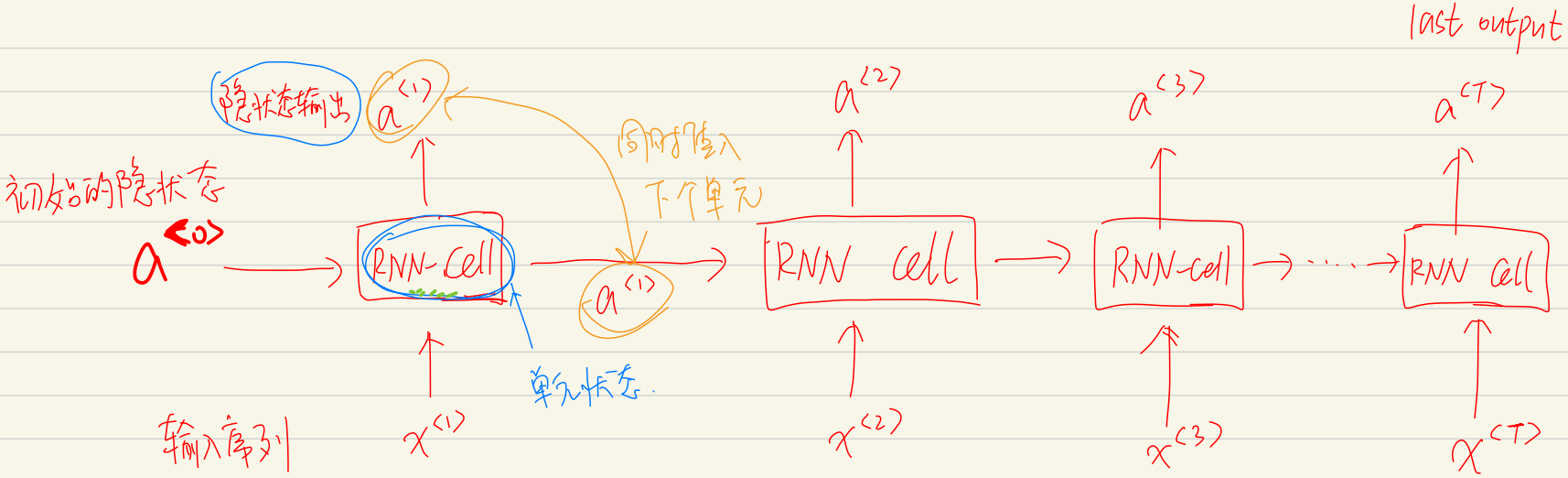



return sequences：返回的是隐藏状态输出 $a^{(t)}$，默认设置 = False，只输出 $a^{(T)}$

设置 = True，会输出所有 $a^{(t)}$

return states：返回的是 LSTM 单元状态.

train_x 的维度 (n_samples, time_steps, input_dim)

input_dim

output_dim

input_length

time_steps

(32, 50, 300)

32 — batch_size

50 — 最长的句子

300 — 词向量长度

模型的基本框架：对 NLI 中的两个句子分别建模，生成 sentence embedding，然后输入到 神经网络分类器中。（即且 sentence embedding 可以用于迁移学习）

# Natural Language Inference with Hierarchical BiLSTM Max Pooling Architecture

**Aarne Talman, Anssi Yli-Jyrä and Jörg Tiedemann**
University of Helsinki
{aarne.talman, anssi.yli-jyra, jorg.tiedemann}@helsinki.fi

arXiv:1808.08762v1 [cs.CL] 27 Aug 2018

## Abstract

Recurrent neural networks have proven to be very effective for natural language inference tasks. We build on top of one such model, namely BiLSTM with max pooling, and show that adding a hierarchy of BiLSTM and max pooling layers yields state of the art results for the SNLI sentence encoding-based models and the SciTail dataset, as well as provides strong results for the MultiNLI dataset. We also show that our sentence embeddings can be utilized in a wide variety of transfer learning tasks, outperforming InferSent on 7 out of 10 and SkipThought on 8 out of 9 SentEval sentence embedding evaluation tasks. Furthermore, our model beats the InferSent model in 8 out of 10 recently published SentEval probing tasks designed to evaluate sentence embeddings' ability to capture some of the important linguistic properties of sentences.

## 1 Introduction

Neural networks have been shown to provide a powerful tool for building representations of natural language on multiple levels of abstraction. Perhaps the most widely used representations in natural language processing are word embeddings (e.g. Bengio et al., 2003; Mikolov et al., 2013; Pennington et al., 2014). Recently there has been a growing interest in models for sentence-level representations using neural networks. Sentence embeddings are distributed representations of natural language sentences with the intention to encode the meaning of the sentences in a neural network representation. Sentence embeddings have been generated using unsupervised learning approaches (e.g. Kiros et al., 2015; Hill et al., 2016), and supervised learning (e.g. Bowman et al., 2016; Conneau et al., 2017).

Sentence-level representations have shown promise in multiple different NLP tasks. One prominent example is natural language inference. Natural language inference (NLI) is the task of determining the inferential relationship between two or more sentences. That is, given two sentences, the premise $p$ and the hypothesis $h$, the task is to determine whether $h$ is entailed by $p$, whether the sentences are in contradiction with each other or whether they are neutral. There are two main approaches to NLI utilizing neural networks. Some approaches focus on building sentence embeddings for the premises and the hypothesis separately and then combine those using a classifier (e.g. Bowman et al., 2015, 2016; Conneau et al., 2017). Other approaches do not treat the two sentences separately but utilize e.g. cross-sentence attention (Tay et al., 2018; Chen et al., 2017a).

用神经网络处理 NLI 任务的两种方式：①. 分别建立两个句子的 embedding，然后输到分类器中。②. 利用两个句子之间的关系。

In this paper we focus on the sentence embedding approach. Motivated by the success of the architecture of InferSent (Conneau et al., 2017), we build a hierarchical architecture utilizing bidirectional LSTM (BiLSTM) layers and max pooling. All in all, our model in on par with the state of the art for Stanford Natural Language Inference corpus (SNLI) (Bowman et al., 2015) sentence encoding-based models and improves the previous state of the art for SciTail (Khot et al., 2018). We also achieve strong results for the Multi-Genre Natural Language Inference corpus (MultiNLI) (Williams et al., 2018).

We also test our model on a number of transfer learning tasks using the SentEval testing library (Conneau et al., 2017), and show that our model outperforms the InferSent model on 7 out of 10 and SkipThought on 8 out of 9 tasks, comparing to the scores reported by Conneau et al. (2017). Moreover, our model outperforms the InferSent model in 8 out of 10 recently published SentEval probing tasks designed to evaluate sentence embeddings' ability to capture some of the important

实验评测和对比模型

linguistic properties of sentences (Conneau et al., 2018). This highlights the generalization capability of our proposed model, confirming that the proposed architecture is able to produce sentence embeddings with strong performance across a wide variety of different NLP tasks.

## 2 Related Work

Sentence embeddings have been utilized in a wide variety of approaches to natural language inference. Bowman et al. (2015, 2016) explore RNN and LSTM architectures, Mou et al. (2016) convolutional neural networks and Vendrov et al. (2015) GRUs, to name a few. The basic idea behind these approaches is to encode the premise and hypothesis sentences separately and then combine those using a neural network classifier.

Conneau et al. (2017) explore multiple different sentence embedding architectures ranging from LSTM, BiLSTM and intra-attention to convolution neural networks and the performance of these architectures on NLI tasks. They show that out of these models BiLSTM with max pooling achieves the strongest results in NLI. They also show that their model trained on NLI data achieves strong performance on various transfer learning tasks.

Although sentence embedding approaches have shown their effectiveness in NLI, there are multiple studies showing that treating the hypothesis and premise sentences together and focusing on the relationship between those sentences yields better results (e.g. Tay et al., 2018; Chen et al., 2017a). However, as these methods are focused on the inference relations rather than the internal semantics of the sentences, they cannot as straightforwardly be used outside of the NLI context and do not offer similar insights about the sentence level semantics, as sentence embeddings do. By choosing a sentence embedding-based architecture we can more easily use the models in a wide variety of NLP tasks requiring sentence-level semantic information.

## 3 Model Architecture

Our proposed architecture follows a sentence embedding-based approach for NLI introduced by Bowman et al. (2015). The model illustrated in Figure 1 contains sentence embeddings for the two input sentences, where the output of the sentence embeddings are combined using a heuristic introduced by (Mou et al., 2015), putting together the concatenation $(u, v)$, absolute element-wise difference $|u - v|$, and element-wise product $u * v$. The combined vector is then passed on to a 3-layered multi-layer perceptron (MLP) with a 3-way softmax classifier. The first two layers of the MLP both utilize dropout and a ReLU activation function. We use a variant of ReLU called Leaky ReLU (Maas et al., 2013), defined by:

$$LeakyReLU(x) = \max(0, x) + y * \min(0, x)$$

where we set $y = 0.01$ as the negative slope for $x < 0$. This prevents the gradient from dying when $x < 0$.

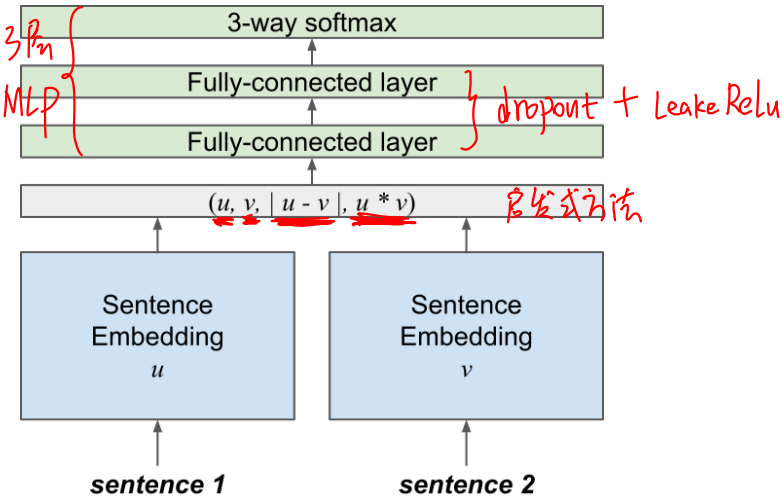



Figure 1: Overall NLI Architecture

For the sentence representations we first embed the individual words with pre-trained word embedding. The sequence of the embedded words is then passed on to the sentence encoder which utilizes BiLSTM with max pooling.

Given a sequence $T$ of words $(w_1 \ldots, w_T)$, the output of the bi-directional LSTM is a set of vectors $(h_1, \ldots, h_T)$, where each $h_t \in (h_1, \ldots, h_T)$ is the concatenation

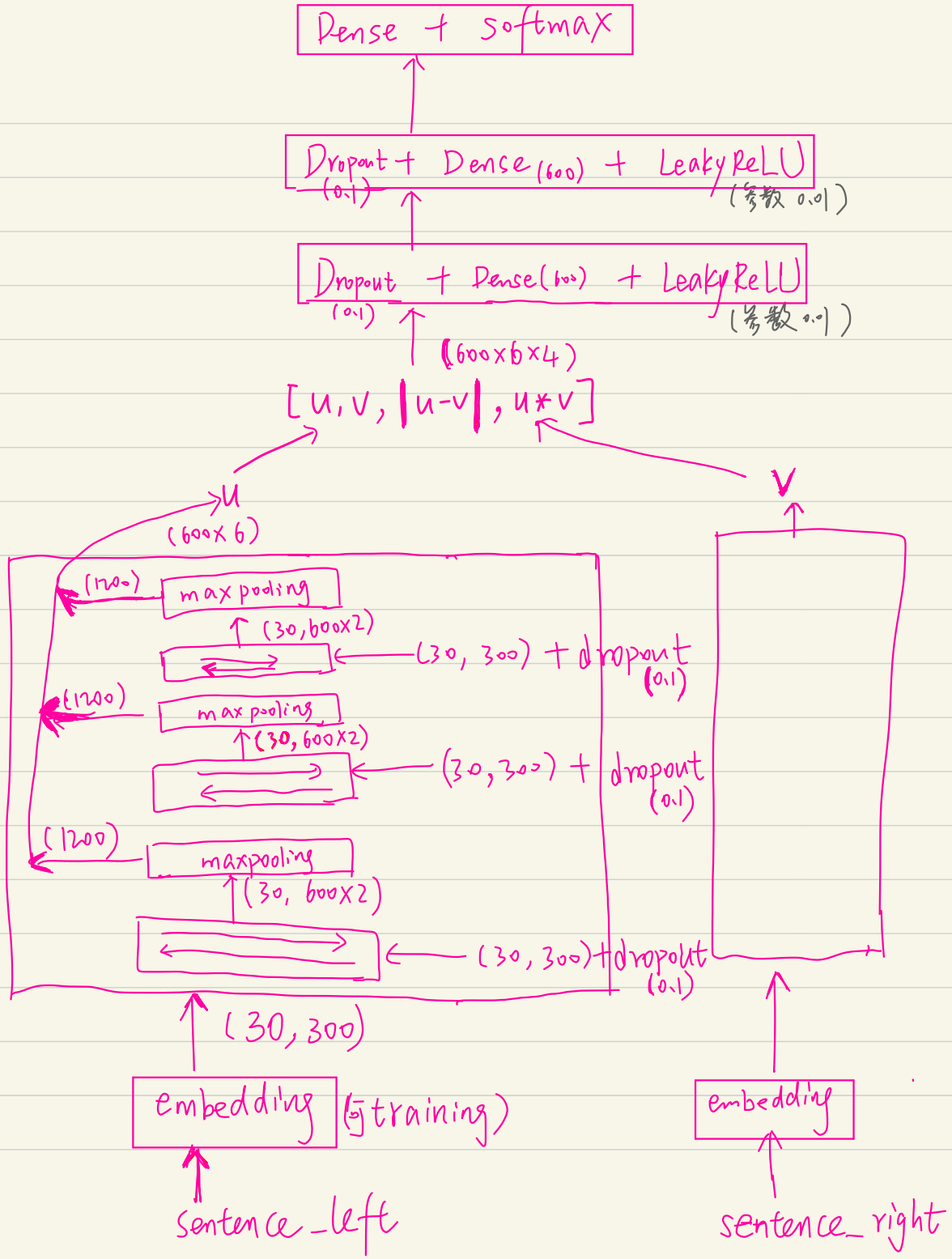
$$h_t = [\overrightarrow{h}_t, \overleftarrow{h}_t]$$

of a forward and backward LSTMs

$$\overrightarrow{h}_t = \overrightarrow{LSTM}_t(w_1, \ldots, w_T)$$

$$\overleftarrow{h}_t = \overleftarrow{LSTM}_t(w_1, \ldots, w_T).$$

The max pooling layer produces a vector of the same dimensionality as $h_t$, returning, for each dimension, its maximum value over the hidden units $(h_1, \ldots, h_T)$.


Dense + softmax
Dropout + Dense(600) + LeakyReLU
(0.1)
(参数 0.01)
Dropout + Dense(600) + LeakyReLU
(0.1)
(参数 0.01)
(600x6x4)
[u, v, |u-v|, u*v]
u
(600x6)
v
(1200)
max pooling
(30, 600x2)
(30, 300) + dropout
(0.1)
(1200)
max pooling
(30, 600x2)
(30, 300) + dropout
(0.1)
(1200)
maxpooling
(30, 600x2)
(30, 300) + dropout
(0.1)
(30, 300)
embedding
(与training)
embedding
sentence_left
sentence_right

训练细节：

① 优化：Adam

② 学习率：5e-4 在哪设置 ??

每个 epoch 后如果模型没有提升，则 lr 下降因子 0.2

③ batch-size：64

④ 每次 epoch 后，在 dev set 上评估，

如果超过 3 个 epoch 的 loss 都在上升，则训练停止，

选在 dev set 上 acc 最高的 model 用于测试集

⑤ 使用预训练好的 Glove word embedding

300 维

Motivated by the strong results of the BiLSTM max pooling network by Conneau et al. (2017), we experimented with combining BiLSTM max pooling networks as a hierarchical structure.[1] To improve the BiLSTM layers' ability to remember the input words, we let each layer of the stack re-read the input sentence. In our baseline model we stack three BiLSTM max pooling networks as a hierarchical structure, where each BiLSTM reads the input sentence as the input. At each BiLSTM layer except the first one, we initialize the initial hidden state and the cell state with the final state of the previous layer. We take the max value over each dimension of the hidden units for each BiLSTM layer. The final output of the sentence embedding is the concatenation of each of these max pooling layers. Our sentence embedding architecture is visualized in Figure 2.

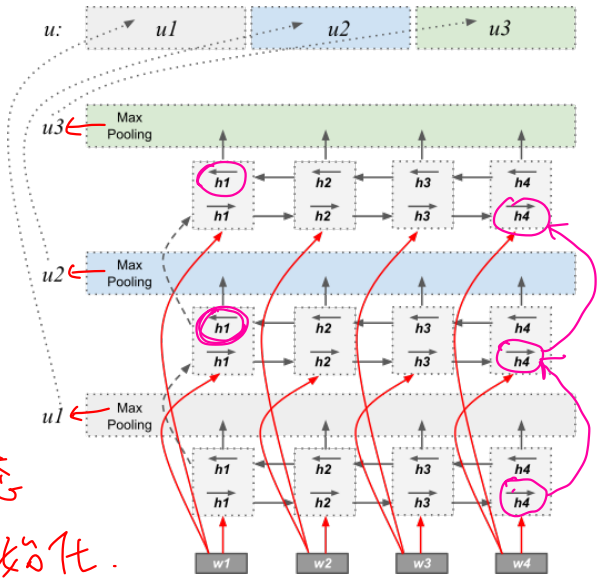



Figure 2: Architecture of the sentence embeddings: Hierarchical BiLSTM max pooling (HBMP) architecture (where $T = 4$).

## 4 Experimental Setup

### 4.1 Data

We evaluated the sentence embedding architecture with three different natural language inference datasets, including the Stanford Natural Language Inference (SNLI) corpus, the Multi-Genre Natural Language Inference (MultiNLI) corpus and the SciTail dataset. In all our experiments with the three datasets we used only the training data provided in the respective corpus. For the transfer learning tasks, described in Section 7, we used training data from both the SNLI and the MultiNLI datasets in order to compare to the results by Conneau et al. (2017).

**SNLI**

The Stanford Natural Language Inference (SNLI) corpus (Bowman et al., 2015) is a dataset of 570k human-written sentence pairs manually labeled with entailment, contradiction, and neutral. The dataset is divided into training (550,152 pairs), development (10,000 pairs) and test sets (10,000 pairs). The source for the premise sentences in SNLI were image captions taken from the Flickr30k corpus (Young et al., 2014).

**MultiNLI**

The Multi-Genre Natural Language Inference (MultiNLI) corpus (Williams et al., 2018) is a broad-coverage corpus for natural language inference, consisting of 433k human-written sentence pairs labeled with entailment, contradiction and neutral. Unlike the SNLI corpus, which draws the premise sentence from image captions, MultiNLI consists of sentence pairs from ten distinct genres of both written and spoken English. The dataset is divided into training (392,702 pairs), development (20,000 pairs) and test sets (20,000 pairs).

All of the genres are included in the test and development sets, but only five are included in the training set. The development and test datasets have been divided into matched and mismatched, where the former includes only sentences from the same genres as the training data, and the latter includes sentences from the remaining genres not present in the training data.

In addition to the training, development and test sets, MultiNLI provides a smaller annotation dataset, which contains approximately 1000 sentence pairs annotated with linguistic properties of the sentences and is split between the matched and mismatched datasets.[2] This annotation dataset provides a simple way to assess what kind of sentence pairs an NLI system is able to predict correctly and where it makes errors.

[1] Conneau et al. (2017) explore a similar hierarchical architecture of convolutional neural networks.

[2] The annotated dataset and description of the annotations are available at http://www.nyu.edu/projects/bowman/multinli/multinli_1.0_annotations.zip

### SciTail

The SciTail dataset (Khot et al., 2018) is an NLI dataset created from multiple-choice science exams consisting of 27k sentence pairs. Each question and the correct answer choice have been converted into an assertive statement to form the hypothesis. The dataset is divided into training (23,596 pairs), development (1,304 pairs) and test sets (2,126 pairs). Unlike the SNLI and MultiNLI datasets, SciTail uses only two labels: *entailment* and *neutral*.

## 4.2 Training Details

The architecture was implemented using PyTorch. We have published our code in GitHub: `https://github.com/Helsinki-NLP/HBMP`.

For all of our models we used a gradient descent optimization algorithm based on the Adam update rule (Kingma and Ba, 2014), which is pre-implemented in PyTorch. We used a learning rate of 5e-4 for all our models. The learning rate was decreased by the factor of 0.2 after each epoch if the model did not improve. We used a batch size of 64. The models were evaluated with the development data after each epoch and training was stopped if the development loss increased for more than 3 epochs. The model with the highest development accuracy was selected for testing.

We use pre-trained GloVe word embeddings of size 300 dimensions (GloVe 840B 300D) (Pennington et al., 2014), which were fine-tuned during training. The sentence embeddings have hidden dimensionality of 600 for both direction (except for SentEval test, where we test models with 600D and 1200D per direction) and the 3-layer multi-layer perceptron (MLP) have the size of 600 dimensions. We use a dropout of 0.1 in the BiLSTM layers and between the MLP layers (except just before the final layer).

All our models were trained using one NVIDIA Tesla P100 GPU.

# 5 Experimental Results

The proposed architecture provides strong results on all of the three datasets. We achieve the new state of the art for SciTail 86.0%. On SNLI our results are on par with the current state of the art 86.6%. On MultiNLI our model provides strong results on both the matched and mismatched test sets.

### SNLI

For for the SNLI corpus our model provides the test accuracy of 86.6% after 4 epochs of training, which is on par with the previously published state of the art for sentence embedding approaches using BiLSTM and generalized pooling (Chen et al.). However, our model requires less trainable parameters than Chen et al.'s model (32m vs 65m). Comparison of our results with the previous state of the art and selected other sentence embedding based results are reported in Table 1.

| Model | Accuracy |
|---|---|
| BiLSTM Max Pool (InferSent)[a] | 84.5 |
| Distance-based Self-Attention[b] | 86.3 |
| ReSA[c] | 86.3 |
| 600D BiLSTM with generalized pooling[d] | 86.6 |
| Our HBMP (4 epochs of training) | **86.6** |

Table 1: SNLI test accuracies (%). Results marked with [a] by Conneau et al. (2017), [b] by Im and Cho (2017), [c] by Shen et al. (2018). and [d] by (Chen et al.)

### MultiNLI

For the MultiNLI matched test set (MultiNLI-m) our model achieves a test accuracy of 73.7% after 3 epochs of training, which is 0.8% points lower than the state of the art 74.5% by Nie and Bansal (2017a). For the mismatched test set (MultiNLI-mm) our model achieves a test accuracy of 73.0% after 3 epochs of training, which is 0.6% points lower than the state of the art 73.6% by Chen et al. (2017b).

Comparison of our results with the previous state of the art and selected sentence embedding based results are reported in Table 2.

Although we did not achieve state of the art results for the MultiNLI dataset, we believe that a systematic study of different hierarchical BiLSTM max pooling structures could reveal an architecture providing the needed improvement.

### SciTail

On the SciTail dataset we compared our model against non-sentence embedding-based models, as no results have been previously published which are based only on sentence embeddings. For this reason we also implemented the basic LSTM (using the last hidden state as the output) and BiLSTM with max pooling models and tested them on the SciTail dataset.

For SciTail we obtain a test score of 86.0% after 4 epochs of training, which is +2.7% points abso-

| Model | Accuracy (MultiNLI-m) | Accuracy (MultiNLI-mm) |
|---|---|---|
| CBOW[a] | 66.2 | 64.6 |
| BiLSTM[a] | 67.5 | 67.1 |
| BiLSTM + enh embed + max pooling[b] | 70.7 | 70.8 |
| BiLSTM + Inner-attention[c] | 72.1 | 72.1 |
| Deep Gated Attn. BiLSTM encoders[d] | 73.5 | **73.6** |
| Shortcut-Stacked BiLSTM[e] | **74.5** | 73.5 |
| Our HBMP | 73.7 | 73.0 |

Table 2: MultiNLI test accuracies (%). Results marked with [a] are baseline results by Williams et al. (2018), [b] by Vu (2017), [c] by Balazs et al. (2017), [d] by Chen et al. (2017b) and [e] by Nie and Bansal (2017a). Our results for the MultiNLI test sets were obtained by submitting the predictions to the respective Kaggle competitions.

| Model | Accuracy |
|---|---|
| DecompAtt[a] | 72.3 |
| ESIM[a] | 70.6 |
| Ngram[a] | 70.6 |
| DGEM w/o edges[a] | 70.8 |
| DGEM[a] | 77.3 |
| CAFE[b] | 83.3 |
| Our LSTM | 67.3 |
| Our BiLSTM max pooling | 84.9 |
| Our HBMP | **86.0** |

Table 3: SciTail test accuracies (%). As the previously published results are not based on purely sentence embeddings we include our results for simple LSTM and BiLSTM with max pooling. Results marked with [a] are baseline results reported by Khot et al. (2018) and [b] by Tay et al. (2018).

lute improvement on the previous state of the art by Tay et al. (2018).

The results achieved by our proposed model are significantly higher than the previously published results. It has been argued that the lexical similarity of the sentences in SciTail sentence pairs make it a particularly difficult dataset (Khot et al., 2018). We hypothesize that our model is indeed better at identifying entailment relations beyond focusing on the lexical similarity or difference of the sentences.

## 6 Error Analysis

To understand better what kind of inferential relationships our model is able to identify, we conducted an error analysis for the three datasets. We report the results below.[3] We also conducted a linguistic error analysis and compared our results to the results obtained with the InferSent BiLSTM max pooling model of Conneau et al. (2017) (our implementation).[4]

[3]For more detailed error statistics, see the appendix.

[4]The scores for our implementation of InferSent are on par or slightly higher than the scores reported by Conneau et al. (2017) using their training setup.

### SNLI

On the SNLI dataset our model makes the least errors on sentence pairs labeled with entailment, having an accuracy of: 90.5%. The model is also almost effective in predicting contradictions, with an accuracy of 87.7%. Our model is less effective in recognizing correctly sentence pairs labeled with neutral, having an accuracy of 81.5%. Table 4 summarizes the prediction accuracies per each gold label and compares our results to InferSent results.

We also tested our model against the recently published Breaking NLI test set for the SNLI (Glockner et al., 2018), designed to test NLI models' ability to recognize inferences requiring lexical knowledge. Our model outperforms InferSent on 7 out of 14 lexical categories and achieves the overall accuracy of 65.1% compared to InferSent's score of 65.6%. Detailed description of our results for the Breaking NLI dataset are included in the appendix.

| Label | HBMP | InferSent |
|---|---|---|
| contradiction | **87.7** | 85.1 |
| entailment | **90.5** | 88.1 |
| neutral | **81.5** | 79.9 |

Table 4: SNLI error analysis by gold label (accuracy %), comparing HBMP to InferSent BiLSTM max pooling (Conneau et al., 2017) (our implementation).

### MultiNLI

As the labeled test data is not openly available for MultiNLI, we analyzed the error statistics for this dataset based on the development data.

For the matched dataset (MultiNLI-m) our model had a development accuracy of 73.2%. On this dataset our model makes fewer errors on sentence pairs labeled as entailment, having an accuracy of 75.4% compared to pairs with label contradiction, where the accuracy was 73.1% and neu-

| Label | HBMP | InferSent |
|---|---|---|
| contradiction | **73.1** | 69.8 |
| entailment | **75.4** | 75.0 |
| neutral | **70.8** | 68.5 |

Table 5: MultiNLI-m error analysis by gold label (accuracy %), comparing HBMP to InferSent BiLSTM max pooling (Conneau et al., 2017) (our implementation).

tral, where the accuracy was 70.8%. Table 5 lists the prediction accuracies per each gold label and compares the results to InferSent results.

For the mismatched dataset (MultiNLI-mm) our model had a development accuracy of 74.2%. On this dataset our model makes significantly fewer errors on sentence pairs labeled as entailment, having an accuracy of 80.3% compared to pairs with label contradiction, where the accuracy was 72.7% and neutral, where the accuracy was just 69.0%. Table 6 summarizes the prediction accuracies per each gold label and compares the results to InferSent results.

We also conducted additional linguistic error analysis using the annotation test set provided for MultiNLI. The results are reported in the appendix and they are mostly inconclusive.

| Label | HBMP | InferSent |
|---|---|---|
| contradiction | **72.7** | 68.7 |
| entailment | **80.3** | 78.9 |
| neutral | **69.0** | 64.2 |

Table 6: MultiNLI-mm error analysis by gold label (accuracy %), comparing HBMP to InferSent BiLSTM max pooling (Conneau et al., 2017) (our implementation).

### SciTail

For SciTail, unlike for the SNLI and both of the MultiNLI datasets, our model makes more errors on sentence pairs labeled as entailment, having an accuracy of 75.1% compared to pairs marked with neutral, where the accuracy was 93.1%. Table 7 summarizes the prediction accuracies per each of the two gold labels and compares the results to InferSent results.

| Label | HBMP | InferSent |
|---|---|---|
| entailment | 75.1 | **79.9** |
| neutral | **93.1** | 89.1 |

Table 7: SciTail error analysis by gold label (accuracy %), comparing HBMP to InferSent BiLSTM max pooling (Conneau et al., 2017) (our implementation).

## 7 Transfer Learning

To better understand how well our model generalizes to different tasks, we conducted additional transfer learning tests using the SentEval sentence embedding evaluation library[5] (Conneau et al., 2017) and compared our results to the results published for InferSent and SkipThought (Kiros et al., 2015).

For the transfer learning tasks, we trained our model on NLI data consisting of the concatenation of the SNLI and MultiNLI training sets consisting of 942,854 sentence pairs in total. This allows us to compare our results to the InferSent results which were obtained using a model trained on the same data (Conneau et al., 2017). Conneau et al. (2017) have shown that including all the training data from SNLI and MultiNLI improves significantly the model performance on transfer learning tasks, compared to training the model only on SNLI data. For training the model we used the same setup as described above in Section 4.

We used the SentEval sentence embedding evaluation library using the default settings[6] recommended on the SentEval website, with a logistic regression classifier, Adam optimizer with learning rate of 0.001, batch size of 64 and epoch size of 4. Table 8 lists the transfer learning results for our models with 1200D and 2400D hidden dimensionality and compares our model to the InferSent and SkipThought scores reported by Conneau et al. (2017). The downstream datasets included in the tests were MR movie reviews, CR product reviews, SUBJ subjectivity status, MPQA opinion-polarity, SST binary sentiment analysis, TREC question-type classification, MRPC paraphrase detection, SICK-Relatedness (SICK-R) semantic textual similarity, SICK-Entailment (SICK-E) natural language inference and STS14 semantic textual similarity.

Our 2400D model outperforms the InferSent model on 7 out of 10 tasks. The model achieves higher score on 8 out of 9 tasks reported for SkipThought, having equal score on the SUBJ dataset. No MRPC results have been reported for SkipThought.

To study in more detail the linguistic properties

[5]The SentEval test suite for evaluating sentence embeddings is available online at https://github.com/facebookresearch/SentEval.

[6]Refer to the SentEval website for details about the different settings: https://github.com/facebookresearch/SentEval.

| Model | MR | CR | SUBJ | MPQA | SST | TREC | MRPC | SICK-R | SICK-E | STS14 |
|---|---|---|---|---|---|---|---|---|---|---|
| InferSent | 81.1 | 86.3 | 92.4 | 90.2 | **84.6** | 88.2 | 76.2/83.1 | **0.884** | **86.3** | .70/.67 |
| SkipThought | 79.4 | 83.1 | **93.7** | 89.3 | 82.9 | 88.4 | - | 0.858 | 79.5 | .44/.45 |
| Our 1200D HBMP | 81.5 | 86.4 | 92.7 | 89.8 | 83.6 | 86.4 | 74.6/82.0 | 0.876 | 85.3 | .70/.66 |
| Our 2400D HBMP | **81.7** | **87.0** | **93.7** | **90.3** | 84.0 | **88.8** | **76.7/83.4** | 0.876 | 84.7 | **.71/.68** |

Table 8: Transfer learning test results for the HBMP model on a number of SentEval downstream sentence embedding evaluation tasks. InferSent and SkipThought results as reported by Conneau et al. (2017).

| Model | SentLen | WC | TreeDepth | TopConst | BShift | Tense | SubjNum | ObjNum | SOMO | CoordInv |
|---|---|---|---|---|---|---|---|---|---|---|
| InferSent | 71.7 | **87.3** | 41.6 | 70.5 | 65.1 | 86.7 | 80.7 | 80.3 | **62.1** | 66.8 |
| Our 1200D HBMP | **75.9** | 84.1 | 42.9 | 76.6 | 64.3 | 86.2 | 83.7 | 79.3 | 58.9 | 68.5 |
| Our 2400D HBMP | 75.0 | 85.3 | **43.8** | **77.2** | **65.6** | **88.0** | **87.0** | **81.8** | 59.0 | **70.8** |

Table 9: SentEval probing task results (accuracy %). InferSent results are BiLSTM Max (NLI) results as reported by Conneau et al. (2018).

of our proposed model, we ran the recently published SentEval probing tasks described in (Conneau et al., 2018). Our 24000D HBMP model outperforms the InferSent BiLSTM max pooling model in 8 out of 10 probing tasks. These results further highlight our model's applicability for various NLP tasks. The results are listed in Table 9.

## 8 Conclusion

In this paper we have introduced an architecture based on BiLSTM max pooling, that achieves on par results with the current state of the art for SNLI sentence encoding-based models and a new state of the art score for SciTail.

We furthermore tested our model using the SentEval sentence embedding evaluation library, showing that our model achieves great generalization capability, outperforming InferSent on 7 out of 10 downstream and 8 out of 10 probing tasks, and SkipThought on 8 out of 9 downstream tasks. Our model also achieves a strong result on the Breaking NLI dataset outperforming Infersent on 11 out of 14 lexical categories.

The success of the proposed hierarchical architecture raises a number of additional interesting questions. First, it would be important to understand what kind of semantic information the different layers are able to capture. Second, a detailed and systematic comparison of different hierarchical architecture configurations, combining BiLSTM and max pooling in different ways, could lead to even stronger results, as indicated by the results we obtained on the SciTail dataset with the modified 4-layered model.

Also, as the sentence embedding approaches for NLI focus mostly on the sentence encoder, we think that more should be done to study the classifier part of the overall NLI architecture. There is not enough research on classifiers for NLI and we hypothesize that further improvements can be achieved by a systematic study of different classifier architectures, starting from the way the two sentence embeddings are combined before passing on to the classifier. This is also something we intend to undertake as a next step.

Finally, there are number of other NLP tasks, including neural machine translation (NMT), where HBMP models should be evaluated. We plan to evaluate the performance of the HBMP encoder in other NLP tasks like on encoder-decoder NMT models in the future.

## References


Jorge Balazs, Edison Marrese-Taylor, Pablo Loyola, and Yutaka 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*, pages 51–55. Association for Computational Linguistics.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3:1137–1155.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Samuel R. Bowman, Jon Gauthier, Abhinav Rastogi, Raghav Gupta, Christopher D. Manning, and Christopher Potts. 2016. A fast unified model for

parsing and sentence understanding. *arXiv preprint arXiv:1603.06021*.

Q. Chen, Z.-H. Ling, and X. Zhu. Enhancing Sentence Embedding with Generalized Pooling. *arXiv preprint arXiv:1806.09828*.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Diana Inkpen, and Si Wei. 2018. Neural natural language inference models enhanced with external knowledge. In *The 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, Melbourne, Australia.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017a. Enhanced lstm for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1657–1668. Association for Computational Linguistics.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017b. Recurrent neural network-based sentence encoder with gated attention for natural language inference. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pages 36–40. Association for Computational Linguistics.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364*.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loc Barrault, and Marco Baroni. 2018. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. *arXiv preprint arXiv:1805.01070*.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking nli systems with sentences that require simple lexical inferences. In *The 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, Melbourne, Australia.

Felix Hill, Kyunghyun Cho, and Anna Korhonen. 2016. Learning distributed representations of sentences from unlabelled data. *arXiv preprint arXiv:1602.03483*.

Jinbae Im and Sungzoon Cho. 2017. Distance-based self-attention network for natural language inference. *arXiv preprint arXiv:1712.02047*.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2018. Scitail: A textual entailment dataset from science question answering. In *AAAI*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-thought vectors. *arXiv preprint arXiv:1506.06726*.

Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the 30th International Conference on Machine Learning*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, pages 3111–3119, USA. Curran Associates Inc.

L. Mou, R. Men, G. Li, Y. Xu, L. Zhang, R. Yan, and Z. Jin. 2015. Natural Language Inference by Tree-Based Convolution and Heuristic Matching. *arXiv preprint arXiv:1512.08422*.

Lili Mou, Rui Men, Ge Li, Yan Xu, Lu Zhang, Rui Yan, and Zhi Jin. 2016. Natural language inference by tree-based convolution and heuristic matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 130–136. Association for Computational Linguistics.

Yixin Nie and Mohit Bansal. 2017a. Shortcut-stacked sentence encoders for multi-domain inference. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pages 41–45. Association for Computational Linguistics.

Yixin Nie and Mohit Bansal. 2017b. Shortcut-stacked sentence encoders for multi-domain inference. *arXiv preprint arXiv:1708.02312*.

Ankur P. Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933*.

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

Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, Sen Wang, and Chengqi Zhang. 2018. Reinforced self-attention network: a hybrid of hard and soft attention for sequence modeling. *arXiv preprint arXiv:1801.10296*.

Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. A compare-propagate architecture with alignment factorization for natural language inference. *arXiv preprint arXiv:1801.00102*.

Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. 2015. Order-embeddings of images and language. *arXiv preprint arXiv:1511.06361*.

Hoa Vu. 2017. Lct-malta's submission to repeval 2017 shared task. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pages 56–60. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel R. 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 (NAACL)*. Association for Computational Linguistics.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

## A Detailed Error Statistics for the HBMP model

To see in more detail how our HBMP model is able to classify sentence pairs with different gold labels and what kind of errors it makes, we summarize error statistics as confusion matrices for the different datasets. Figure 3 contains the confusion matrix for the SNLI dataset. Figure 4 contains the confusion matrix for the MultiNLI Matched dataset. Figure 5 contains the confusion matrix for the MultiNLI Mismatched dataset. Finally, Figure 6 contains the confusion matrix for the SciTail dataset.

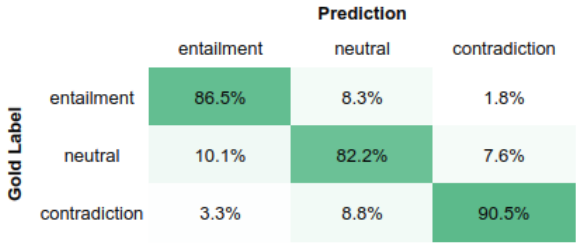



Figure 3: SNLI Confusion Matrix.

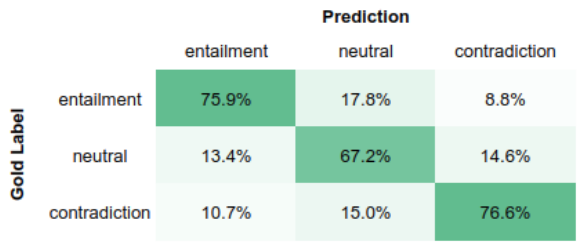



Figure 4: MultiNLI Matched Confusion Matrix.

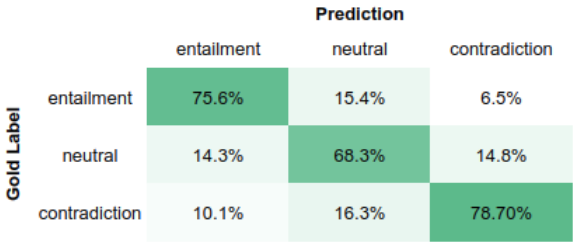



Figure 5: MultiNLI Mismatched Confusion Matrix.

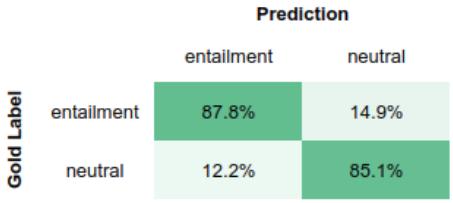



Figure 6: SciTail Confusion Matrix.

The confusion matrices highlight the HBMP model's strong performance across all the labels. However, the accuracy for neutral in the SNLI and MultiNLI datasets is clearly lower than for the other two labels.

## B Linguistic Error Analysis

We also conducted a linguistic error analysis for the HBMP model using the MultiNLI annotation set. We compare our model and the type of linguistic reasoning it is capable of to InferSent BiLSTM max pooling model (Conneau et al., 2017), in order to see what benefits are achieved by adding a hierarchical BiLSTM max pooling structure on top of the basic BiLSTM max pooling architecture. We provide a detailed comparison of the prediction accuracies of our HBMP model with InferSent with respect to the type of linguistic properties present in the sentence pairs. Table 10 contains the comparison for MultiNLI-m dataset and Table 11 for MultiNLI-mm dataset.

The analysis show that our HBMP model outperforms InferSent in some of the categories, but the results are mostly at this point inconclusive.

## C Additional tests with the Breaking NLI dataset

We conducted additional testing of the proposed sentence embedding architecture using the Breaking NLI test set recently published by Glockner et al. (2018). The test set is designed to highlight

| | Entailment | | Contradiction | | Neutral | |
|---|---|---|---|---|---|---|
| | **HBMP** | **InferSent** | **HBMP** | **InferSent** | **HBMP** | **InferSent** |
| ACTIVE/PASSIVE | 88.9 | **91.7** | 100 | 100 | - | **100** |
| ANTO | - | **71.4** | 76.9 | **100** | **85.7** | 66.7 |
| BELIEF | 70.6 | **82.6** | 55.6 | **62.5** | **91.3** | 63.2 |
| CONDITIONAL | **81.8** | 50 | **37.5** | 36.4 | 71.4 | **83.3** |
| COREF | **75** | 63.6 | 78.6 | **100** | 72.7 | **83.3** |
| LONG SENTENCE | **67.7** | 67.4 | **61.1** | 52.4 | **76.2** | 71.4 |
| MODAL | 76.6 | **77.6** | 60 | **71.1** | **84.1** | 82 |
| NEGATION | **60** | 56.3 | **76.4** | 76.3 | 66.7 | **68.4** |
| PARAPHRASE | 83.8 | **88** | - | - | - | - |
| QUANTIFIER | 67.5 | **80** | 71.2 | **75** | **81.3** | 73.9 |
| QUANTITY/TIME REASONING | **58.3** | 50 | 25 | **36.4** | 0 | - |
| TENSE DIFFERENCE | 75 | **92.9** | 0 | **64.7** | **91.7** | 80 |
| WORD OVERLAP | 95.2 | **100** | 33.3 | **50** | 75 | **100** |
| Total | **81.2** | 76.4 | 66.7 | **72.1** | **76.4** | 75.2 |

Table 10: MultiNLI-m linguistic error analysis (accuracy %), comparing our HBMP results to the InferSent BiLSTM max pooling model (Conneau et al., 2017) results (our implementation).

| | Entailment | | Contradiction | | Neutral | |
|---|---|---|---|---|---|---|
| | **HBMP** | **InferSent** | **HBMP** | **InferSent** | **HBMP** | **InferSent** |
| ACTIVE/PASSIVE | 88.9 | **100.0** | 100.0 | 100.0 | - | - |
| ANTO | - | - | **76.9** | 69.2 | **85.7** | 71.4 |
| BELIEF | 70.6 | **82.4** | 55.6 | **61.1** | **91.3** | 78.3 |
| CONDITIONAL | 81.8 | 81.8 | 37.5 | **62.5** | 71.4 | 71.4 |
| COREF | 75.0 | 75.0 | **78.6** | 64.3 | 72.7 | **81.8** |
| LONG SENTENCE | 67.7 | **77.4** | 61.1 | 61.1 | **76.2** | 71.4 |
| MODAL | 76.6 | **78.7** | 60.0 | **68.6** | **84.1** | 70.5 |
| NEGATION | 60.0 | **64.0** | 76.4 | 76.4 | **66.7** | 45.8 |
| PARAPHRASE | 83.8 | **86.5** | - | - | - | - |
| QUANTIFIER | 67.5 | **72.5** | 71.2 | **73.1** | **81.3** | 77.1 |
| QUANTITY/TIME REASONING | **58.3** | 50.0 | 25.0 | **41.7** | 0.0 | **33.3** |
| TENSE DIFFERENCE | 75.0 | 75.0 | 0.0 | 0.0 | **91.7** | 75.0 |
| WORD OVERLAP | 95.2 | 95.2 | 33.3 | **41.7** | **75.0** | 0.0 |
| Total | 81.2 | **83.0** | 66.7 | **68.5** | **76.4** | 72.7 |

Table 11: MultiNLI-mm linguistic error analysis (accuracy %), comparing our HBMP results to the InferSent BiLSTM max pooling model (Conneau et al., 2017) results (our implementation).

the lack of lexical reasoning capability of NLI systems.

We trained our HBMP model and the InferSent model using the SNLI training data. We compare our results with the results published by Glockner et al. (2018) and to results obtained with InferSent sentence encoder (our implementation).

Our HBMP model outperforms the InferSent model in 7 out of 14 categories, receiving an overall score of 65.1% (InferSent: 65.6%). Our model also compares well against the other models, outperforming Decomposable Attention model (51.90%) (Parikh et al., 2016) and Residual Encoders (62.20%) (Nie and Bansal, 2017b) in the overall score. As these models are not based purely on sentence embeddings, the obtained result highlights that sentence embedding approaches can be competitive when handling inferences requiring lexical information. Our model is still outperformed by and ESIM (Chen et al., 2017a) and KIM, an ESIM model incorporating external knowledge (Chen et al., 2018). The results of the comparison are summarized in Table 12.

| **Category** | **Decomposable Attention*** | **ESIM*** | **Residual Encoders*** | **WordNet Baseline*** | **KIM*** | **InferSent** | **Our HBMP** |
|---|---|---|---|---|---|---|---|
| antonyms | 41.6 | 70.4 | 58.2 | 95.5 | 86.5 | 51.6 | **54.7** |
| antonyms (wordnet) | 55.1 | 74.6 | 67.9 | 94.5 | 78.8 | 63.7 | **69.1** |
| cardinals | 53.5 | 75.5 | 53.1 | 98.6 | 93.4 | 49.4 | **58.8** |
| colors | 85 | 96.1 | 87 | 98.7 | 98.3 | **90.6** | 90.4 |
| countries | 15.2 | 25.4 | 66.2 | 100 | 70.8 | 77.2 | **81.2** |
| drinks | 52.9 | 63.7 | 52 | 94.8 | 96.6 | **85.1** | 81.3 |
| instruments | 96.9 | 90.8 | 96.9 | 67.7 | 96.9 | **98.5** | 96.9 |
| materials | 65.2 | 89.7 | 79.9 | 75.3 | 98.7 | 81.6 | **82.6** |
| nationalities | 37.5 | 35.9 | 70.9 | 78.5 | 73.5 | 47.3 | **49.8** |
| ordinals | 2.1 | 21 | 5.4 | 40.7 | 56.6 | **7.4** | 4.5 |
| planets | 31.7 | 3.3 | 21.7 | 100 | 5 | **75** | 45.0 |
| rooms | 59.2 | 69.4 | 63.4 | 89.9 | 77.6 | **76.3** | 72.1 |
| synonyms | 97.5 | 99.7 | 86.1 | 70.5 | 92.1 | **99.6** | 84.5 |
| vegetables | 43.1 | 31.2 | 37.6 | 86.2 | 79.8 | 39.5 | **40.4** |
| Total | 51.9 | 65.6 | 62.2 | 85.8 | 83.5 | **65.6** | 65.1 |

Table 12: Breaking NLI scores (accuracy %). Results marked with * as reported by Glockner et al. (2018). InferSent results obtained with our implementation using the architecture and training set-up described in (Conneau et al., 2017). Scores highlighted with bold are top scores when comparing the InferSent and our HBMP model.