Computational Linguistics.

Comrie, B. (1985). *Tense*. Cambridge Textbooks in Linguistics. Cambridge University Press.

Conneau, A., D. Kiela, H. Schwenk, L. Barrault, and A. Bordes (2017, September). Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark, pp. 681–691. Association for Computational Linguistics.

Dagan, I., O. Glickman, and B. Magnini (2006). The pascal recognising textual entailment challenge. In J. Quiñonero-Candela, I. Dagan, B. Magnini, and F. d'Alché Buc (Eds.), *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, Berlin, Heidelberg, pp. 177–190. Springer Berlin Heidelberg.

Dahl, Ö. (1985). *Tense and Aspect Systems*. Blackwell Publishing Ltd.

Depraetere, I. (1998). On the resultative character of present perfect sentences. *Journal of Pragmatics 29*(5), 597 – 613.

Devlin, J., M.-W. Chang, K. Lee, and K. Toutanova (2018, October). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *ArXiv e-prints*.

Dowty, D. (1979). *Word Meaning and Montague Grammar*. Holland: Dordrecht.

Efron, B. and R. Tibshirani (1994). *An Introduction to the Bootstrap*. CRC press.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Bradford Books.

Gardner, M., J. Grus, M. Neumann, O. Tafjord, P. Dasigi, N. F. Liu, M. Peters, M. Schmitz, and L. S. Zettlemoyer (2017). Allennlp: A deep semantic natural language processing platform.

Hashimoto, C., K. Torisawa, K. Kuroda, S. De Saeger, M. Murata, and J. Kazama (2009, August). Large-scale verb entailment acquisition from the Web. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, Singapore, pp. 1172–1181. Association for Computational Linguistics.

Henaff, M., J. Weston, A. Szlam, A. Bordes, and Y. LeCun (2017). Tracking the world state with recurrent entity networks. In *In Proceedings of ICLR*.

Hochreiter, S. and J. Schmidhuber (1997, nov). Long short-term memory. *Neural Computation 9*(8), 1735–1780.

Hosseini, M. J., N. Chambers, S. Reddy, X. R. Holt, S. B. Cohen, M. Johnson, and M. Steedman (2018). Learning typed entailment graphs with global soft constraints. *Transactions of the Association for Computational Linguistics 6*, 703–717.

Iyyer, M., V. Manjunatha, J. Boyd-Graber, and H. Daumé III (2015, July). Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Beijing, China, pp. 1681–1691. Association for Computational Linguistics.

Katz, G. (2003). On the stativity of the english perfect. In *Perfect explorations*. de Gruyter, The Hague.

Kingma, D. P. and J. Ba (2014). Adam: A method for stochastic optimization. *CoRR abs/1412.6980*.

Kober, T. (2018). *Inferring Unobserved Co-occurrence Events in Anchored Packed Trees*. Ph. D. thesis, University of Sussex.

Kober, T., J. Weeds, J. Reffin, and D. Weir (2016, November). Improving sparse word representations with distributional inference for semantic composition. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas, pp. 1691–1702.

Kober, T., J. Weeds, J. Reffin, and D. Weir (2017, July). Improving semantic composition with offset inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Vancouver, Canada, pp. 433–440. Association for Computational Linguistics.

Kober, T., J. Weeds, J. Wilkie, J. Reffin, and D. Weir (2017, April). One representation per word - does it make sense for composition? In *Proceedings of the 1st Workshop on Sense, Concept and Entity Representations and their Applications*, Valencia, Spain, pp. 79–90.

Levy, O. and I. Dagan (2016, August). Annotating relation inference in context via question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Berlin, Germany, pp. 249–255. Association for Computational Linguistics.

Lewis, M. and M. Steedman (2014, June). Combining formal and distributional models of temporal and intensional semantics. In *Proceedings of the ACL 2014 Workshop on Semantic Parsing*, Baltimore, MD, pp. 28–32. Association for Computational Linguistics.

Lin, D. and P. Pantel (2001, aug 26–29). DIRT — discovery of inference rules from text. In F. Provost and R. Srikant (Eds.), *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-01)*, New York, pp. 323–328. ACM Press.

Linzen, T., E. Dupoux, and Y. Goldberg (2016). Assessing the ability of lstms to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics 4*, 521–535.

Melamud, O., J. Berant, I. Dagan, J. Goldberger, and I. Szpektor (2013, August). A two level model for context sensitive inference rules. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Sofia, Bulgaria, pp. 1331–1340. Association for Computational Linguistics.

Mikolov, T., E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin (2018). Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, and J. Dean (2013). Distributed representations of words and phrases and their compositionality. In C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger (Eds.), *Advances in Neural Information Processing Systems 26*, pp. 3111–3119. Curran Associates, Inc.

Mirza, P., R. Sprugnoli, S. Tonelli, and M. Speranza (2014). Annotating causality in the tempeval-3 corpus. In *Proceedings of the EACL 2014 Workshop on Computational Approaches to Causality in Language (CAtoCL)*, pp. 10–19. Association for Computational Linguistics.

Mirza, P. and S. Tonelli (2014). An analysis of causality between events and its relation to temporal information. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pp. 2097–2106. Dublin City University and Association for Computational Linguistics.

Mitchell, J. and M. Steedman (2015). Orthogonality of syntax and semantics within distributional spaces. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1301–1310. Association for Computational Linguistics.

Moens, M. and M. Steedman (1988). Temporal ontology and temporal reference. *Computational Linguistics 14*(2), 15–28.

Moss, H., D. Leslie, and P. Rayson (2018). Using j-k-fold cross validation to reduce variance when tuning nlp models. In *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 2978–2989. Association for Computational Linguistics.

Paszke, A., S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer (2017). Automatic differentiation in pytorch. In *NIPS-W*.

Pavlick, E. and C. Callison-Burch (2016, November). Tense manages to predict implicative behavior in verbs. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas, pp. 2225–2229. Association for Computational Linguistics.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay (2011, November). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research 12*, 2825–2830.

Peters, M., M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer (2018). Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 2227–2237. Association for Computational Linguistics.

Poliak, A., A. Haldar, R. Rudinger, J. E. Hu, E. Pavlick, A. S. White, and B. Van Durme (2018). Collecting diverse natural language inference problems for sentence representation evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 67–81. Association for Computational Linguistics.

Pustejovsky, J., P. Hanks, R. Sauri, A. See, R. Gaizauskas, A. Setzer, D. Radev, B. Sundheim, D. Day, L. Ferro, and M. Lazo (2003, March). The TIMEBANK corpus. In *Proceedings of Corpus Linguistics 2003*, Lancaster, pp. 647–656.

Reichenbach, H. (1947). *Elements of Symbolic Logic*. London: Macmillan.

Shoemark, P., J. Kirby, and S. Goldwater (2017). Topic and audience effects on distinctively scottish vocabulary usage in twitter data. In *Proceedings of the Workshop on Stylistic Variation*, pp. 59–68. Association for Computational Linguistics.

Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research 15*, 1929–1958.

Steedman, M. (1997). Temporality. In J. van Benthem and A. ter Meulen (Eds.), *Handbook of Logic and Language*, pp. 895–938. Amsterdam: North-Holland.

Szpektor, I. and I. Dagan (2008, August). Learning entailment rules for unary templates. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, Manchester, UK, pp. 849–856. Coling 2008 Organizing Committee.

Szpektor, I., E. Shnarch, and I. Dagan (2007, June). Instance-based evaluation of entailment rule acquisition. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, Prague, Czech Republic, pp. 456–463. Association for Computational Linguistics.

Szpektor, I., H. Tanev, I. Dagan, and B. Coppola (2004, July). Scaling web-based acquisition of entailment relations. In D. Lin and D. Wu (Eds.), *Proceedings of EMNLP 2004*, Barcelona, Spain, pp. 41–48. Association for Computational Linguistics.

Tian, R., N. Okazaki, and K. Inui (2017). The mechanism of additive composition. *Machine Learning 106*(7), 1083–1130.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), *Advances in Neural Information Processing Systems 30*, pp. 5998–6008. Curran Associates, Inc.

Vendler, Z. (1957). Verbs and times. *Linguistics in Philosophy*, 97–121.

Vulić, I., D. Gerz, D. Kiela, F. Hill, and A. Korhonen (2017). Hyperlex: A large-scale evaluation of graded lexical entailment. *Computational Linguistics 43*(4), 781–835.

Vulić, I. and N. 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)*, pp. 1134–1145. Association for Computational Linguistics.

Vylomova, E., L. Rimell, T. Cohn, and T. Baldwin (2016). Take and took, gaggle and goose, book and read: Evaluating the utility of vector differences for lexical relation learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1671–1682. Association for Computational Linguistics.

Weir, D., J. Weeds, J. Reffin, and T. Kober (2016, December). Aligning packed dependency trees: a theory of composition for distributional semantics. *Computational Linguistics, special issue on Formal Distributional Semantics 42*(4), 727–761.

Weisman, H., J. Berant, I. Szpektor, and I. Dagan (2012). Learning verb inference rules from linguistically-motivated evidence. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 194–204. Association for Computational Linguistics.

Wieting, J., M. Bansal, K. Gimpel, and K. Livescu (2016). Towards universal paraphrastic sentence embeddings. In *Proceedings of the International Conference on Learning Representations*.

Williams, A., N. Nangia, and S. R. Bowman (2017). A broad-coverage challenge corpus for sentence understanding through inference. *ArXiv e-prints*.

A Supplemental Material

A.1 Dataset Details

Table 5 shows a detailed overview of the number of examples per tense and aspect pair, as well as their class distribution.

Category	**Num. Examples**	**Class distribution** (*entailment : non-entailment*)
Present progressive - Present progressive	188	33 : 67
Present progressive - Past progressive	188	23 : 77
Present progressive - Present perfect	213	20 : 80
Present progressive - Past perfect	213	12 : 88
Present progressive - Future simple	216	28 : 72
Present progressive - Present simple	216	27 : 73
Present progressive - Past simple	216	26 : 74
Past progressive - Present progressive	188	0 : 100
Past progressive - Past progressive	188	55 : 45
Past progressive - Present perfect	213	7 : 93
Past progressive - Past perfect	213	46 : 54
Past progressive - Future simple	216	1 : 99
Past progressive - Present simple	216	0 : 100
Past progressive - Past simple	216	49 : 51
Present perfect - Present progressive	213	12 : 88
Present perfect - Past progressive	213	44 : 56
Present perfect - Present perfect	240	26 : 74
Present perfect - Past perfect	240	26 : 74
Present perfect - Future simple	243	16 : 84
Present perfect - Present simple	243	17 : 83
Present perfect - Past simple	243	42 : 58
Past perfect - Present progressive	213	0 : 100
Past perfect - Past progressive	213	58 : 42
Past perfect - Present perfect	240	3 : 97
Past perfect - Past perfect	240	59 : 41
Past perfect - Future simple	243	0 : 100
Past perfect - Present simple	243	0 : 100
Past perfect - Past simple	243	58 : 42
Future simple - Present progressive	216	3 : 97
Future simple - Past progressive	216	1 : 99
Future simple - Present perfect	243	1 : 99
Future simple - Past perfect	243	1 : 99
Future simple - Future simple	246	47 : 53
Future simple - Present simple	246	2 : 98
Future simple - Past simple	246	1 : 99
Present simple - Present progressive	216	21 : 79
Present simple - Past progressive	216	29 : 71
Present simple - Present perfect	243	15 : 85
Present simple - Past perfect	243	17 : 83
Present simple - Future simple	246	19 : 81
Present simple - Present simple	246	29 : 71
Present simple - Past simple	246	26 : 74
Past simple - Present progressive	216	0 : 100
Past simple - Past progressive	216	55 : 45
Past simple - Present perfect	243	5 : 95
Past simple - Past perfect	243	54 : 46
Past simple - Future simple	246	0 : 100
Past simple - Present simple	246	1 : 99
Past simple - Past simple	246	56 : 44
Progressive - Progressive	3464	20 : 80
Progressive - Perfect	2748	18 : 82
Perfect - Progressive	2748	16 : 84
Perfect - Perfect	2178	37 : 63
TOTAL	11138	22 : 78

Table 5: Detailed statistics of **TEA**.

B Supplemental Material

B.1 Model Details

word2vec. We used the 300-dimensional vectors trained on GoogleNews, available from `https://code.google.com/archive/p/word2vec/`.

APTs. We used order 2 APTs trained on Gigaword, with PPMI weighting and no negative SPPMI shift which were used in Kober (2018). As composition function we used *composition by intersection* which has previously been shown to work well for modelling the similarity of short phrases (Kober et al., 2016, 2017).

fastText. We used the 300-dimensional pre-trained vectors with subword information trained on Wikipedia (Mikolov et al., 2018).

ELMo. We are using the pre-trained model released by Peters et al. (2018) and accessible via the AllenNLP toolkit (Gardner et al., 2017).

BERT. We are using the BERT-big model released by Devlin et al. (2018) and available from `https://github.com/google-research/bert`.

Pre-trained biLSTM. We are using a bi-directional LSTM (Hochreiter and Schmidhuber, 1997) with max pooling, but without an attention layer. We follow Balazs et al. (2017) in aggregating the embedded and pooled premise and hypothesis representations before passing them to a single fully connected layer, with a relu activation function and a dropout (Srivastava et al., 2014) probability of 0.3. The model is optimised with Adam (Kingma and Ba, 2014) using a learning rate of 0.01. The model is implemented in PyTorch (Paszke et al., 2017). Table 6 lists the accuracies on the SNLI and DNC development and test sets for our model.

Dataset	Dev Accuracy	Test Accuracy
SNLI	0.83	0.82
DNC	0.88	0.87

Table 6: Accuracies on the development and test sets for the pre-trained biLSTMs on SNLI and DNC, respectively.