Predicting Human Acceptability Judgments in Context

Deep Learning and the Nature of Linguistic Representation

Shalom Lappin

University of Gothenburg, Queen Mary University of London, and King's College London

WeSSLLI 2020, Brandeis University

July 16, 2020

Outline

Modeling Acceptability Independently of Context

- LCL and EBL test speakers' acceptability judgments for sentences presented outside of any context beyond the HIT in which they appear.
- The sentences in each HIT are randomly selected.
- Their models predict acceptability without reference to context.
- Document context is not explicitly represented in any of the models in either training or testing.

Modeling Acceptability Independently of Context

- LCL and EBL test speakers' acceptability judgments for sentences presented outside of any context beyond the HIT in which they appear.
- The sentences in each HIT are randomly selected.
- Their models predict acceptability without reference to context.
- Document context is not explicitly represented in any of the models in either training or testing.

Modeling Acceptability Independently of Context

- LCL and EBL test speakers' acceptability judgments for sentences presented outside of any context beyond the HIT in which they appear.
- The sentences in each HIT are randomly selected.
- Their models predict acceptability without reference to context.
- Document context is not explicitly represented in any of the models in either training or testing.

Modeling Acceptability Independently of Context

- LCL and EBL test speakers' acceptability judgments for sentences presented outside of any context beyond the HIT in which they appear.
- The sentences in each HIT are randomly selected.
- Their models predict acceptability without reference to context.
- Document context is not explicitly represented in any of the models in either training or testing.

Human Acceptability Judgments in Context

- Bernardy, Lappin, and Lau (2018) (BLL) constructed two datasets of sentences annotated with acceptability ratings, one judged with, and the other without document context.
- They extracted 100 random articles from the English Wikipedia and sampled a sentence from each article.
- They tried LCL's method of using Google Translate (GT) for round-trip MT to generate a set of sentences with varying degrees of acceptability.
- GT has improved to the point that a pilot study indicated that human annotators rated most round-trip translated sentences as highly as the English originals.

Human Acceptability Judgments in Context

- Bernardy, Lappin, and Lau (2018) (BLL) constructed two datasets of sentences annotated with acceptability ratings, one judged with, and the other without document context.
- They extracted 100 random articles from the English Wikipedia and sampled a sentence from each article.
- They tried LCL's method of using Google Translate (GT) for round-trip MT to generate a set of sentences with varying degrees of acceptability.
- GT has improved to the point that a pilot study indicated that human annotators rated most round-trip translated sentences as highly as the English originals.

Human Acceptability Judgments in Context

- Bernardy, Lappin, and Lau (2018) (BLL) constructed two datasets of sentences annotated with acceptability ratings, one judged with, and the other without document context.
- They extracted 100 random articles from the English Wikipedia and sampled a sentence from each article.
- They tried LCL's method of using Google Translate (GT) for round-trip MT to generate a set of sentences with varying degrees of acceptability.
- GT has improved to the point that a pilot study indicated that human annotators rated most round-trip translated sentences as highly as the English originals.

Human Acceptability Judgments in Context

- Bernardy, Lappin, and Lau (2018) (BLL) constructed two datasets of sentences annotated with acceptability ratings, one judged with, and the other without document context.
- They extracted 100 random articles from the English Wikipedia and sampled a sentence from each article.
- They tried LCL's method of using Google Translate (GT) for round-trip MT to generate a set of sentences with varying degrees of acceptability.
- GT has improved to the point that a pilot study indicated that human annotators rated most round-trip translated sentences as highly as the English originals.

The Out-of-Context Annotated Test Set

- As an alternative, BLL used the more traditional statistical phrase based Moses MT system (Koehn et al., 2007).
- They applied the pretrained Moses models for round-trip MT into Czech, Spanish, German, and French, and then back to English.
- This provided a distribution of acceptability judgments over sentences comparable to those which LCL obtained in their experiments.
- Following LCL's protocol, BLL used HITS with a four category acceptability rating for crowd source annotation of both out-of-context and in-context data sets.

The Out-of-Context Annotated Test Set

- As an alternative, BLL used the more traditional statistical phrase based Moses MT system (Koehn et al., 2007).
- They applied the pretrained Moses models for round-trip MT into Czech, Spanish, German, and French, and then back to English.
- This provided a distribution of acceptability judgments over sentences comparable to those which LCL obtained in their experiments.
- Following LCL's protocol, BLL used HITS with a four category acceptability rating for crowd source annotation of both out-of-context and in-context data sets.

The Out-of-Context Annotated Test Set

- As an alternative, BLL used the more traditional statistical phrase based Moses MT system (Koehn et al., 2007).
- They applied the pretrained Moses models for round-trip MT into Czech, Spanish, German, and French, and then back to English.
- This provided a distribution of acceptability judgments over sentences comparable to those which LCL obtained in their experiments.
- Following LCL's protocol, BLL used HITS with a four category acceptability rating for crowd source annotation of both out-of-context and in-context data sets.

The Out-of-Context Annotated Test Set

- As an alternative, BLL used the more traditional statistical phrase based Moses MT system (Koehn et al., 2007).
- They applied the pretrained Moses models for round-trip MT into Czech, Spanish, German, and French, and then back to English.
- This provided a distribution of acceptability judgments over sentences comparable to those which LCL obtained in their experiments.
- Following LCL's protocol, BLL used HITS with a four category acceptability rating for crowd source annotation of both out-of-context and in-context data sets.

The In-Context Annotated Test Set

- The target sentence was highlighted in boldface, with one preceding and one succeeding sentence included as additional context.
- Annotators had the option of revealing the full document context by clicking on the preceding and succeeding sentences.
- As in the out-of-context test set, sentences were presented in HITS of five, one from the original English set, and four from the round-trip translations.
- Each HIT contained one sentence per target language, with no sentence type appearing more than once in a HIT.

The In-Context Annotated Test Set

- The target sentence was highlighted in boldface, with one preceding and one succeeding sentence included as additional context.
- Annotators had the option of revealing the full document context by clicking on the preceding and succeeding sentences.
- As in the out-of-context test set, sentences were presented in HITS of five, one from the original English set, and four from the round-trip translations.
- Each HIT contained one sentence per target language, with no sentence type appearing more than once in a HIT.

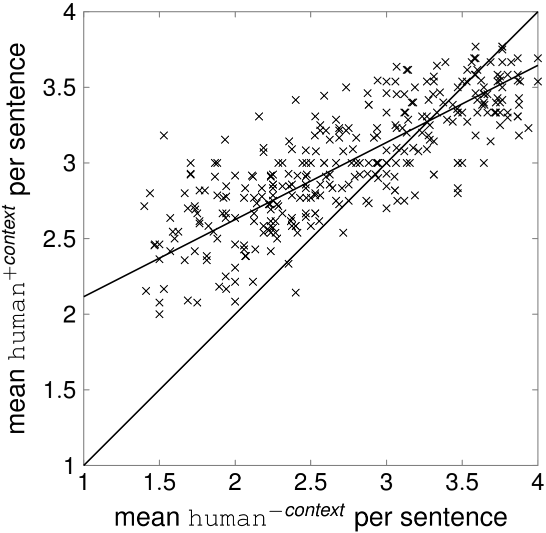
The In-Context Annotated Test Set

- The target sentence was highlighted in boldface, with one preceding and one succeeding sentence included as additional context.
- Annotators had the option of revealing the full document context by clicking on the preceding and succeeding sentences.
- As in the out-of-context test set, sentences were presented in HITS of five, one from the original English set, and four from the round-trip translations.
- Each HIT contained one sentence per target language, with no sentence type appearing more than once in a HIT.

The In-Context Annotated Test Set

- The target sentence was highlighted in boldface, with one preceding and one succeeding sentence included as additional context.
- Annotators had the option of revealing the full document context by clicking on the preceding and succeeding sentences.
- As in the out-of-context test set, sentences were presented in HITS of five, one from the original English set, and four from the round-trip translations.
- Each HIT contained one sentence per target language, with no sentence type appearing more than once in a HIT.

Annotation Results



Analysing the Effect of Context on Acceptability Judgments

- BLL found a strong Pearson's r correlation of 0.80 between mean out-of-context and in-context judgments.
- The average difference between $\texttt{human}^{-context}$ and $\texttt{human}^{+context}$ is represented by the distance between the linear regression and the full diagonal in the graph.
- These lines cross at $\texttt{human}^{+context} = \texttt{human}^{-context} = 3.28$, the point where context no longer boosts acceptability.

Analysing the Effect of Context on Acceptability Judgments

- BLL found a strong Pearson's r correlation of 0.80 between mean out-of-context and in-context judgments.
- The average difference between $\texttt{human}^{-context}$ and $\texttt{human}^{+context}$ is represented by the distance between the linear regression and the full diagonal in the graph.
- These lines cross at $\texttt{human}^{+context} = \texttt{human}^{-context} = 3.28$, the point where context no longer boosts acceptability.

Analysing the Effect of Context on Acceptability Judgments

- BLL found a strong Pearson's r correlation of 0.80 between mean out-of-context and in-context judgments.
- The average difference between $\texttt{human}^{-context}$ and $\texttt{human}^{+context}$ is represented by the distance between the linear regression and the full diagonal in the graph.
- These lines cross at $\texttt{human}^{+context} = \texttt{human}^{-context} = 3.28$, the point where context no longer boosts acceptability.

The Compression Effect

- Adding context generally improves acceptability, but the pattern reverses as acceptability approaches maximal mean rating values.
- This “compresses” the distribution of (mean) ratings, pushing the extremes to the middle.
- The net effect of this compression lowers correlation, as the good and bad sentences for the in-context test set are not as clearly separable as they are in the out-of context test set.

The Compression Effect

- Adding context generally improves acceptability, but the pattern reverses as acceptability approaches maximal mean rating values.
- This "compresses" the distribution of (mean) ratings, pushing the extremes to the middle.
- The net effect of this compression lowers correlation, as the good and bad sentences for the in-context test set are not as clearly separable as they are in the out-of context test set.

The Compression Effect

- Adding context generally improves acceptability, but the pattern reverses as acceptability approaches maximal mean rating values.
- This “compresses” the distribution of (mean) ratings, pushing the extremes to the middle.
- The net effect of this compression lowers correlation, as the good and bad sentences for the in-context test set are not as clearly separable as they are in the out-of context test set.

A Similar Pattern for a Different Task

- Bizzoni and Lappin (2019) (BL19) test the effect of context on gradient judgments of paraphrase for a metaphorical sentence.
- They solicit AMT crowd source ratings for pairs containing a metaphorical sentence, and a candidate for a literal paraphrase of that sentence.
- In one test set 200 pairs are rated on a four category scale of paraphrase appropriateness, independently of context.
- In the second test set the same pairs are judged within a context of a preceding and a following sentence.
- BL19 observe the same compression effect with in-context paraphrase judgments that BLL obtain for in-context acceptability ratings.

A Similar Pattern for a Different Task

- Bizzoni and Lappin (2019) (BL19) test the effect of context on gradient judgments of paraphrase for a metaphorical sentence.
- They solicit AMT crowd source ratings for pairs containing a metaphorical sentence, and a candidate for a literal paraphrase of that sentence.
- In one test set 200 pairs are rated on a four category scale of paraphrase appropriateness, independently of context.
- In the second test set the same pairs are judged within a context of a preceding and a following sentence.
- BL19 observe the same compression effect with in-context paraphrase judgments that BLL obtain for in-context acceptability ratings.

A Similar Pattern for a Different Task

- Bizzoni and Lappin (2019) (BL19) test the effect of context on gradient judgments of paraphrase for a metaphorical sentence.
- They solicit AMT crowd source ratings for pairs containing a metaphorical sentence, and a candidate for a literal paraphrase of that sentence.
- In one test set 200 pairs are rated on a four category scale of paraphrase appropriateness, independently of context.
- In the second test set the same pairs are judged within a context of a preceding and a following sentence.
- BL19 observe the same compression effect with in-context paraphrase judgments that BLL obtain for in-context acceptability ratings.

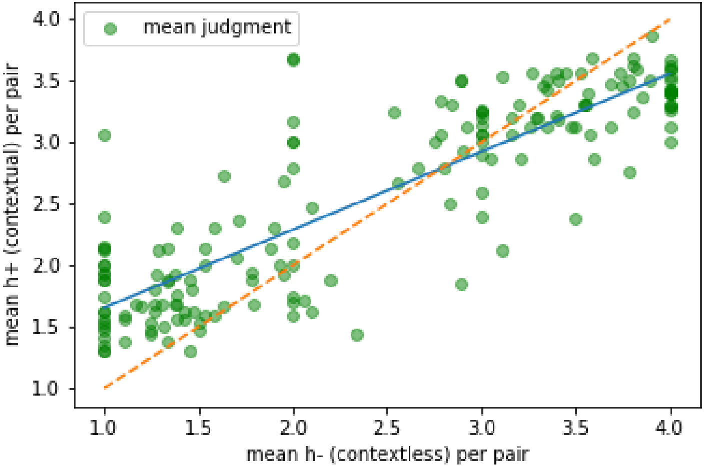
A Similar Pattern for a Different Task

- Bizzoni and Lappin (2019) (BL19) test the effect of context on gradient judgments of paraphrase for a metaphorical sentence.
- They solicit AMT crowd source ratings for pairs containing a metaphorical sentence, and a candidate for a literal paraphrase of that sentence.
- In one test set 200 pairs are rated on a four category scale of paraphrase appropriateness, independently of context.
- In the second test set the same pairs are judged within a context of a preceding and a following sentence.
- BL19 observe the same compression effect with in-context paraphrase judgments that BLL obtain for in-context acceptability ratings.

A Similar Pattern for a Different Task

- Bizzoni and Lappin (2019) (BL19) test the effect of context on gradient judgments of paraphrase for a metaphorical sentence.
- They solicit AMT crowd source ratings for pairs containing a metaphorical sentence, and a candidate for a literal paraphrase of that sentence.
- In one test set 200 pairs are rated on a four category scale of paraphrase appropriateness, independently of context.
- In the second test set the same pairs are judged within a context of a preceding and a following sentence.
- BL19 observe the same compression effect with in-context paraphrase judgments that BLL obtain for in-context acceptability ratings.

BL19's Regression Graph for Paraphrase Judgments



Two DNN LMs

- BLL experiment with two Deep Neural Network Language models to predict the human sentence ratings for each of their test sets.
- `lstm` is a standard LSTM language model, trained over a corpus to predict word sequences.
- `tdlm` (Lau et al., 2017) is a topic driven neural LM.
- The topic model component of `tdlm` produces topics by processing documents through a convolutional layer, and aligning it with trainable topic embeddings.
- The language model component of `tdlm` incorporates context by combining its topic vector with its LSTM's hidden state, to generate the probability distribution for the next word.

Two DNN LMs

- BLL experiment with two Deep Neural Network Language models to predict the human sentence ratings for each of their test sets.
- `lstm` is a standard LSTM language model, trained over a corpus to predict word sequences.
- `tdlm` (Lau et al., 2017) is a topic driven neural LM.
- The topic model component of `tdlm` produces topics by processing documents through a convolutional layer, and aligning it with trainable topic embeddings.
- The language model component of `tdlm` incorporates context by combining its topic vector with its LSTM's hidden state, to generate the probability distribution for the next word.

Two DNN LMs

- BLL experiment with two Deep Neural Network Language models to predict the human sentence ratings for each of their test sets.
- `lstm` is a standard LSTM language model, trained over a corpus to predict word sequences.
- `tdlm` (Lau et al., 2017) is a topic driven neural LM.
- The topic model component of `tdlm` produces topics by processing documents through a convolutional layer, and aligning it with trainable topic embeddings.
- The language model component of `tdlm` incorporates context by combining its topic vector with its LSTM's hidden state, to generate the probability distribution for the next word.

Two DNN LMs

- BLL experiment with two Deep Neural Network Language models to predict the human sentence ratings for each of their test sets.
- `lstm` is a standard LSTM language model, trained over a corpus to predict word sequences.
- `tdlm` (Lau et al., 2017) is a topic driven neural LM.
- The topic model component of `tdlm` produces topics by processing documents through a convolutional layer, and aligning it with trainable topic embeddings.
- The language model component of `tdlm` incorporates context by combining its topic vector with its LSTM's hidden state, to generate the probability distribution for the next word.

Two DNN LMs

- BLL experiment with two Deep Neural Network Language models to predict the human sentence ratings for each of their test sets.
- `lstm` is a standard LSTM language model, trained over a corpus to predict word sequences.
- `tdlm` (Lau et al., 2017) is a topic driven neural LM.
- The topic model component of `tdlm` produces topics by processing documents through a convolutional layer, and aligning it with trainable topic embeddings.
- The language model component of `tdlm` incorporates context by combining its topic vector with its LSTM's hidden state, to generate the probability distribution for the next word.

Four LM Variants

- Both LMs can use the document context as a prefix input to the sentence at test time.
- This yields 4 variant LMs at test time.
  1. $\texttt{lstm}^{-c}$ and $\texttt{tdlm}^{-c}$, which use only sentences from a test set as input.
  2. $\texttt{lstm}^{+c}$ and $\texttt{tdlm}^{+c}$, which use sentence and context at test time.
- To map sentence probability to acceptability BLL use LCL's 3 scoring functions.

Four LM Variants

- Both LMs can use the document context as a prefix input to the sentence at test time.
- This yields 4 variant LMs at test time.
  1. $\texttt{lstm}^{-c}$ and $\texttt{tdlm}^{-c}$, which use only sentences from a test set as input.
  2. $\texttt{lstm}^{+c}$ and $\texttt{tdlm}^{+c}$, which use sentence and context at test time.
- To map sentence probability to acceptability BLL use LCL's 3 scoring functions.

Four LM Variants

- Both LMs can use the document context as a prefix input to the sentence at test time.
- This yields 4 variant LMs at test time.
  1. $\texttt{lstm}^{-c}$ and $\texttt{tdlm}^{-c}$, which use only sentences from a test set as input.
  2. $\texttt{lstm}^{+c}$ and $\texttt{tdlm}^{+c}$, which use sentence and context at test time.
- To map sentence probability to acceptability BLL use LCL's 3 scoring functions.

LCL Acceptability Scoring Functions

Scoring Function	Equation
LogProb	$\log P_m(\xi)$
Mean LP	$\frac{\log P_m(\xi)}{\lvert\xi\rvert}$
Norm LP (Div)	$-\frac{\log P_m(\xi)}{\log P_u(\xi)}$
SLOR	$\frac{\log P_m(\xi) - \log P_u(\xi)}{\lvert\xi\rvert}$

ξ = sentence;
$P_m(\xi)$ = the probability of the sentence given by the model;
$P_u(\xi)$ = is the unigram probability of sentence;
SLOR is proposed by Pauls and Klein (2012)

Training and Evaluation of the LMs

- BLL train `tdlm` and `lstm` on a sample of 100K English Wikipedia articles, which has no overlap with the 100 documents used for test set annotation.
- The training data has approximately 40M tokens and a vocabulary size of 66K
- To assess the performance of the acceptability measures, BLL compute Pearson's *r* against mean human ratings.
- BLL also experimented with Spearman's rank correlation, but found similar trends, and so they present only the Pearson results.

Training and Evaluation of the LMs

- BLL train `tdlm` and `lstm` on a sample of 100K English Wikipedia articles, which has no overlap with the 100 documents used for test set annotation.
- The training data has approximately 40M tokens and a vocabulary size of 66K
- To assess the performance of the acceptability measures, BLL compute Pearson's *r* against mean human ratings.
- BLL also experimented with Spearman's rank correlation, but found similar trends, and so they present only the Pearson results.

Training and Evaluation of the LMs

- BLL train `tdlm` and `lstm` on a sample of 100K English Wikipedia articles, which has no overlap with the 100 documents used for test set annotation.
- The training data has approximately 40M tokens and a vocabulary size of 66K
- To assess the performance of the acceptability measures, BLL compute Pearson's *r* against mean human ratings.
- BLL also experimented with Spearman's rank correlation, but found similar trends, and so they present only the Pearson results.

Training and Evaluation of the LMs

- BLL train `tdlm` and `lstm` on a sample of 100K English Wikipedia articles, which has no overlap with the 100 documents used for test set annotation.
- The training data has approximately 40M tokens and a vocabulary size of 66K
- To assess the performance of the acceptability measures, BLL compute Pearson's *r* against mean human ratings.
- BLL also experimented with Spearman's rank correlation, but found similar trends, and so they present only the Pearson results.

Model Performance on the Prediction Task

Rtg	Model	*LP*	*Mean*	*NrmD*	*SLOR*
$\texttt{human}^{-\textit{context}}$	$\texttt{lstm}^{-c}$	0.151	0.487	**0.586**	0.584
	$\texttt{lstm}^{+c}$	0.161	0.529	0.618	**0.633**
	$\texttt{tdlm}^{-c}$	0.147	0.515	0.634	**0.640**
	$\texttt{tdlm}^{+c}$	0.165	0.541	0.645	**0.653**
$\texttt{human}^{+\textit{context}}$	$\texttt{lstm}^{-c}$	0.153	0.421	0.494	**0.503**
	$\texttt{lstm}^{+c}$	0.168	0.459	0.522	**0.546**
	$\texttt{tdlm}^{-c}$	0.153	0.450	0.541	**0.557**
	$\texttt{tdlm}^{+c}$	0.169	0.473	0.552	**0.568**

Discussion of the Models' Performance

- lstm^{-c} against human$^{-context}$ with SLOR achieves 0.584, slightly surpassing the performance of the RNN with SLOR in the original LCL experiment (0.570).
- Across all models (lstm and tdlm) and human ratings (human$^{-context}$ and human$^{+context}$), using context at test time improves model performance.
- Taking context into account helps in modelling acceptability, regardless of whether it is tested against judgments made with (human$^{+context}$) or without context (human$^{-context}$).
- tdlm consistently outperforms lstm over both types of human ratings and test input variants.
- Context helps in the modelling of acceptability, whether it is incorporated during training (lstm vs. tdlm) or at test time (lstm^{-c}/tdlm^{-c} vs. lstm^{+c}/tdlm^{+c}).

Discussion of the Models' Performance

- `lstm`$^{-c}$ against `human`$^{-context}$ with SLOR achieves 0.584, slightly surpassing the performance of the RNN with SLOR in the original LCL experiment (0.570).
- Across all models (`lstm` and `tdlm`) and human ratings (`human`$^{-context}$ and `human`$^{+context}$), using context at test time improves model performance.
- Taking context into account helps in modelling acceptability, regardless of whether it is tested against judgments made with (`human`$^{+context}$) or without context (`human`$^{-context}$).
- `tdlm` consistently outperforms `lstm` over both types of human ratings and test input variants.
- Context helps in the modelling of acceptability, whether it is incorporated during training (`lstm` vs. `tdlm`) or at test time (`lstm`$^{-c}$/`tdlm`$^{-c}$ vs. `lstm`$^{+c}$/`tdlm`$^{+c}$).

Discussion of the Models' Performance

- `lstm`$^{-c}$ against `human`$^{-context}$ with SLOR achieves 0.584, slightly surpassing the performance of the RNN with SLOR in the original LCL experiment (0.570).
- Across all models (`lstm` and `tdlm`) and human ratings (`human`$^{-context}$ and `human`$^{+context}$), using context at test time improves model performance.
- Taking context into account helps in modelling acceptability, regardless of whether it is tested against judgments made with (`human`$^{+context}$) or without context (`human`$^{-context}$).
- `tdlm` consistently outperforms `lstm` over both types of human ratings and test input variants.
- Context helps in the modelling of acceptability, whether it is incorporated during training (`lstm` vs. `tdlm`) or at test time (`lstm`$^{-c}$/`tdlm`$^{-c}$ vs. `lstm`$^{+c}$/`tdlm`$^{+c}$).

Discussion of the Models' Performance

- `lstm`$^{-c}$ against `human`$^{-context}$ with SLOR achieves 0.584, slightly surpassing the performance of the RNN with SLOR in the original LCL experiment (0.570).
- Across all models (`lstm` and `tdlm`) and human ratings (`human`$^{-context}$ and `human`$^{+context}$), using context at test time improves model performance.
- Taking context into account helps in modelling acceptability, regardless of whether it is tested against judgments made with (`human`$^{+context}$) or without context (`human`$^{-context}$).
- `tdlm` consistently outperforms `lstm` over both types of human ratings and test input variants.
- Context helps in the modelling of acceptability, whether it is incorporated during training (`lstm` vs. `tdlm`) or at test time (`lstm`$^{-c}$/`tdlm`$^{-c}$ vs. `lstm`$^{+c}$/`tdlm`$^{+c}$).

Discussion of the Models' Performance

- $\texttt{lstm}^{-c}$ against $\texttt{human}^{-context}$ with SLOR achieves 0.584, slightly surpassing the performance of the RNN with SLOR in the original LCL experiment (0.570).
- Across all models ($\texttt{lstm}$ and $\texttt{tdlm}$) and human ratings ($\texttt{human}^{-context}$ and $\texttt{human}^{+context}$), using context at test time improves model performance.
- Taking context into account helps in modelling acceptability, regardless of whether it is tested against judgments made with ($\texttt{human}^{+context}$) or without context ($\texttt{human}^{-context}$).
- $\texttt{tdlm}$ consistently outperforms $\texttt{lstm}$ over both types of human ratings and test input variants.
- Context helps in the modelling of acceptability, whether it is incorporated during training ($\texttt{lstm}$ vs. $\texttt{tdlm}$) or at test time ($\texttt{lstm}^{-c}$/$\texttt{tdlm}^{-c}$ vs. $\texttt{lstm}^{+c}$/$\texttt{tdlm}^{+c}$).

The Models' In-Context Predictions

- The *SLOR* correlation of $\texttt{lstm}^{+c}$/$\texttt{tdlm}^{+c}$ vs. $\texttt{human}^{+context}$ (0.546/568) is lower than that of $\texttt{lstm}^{-c}$/$\texttt{tdlm}^{-c}$ vs. $\texttt{human}^{-context}$ (0.584/0.640).

- $\texttt{human}^{+context}$ ratings are more difficult to predict than $\texttt{human}^{-context}$.

The Models' In-Context Predictions

- The *SLOR* correlation of $\texttt{lstm}^{+c}$/$\texttt{tdlm}^{+c}$ vs. $\texttt{human}^{+context}$ (0.546/568) is lower than that of $\texttt{lstm}^{-c}$/$\texttt{tdlm}^{-c}$ vs. $\texttt{human}^{-context}$ (0.584/0.640).

- $\texttt{human}^{+context}$ ratings are more difficult to predict than $\texttt{human}^{-context}$.

One Explanation: Discourse Coherence

- But the question remains as to why context reduces the spread between ratings.
- One possible explanation is that annotators focus more on discourse coherence when rating sentences in a document context.
- The issue of discourse coherence does not arise in $\texttt{human}^{-context}$ judgments.
- If this factor is, in fact, significant in annotation, then syntactic infelicities introduced by round-trip MT may play less of a role in rating for the $\texttt{human}^{+context}$ set.

One Explanation: Discourse Coherence

- But the question remains as to why context reduces the spread between ratings.
- One possible explanation is that annotators focus more on discourse coherence when rating sentences in a document context.
- The issue of discourse coherence does not arise in $\texttt{human}^{-context}$ judgments.
- If this factor is, in fact, significant in annotation, then syntactic infelicities introduced by round-trip MT may play less of a role in rating for the $\texttt{human}^{+context}$ set.

One Explanation: Discourse Coherence

- But the question remains as to why context reduces the spread between ratings.
- One possible explanation is that annotators focus more on discourse coherence when rating sentences in a document context.
- The issue of discourse coherence does not arise in $\texttt{human}^{-context}$ judgments.
- If this factor is, in fact, significant in annotation, then syntactic infelicities introduced by round-trip MT may play less of a role in rating for the $\texttt{human}^{+context}$ set.

One Explanation: Discourse Coherence

- But the question remains as to why context reduces the spread between ratings.
- One possible explanation is that annotators focus more on discourse coherence when rating sentences in a document context.
- The issue of discourse coherence does not arise in $\texttt{human}^{-context}$ judgments.
- If this factor is, in fact, significant in annotation, then syntactic infelicities introduced by round-trip MT may play less of a role in rating for the $\texttt{human}^{+context}$ set.

A Second Explanation: General Cognitive Load

- A second explanation is that context imposes additional cognitive load (Sweller, 1988; Ito et al., 2018; Causse et al., 2016; Park et al., 2013), which reduces the speaker's/hearer's resources for identifying syntactic and semantic anomaly in an individual sentence.
- If the discourse coherence account is correct, then we would expect the compression effect to be prominent with coherent contexts, but not with random contexts, which prevent integration of the sentence into a discourse unit.
- By contrast, the general cognitive load explanation predicts that the compression effect should be observable for both types of context, as each of them causes distraction through use of additional processing resources.

A Second Explanation: General Cognitive Load

- A second explanation is that context imposes additional cognitive load (Sweller, 1988; Ito et al., 2018; Causse et al., 2016; Park et al., 2013), which reduces the speaker's/hearer's resources for identifying syntactic and semantic anomaly in an individual sentence.
- If the discourse coherence account is correct, then we would expect the compression effect to be prominent with coherent contexts, but not with random contexts, which prevent integration of the sentence into a discourse unit.
- By contrast, the general cognitive load explanation predicts that the compression effect should be observable for both types of context, as each of them causes distraction through use of additional processing resources.

A Second Explanation: General Cognitive Load

- A second explanation is that context imposes additional cognitive load (Sweller, 1988; Ito et al., 2018; Causse et al., 2016; Park et al., 2013), which reduces the speaker's/hearer's resources for identifying syntactic and semantic anomaly in an individual sentence.
- If the discourse coherence account is correct, then we would expect the compression effect to be prominent with coherent contexts, but not with random contexts, which prevent integration of the sentence into a discourse unit.
- By contrast, the general cognitive load explanation predicts that the compression effect should be observable for both types of context, as each of them causes distraction through use of additional processing resources.

A New Set of Context Experiments

- Following BLL's protocol, Lau, Armendariz, Lappin, Purver, and Shu (2020) (LALPS) generate a test set of 250 sentences from 50 English Wikipedia sentences, through round trip MT, with Moses.
- They split the test set into 25 HITs of 10 sentences.
- Each HIT contains 2 original English sentences and 8 translated sentences, which are different from each other and not derived from either of the originals.
- LALPS use AMT crowd sourcing to annotate the sentences for naturalness on a four point scale, for three types of context.

A New Set of Context Experiments

- Following BLL's protocol, Lau, Armendariz, Lappin, Purver, and Shu (2020) (LALPS) generate a test set of 250 sentences from 50 English Wikipedia sentences, through round trip MT, with Moses.
- They split the test set into 25 HITs of 10 sentences.
- Each HIT contains 2 original English sentences and 8 translated sentences, which are different from each other and not derived from either of the originals.
- LALPS use AMT crowd sourcing to annotate the sentences for naturalness on a four point scale, for three types of context.

A New Set of Context Experiments

- Following BLL's protocol, Lau, Armendariz, Lappin, Purver, and Shu (2020) (LALPS) generate a test set of 250 sentences from 50 English Wikipedia sentences, through round trip MT, with Moses.
- They split the test set into 25 HITs of 10 sentences.
- Each HIT contains 2 original English sentences and 8 translated sentences, which are different from each other and not derived from either of the originals.
- LALPS use AMT crowd sourcing to annotate the sentences for naturalness on a four point scale, for three types of context.

A New Set of Context Experiments

- Following BLL's protocol, Lau, Armendariz, Lappin, Purver, and Shu (2020) (LALPS) generate a test set of 250 sentences from 50 English Wikipedia sentences, through round trip MT, with Moses.
- They split the test set into 25 HITs of 10 sentences.
- Each HIT contains 2 original English sentences and 8 translated sentences, which are different from each other and not derived from either of the originals.
- LALPS use AMT crowd sourcing to annotate the sentences for naturalness on a four point scale, for three types of context.

Null, Real, and Random, Contexts

- LALPS presented the sentences in each HIT in null, real, and, random contexts, respectively.
- Each context experiment was performed by a disjoint group of annotators.
- The real contexts consist of the three sentences that immediately precede a sentence in its document.
- The random contexts are consecutive sequences of three sentences taken from other documents.

Null, Real, and Random, Contexts

- LALPS presented the sentences in each HIT in null, real, and, random contexts, respectively.
- Each context experiment was performed by a disjoint group of annotators.
- The real contexts consist of the three sentences that immediately precede a sentence in its document.
- The random contexts are consecutive sequences of three sentences taken from other documents.

Null, Real, and Random, Contexts

- LALPS presented the sentences in each HIT in null, real, and, random contexts, respectively.
- Each context experiment was performed by a disjoint group of annotators.
- The real contexts consist of the three sentences that immediately precede a sentence in its document.
- The random contexts are consecutive sequences of three sentences taken from other documents.

Null, Real, and Random, Contexts

- LALPS presented the sentences in each HIT in null, real, and, random contexts, respectively.
- Each context experiment was performed by a disjoint group of annotators.
- The real contexts consist of the three sentences that immediately precede a sentence in its document.
- The random contexts are consecutive sequences of three sentences taken from other documents.

A Topic Identification Task

- In the context experiments LALPS first show the context paragraph, and they ask users to select the most appropriate description of its topic from a list of 4 candidate topics.
- Each candidate topic is represented by three words generated with a topic model.
- After performing this task the annotator is shown the sentence to be rated for acceptability.
- This experimental set up insures that annotators read the context sentences before assessing the sentences of the HIT.

A Topic Identification Task

- In the context experiments LALPS first show the context paragraph, and they ask users to select the most appropriate description of its topic from a list of 4 candidate topics.
- Each candidate topic is represented by three words generated with a topic model.
- After performing this task the annotator is shown the sentence to be rated for acceptability.
- This experimental set up insures that annotators read the context sentences before assessing the sentences of the HIT.

A Topic Identification Task

- In the context experiments LALPS first show the context paragraph, and they ask users to select the most appropriate description of its topic from a list of 4 candidate topics.
- Each candidate topic is represented by three words generated with a topic model.
- After performing this task the annotator is shown the sentence to be rated for acceptability.
- This experimental set up insures that annotators read the context sentences before assessing the sentences of the HIT.

A Topic Identification Task

- In the context experiments LALPS first show the context paragraph, and they ask users to select the most appropriate description of its topic from a list of 4 candidate topics.
- Each candidate topic is represented by three words generated with a topic model.
- After performing this task the annotator is shown the sentence to be rated for acceptability.
- This experimental set up insures that annotators read the context sentences before assessing the sentences of the HIT.

Modulating the Annotations

- LALPS follow Hill et al.'s (2015) procedure for modulating the mean annotation results to filter out the effect of outlier judgments.
- They calculate the average rating for each user, and the overall average by taking the mean of all average ratings.
- LALPS decrease (increase) the ratings of a user by 1.0 if their average rating is greater (smaller) than the overall average by 1.0.
- To reduce the impact of outliers, for each sentence they remove ratings that are more than 2 standard deviations away from the mean.

Modulating the Annotations

- LALPS follow Hill et al.'s (2015) procedure for modulating the mean annotation results to filter out the effect of outlier judgments.
- They calculate the average rating for each user, and the overall average by taking the mean of all average ratings.
- LALPS decrease (increase) the ratings of a user by 1.0 if their average rating is greater (smaller) than the overall average by 1.0.
- To reduce the impact of outliers, for each sentence they remove ratings that are more than 2 standard deviations away from the mean.

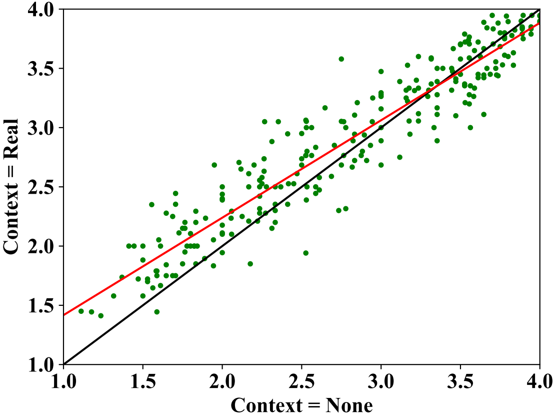
Modulating the Annotations

- LALPS follow Hill et al.'s (2015) procedure for modulating the mean annotation results to filter out the effect of outlier judgments.
- They calculate the average rating for each user, and the overall average by taking the mean of all average ratings.
- LALPS decrease (increase) the ratings of a user by 1.0 if their average rating is greater (smaller) than the overall average by 1.0.
- To reduce the impact of outliers, for each sentence they remove ratings that are more than 2 standard deviations away from the mean.

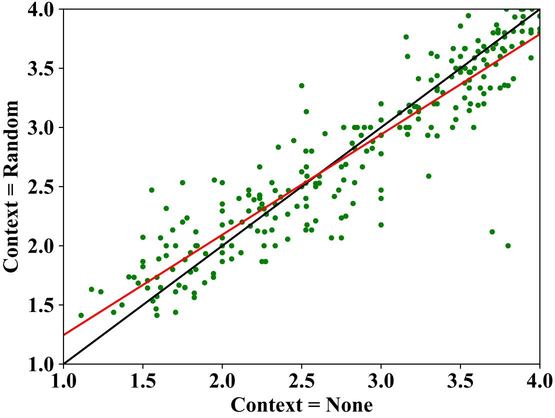
Modulating the Annotations

- LALPS follow Hill et al.'s (2015) procedure for modulating the mean annotation results to filter out the effect of outlier judgments.
- They calculate the average rating for each user, and the overall average by taking the mean of all average ratings.
- LALPS decrease (increase) the ratings of a user by 1.0 if their average rating is greater (smaller) than the overall average by 1.0.
- To reduce the impact of outliers, for each sentence they remove ratings that are more than 2 standard deviations away from the mean.

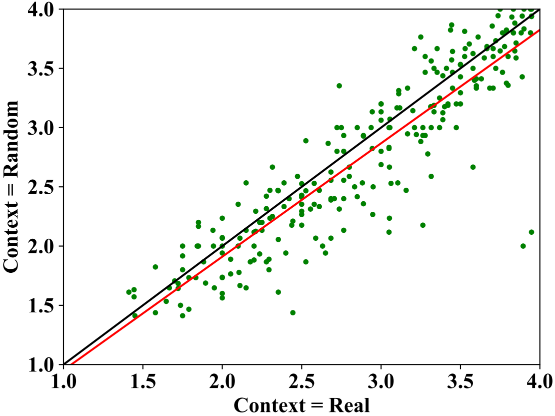
Human Acceptability Judgments: Real Contexts vs No Contexts



Human Acceptability Judgments: Random Contexts vs No Contexts



Human Acceptability Judgments: Random Contexts vs Real Contexts



Regression to the Mean

- At ACL 2020 Roger Levy pointed out to us that when linear regression is applied to two sets of very noisy data, it will yield a distributional pattern that resembles the compression effect exhibited in our regression graphs.
- To determine whether this effect is an artefact of regression to the mean in our annotations, Lau applied total least square errors-in-variables regression to the data.
- He also used this procedure with a swap of the dependent and independent variables, which involves permuting the x and y axes.
- The resulting graphs closely resemble our original linear regression patterns for the corresponding variable pairs, with mirror image graphs for their permuted variants.
- This result strongly indicates that the compression effect is a real property of the data, rather than an epiphenomenon caused by regression to the mean.

Regression to the Mean

- At ACL 2020 Roger Levy pointed out to us that when linear regression is applied to two sets of very noisy data, it will yield a distributional pattern that resembles the compression effect exhibited in our regression graphs.
- To determine whether this effect is an artefact of regression to the mean in our annotations, Lau applied total least square errors-in-variables regression to the data.
- He also used this procedure with a swap of the dependent and independent variables, which involves permuting the x and y axes.
- The resulting graphs closely resemble our original linear regression patterns for the corresponding variable pairs, with mirror image graphs for their permuted variants.
- This result strongly indicates that the compression effect is a real property of the data, rather than an epiphenomenon caused by regression to the mean.

Regression to the Mean

- At ACL 2020 Roger Levy pointed out to us that when linear regression is applied to two sets of very noisy data, it will yield a distributional pattern that resembles the compression effect exhibited in our regression graphs.
- To determine whether this effect is an artefact of regression to the mean in our annotations, Lau applied total least square errors-in-variables regression to the data.
- He also used this procedure with a swap of the dependent and independent variables, which involves permuting the x and y axes.
- The resulting graphs closely resemble our original linear regression patterns for the corresponding variable pairs, with mirror image graphs for their permuted variants.
- This result strongly indicates that the compression effect is a real property of the data, rather than an epiphenomenon caused by regression to the mean.

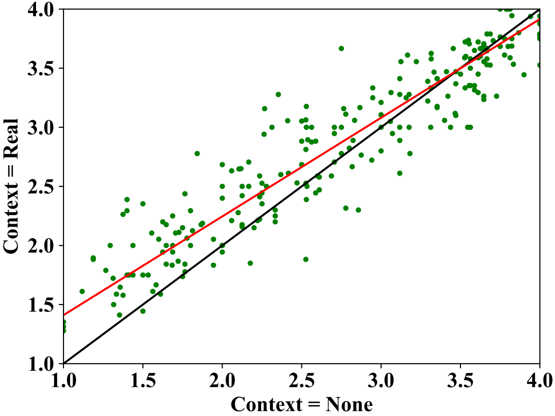
Regression to the Mean

- At ACL 2020 Roger Levy pointed out to us that when linear regression is applied to two sets of very noisy data, it will yield a distributional pattern that resembles the compression effect exhibited in our regression graphs.
- To determine whether this effect is an artefact of regression to the mean in our annotations, Lau applied total least square errors-in-variables regression to the data.
- He also used this procedure with a swap of the dependent and independent variables, which involves permuting the x and y axes.
- The resulting graphs closely resemble our original linear regression patterns for the corresponding variable pairs, with mirror image graphs for their permuted variants.
- This result strongly indicates that the compression effect is a real property of the data, rather than an epiphenomenon caused by regression to the mean.

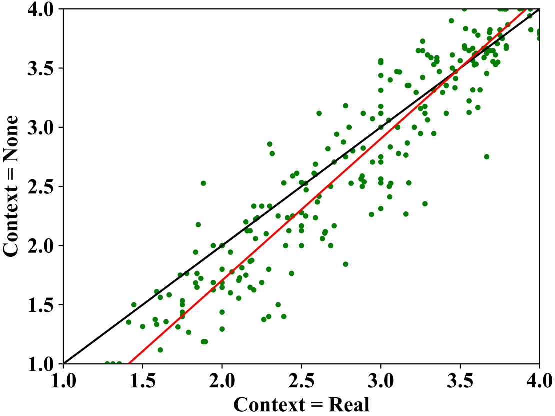
Regression to the Mean

- At ACL 2020 Roger Levy pointed out to us that when linear regression is applied to two sets of very noisy data, it will yield a distributional pattern that resembles the compression effect exhibited in our regression graphs.
- To determine whether this effect is an artefact of regression to the mean in our annotations, Lau applied total least square errors-in-variables regression to the data.
- He also used this procedure with a swap of the dependent and independent variables, which involves permuting the x and y axes.
- The resulting graphs closely resemble our original linear regression patterns for the corresponding variable pairs, with mirror image graphs for their permuted variants.
- This result strongly indicates that the compression effect is a real property of the data, rather than an epiphenomenon caused by regression to the mean.

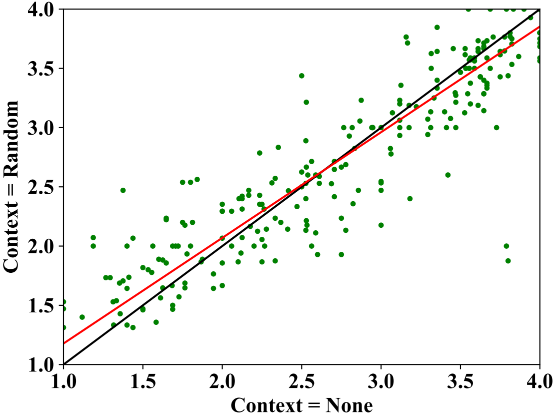
Total Least Squares: Real Contexts vs No Contexts



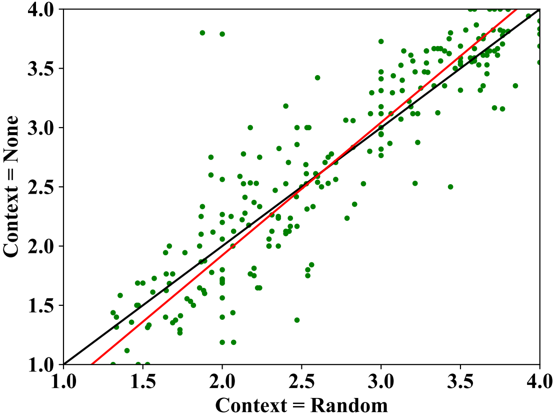
Total Least Squares: No Contexts vs Real Contexts



Total Least Squares: Random Contexts vs No Contexts



Total Least Squares: No Contexts vs Random Contexts



Explaining the Compression and Raising Effects

- The compression effect appears in both the h^{+} (real context) vs. $\mathrm{h}^{\varnothing}$ (null context), and the h^{-} (random context) vs. $\mathrm{h}^{\varnothing}$ cases.
- In addition, the h^{+} vs. $\mathrm{h}^{\varnothing}$ regression diagram exhibits a raising effect in real contexts, which pushes the cross over point towards the upper end of the scale.
- In the h^{-} vs. h^{+} figure the regression line is parallel to and below the diagonal, indicating a consistent decrease in acceptability ratings from h^{+} to h^{-}.
- These effects suggest that the cognitive load of processing contexts produces compression in both h^{+} and h^{-}, while discourse coherence operates only in h^{+} to generate a raising of acceptability ratings.

Explaining the Compression and Raising Effects

- The compression effect appears in both the h^{+} (real context) vs. $\mathrm{h}^{\varnothing}$ (null context), and the h^{-} (random context) vs. $\mathrm{h}^{\varnothing}$ cases.
- In addition, the h^{+} vs. $\mathrm{h}^{\varnothing}$ regression diagram exhibits a raising effect in real contexts, which pushes the cross over point towards the upper end of the scale.
- In the h^{-} vs. h^{+} figure the regression line is parallel to and below the diagonal, indicating a consistent decrease in acceptability ratings from h^{+} to h^{-}.
- These effects suggest that the cognitive load of processing contexts produces compression in both h^{+} and h^{-}, while discourse coherence operates only in h^{+} to generate a raising of acceptability ratings.

Explaining the Compression and Raising Effects

- The compression effect appears in both the $\mathtt{h}^{+}$ (real context) vs. $\mathtt{h}^{\varnothing}$ (null context), and the $\mathtt{h}^{-}$ (random context) vs. $\mathtt{h}^{\varnothing}$ cases.
- In addition, the $\mathtt{h}^{+}$ vs. $\mathtt{h}^{\varnothing}$ regression diagram exhibits a raising effect in real contexts, which pushes the cross over point towards the upper end of the scale.
- In the $\mathtt{h}^{-}$ vs. $\mathtt{h}^{+}$ figure the regression line is parallel to and below the diagonal, indicating a consistent decrease in acceptability ratings from $\mathtt{h}^{+}$ to $\mathtt{h}^{-}$.
- These effects suggest that the cognitive load of processing contexts produces compression in both $\mathtt{h}^{+}$ and $\mathtt{h}^{-}$, while discourse coherence operates only in $\mathtt{h}^{+}$ to generate a raising of acceptability ratings.

Explaining the Compression and Raising Effects

- The compression effect appears in both the $\mathtt{h}^+$ (real context) vs. $\mathtt{h}^\varnothing$ (null context), and the $\mathtt{h}^-$ (random context) vs. $\mathtt{h}^\varnothing$ cases.
- In addition, the $\mathtt{h}^+$ vs. $\mathtt{h}^\varnothing$ regression diagram exhibits a raising effect in real contexts, which pushes the cross over point towards the upper end of the scale.
- In the $\mathtt{h}^-$ vs. $\mathtt{h}^+$ figure the regression line is parallel to and below the diagonal, indicating a consistent decrease in acceptability ratings from $\mathtt{h}^+$ to $\mathtt{h}^-$.
- These effects suggest that the cognitive load of processing contexts produces compression in both $\mathtt{h}^+$ and $\mathtt{h}^-$, while discourse coherence operates only in $\mathtt{h}^+$ to generate a raising of acceptability ratings.

Statistical Significance of the Compression and Discourse Coherence Effects I

- The mean ratings in all three test sets correlate strongly with each other, with Pearson's *r* for h^{+} vs. $\mathrm{h}^{\varnothing} = 0.945$, h^{-} vs. $\mathrm{h}^{\varnothing} = 0.917$, and h^{-} vs. $\mathrm{h}^{+} = 0.901$.
- LALPS use the non-parametric Wilcoxon signed-rank test (one-tailed) to compare the difference between h^{+} and h^{-}.
- The test gives a *p*-value of 2.4×10^{-8}, indicating that the discourse coherence effect is significant.

Statistical Significance of the Compression and Discourse Coherence Effects I

- The mean ratings in all three test sets correlate strongly with each other, with Pearson's *r* for h^{+} vs. $\mathrm{h}^{\varnothing} = 0.945$, h^{-} vs. $\mathrm{h}^{\varnothing} = 0.917$, and h^{-} vs. $\mathrm{h}^{+} = 0.901$.
- LALPS use the non-parametric Wilcoxon signed-rank test (one-tailed) to compare the difference between h^{+} and h^{-}.
- The test gives a *p*-value of 2.4×10^{-8}, indicating that the discourse coherence effect is significant.

Statistical Significance of the Compression and Discourse Coherence Effects I

- The mean ratings in all three test sets correlate strongly with each other, with Pearson's *r* for h^+ vs. $\mathrm{h}^\varnothing = 0.945$, h^- vs. $\mathrm{h}^\varnothing = 0.917$, and h^- vs. $\mathrm{h}^+ = 0.901$.
- LALPS use the non-parametric Wilcoxon signed-rank test (one-tailed) to compare the difference between h^+ and h^-.
- The test gives a *p*-value of 2.4×10^{-8}, indicating that the discourse coherence effect is significant.

Statistical Significance of the Compression and Discourse Coherence Effects II

- LALPS also use the Wilcoxon test to compare the regression lines for $\mathtt{h}^{+}$ vs. $\mathtt{h}^{\varnothing}$, and $\mathtt{h}^{-}$ vs. $\mathtt{h}^{\varnothing}$, to see if their offsets (constants) and slopes (coefficients) are statistically different.
- The *p*-value for the offset is 2.1×10^{-2}, confirming that there is a significant discourse coherence effect.
- The *p*-value for the slope, however, is 3.9×10^{-1}, suggesting that cognitive load compresses the ratings in a consistent way for both $\mathtt{h}^{+}$ and $\mathtt{h}^{-}$, relative to $\mathtt{h}^{\varnothing}$.

Statistical Significance of the Compression and Discourse Coherence Effects II

- LALPS also use the Wilcoxon test to compare the regression lines for $\mathtt{h}^{+}$ vs. $\mathtt{h}^{\varnothing}$, and $\mathtt{h}^{-}$ vs. $\mathtt{h}^{\varnothing}$, to see if their offsets (constants) and slopes (coefficients) are statistically different.
- The *p*-value for the offset is 2.1×10^{-2}, confirming that there is a significant discourse coherence effect.
- The *p*-value for the slope, however, is 3.9×10^{-1}, suggesting that cognitive load compresses the ratings in a consistent way for both $\mathtt{h}^{+}$ and $\mathtt{h}^{-}$, relative to $\mathtt{h}^{\varnothing}$.

Statistical Significance of the Compression and Discourse Coherence Effects II

- LALPS also use the Wilcoxon test to compare the regression lines for $\mathtt{h}^{+}$ vs. $\mathtt{h}^{\varnothing}$, and $\mathtt{h}^{-}$ vs. $\mathtt{h}^{\varnothing}$, to see if their offsets (constants) and slopes (coefficients) are statistically different.
- The *p*-value for the offset is 2.1×10^{-2}, confirming that there is a significant discourse coherence effect.
- The *p*-value for the slope, however, is 3.9×10^{-1}, suggesting that cognitive load compresses the ratings in a consistent way for both $\mathtt{h}^{+}$ and $\mathtt{h}^{-}$, relative to $\mathtt{h}^{\varnothing}$.

Predicting Sentence Acceptability with Deep Neural Language Models

- In addition to `lstm` and `tdlm` LALPS experiment with three transformer language models.
- These are `gpt2` (Radford et al., 2019), `bert` (Devlin et al., 2019), and `xlnet` (Yang et al., 2019).
- These models are equipped with large pre-trained lexical embeddings, and they apply multiple self-attention heads to all input words.
- `bert` processes input strings without regard to sequence, in a massively parallel way, which permits it to efficiently identify large numbers of co-occurrence dependency patterns among the words of a string.

Predicting Sentence Acceptability with Deep Neural Language Models

- In addition to `lstm` and `tdlm` LALPS experiment with three transformer language models.
- These are `gpt2` (Radford et al., 2019), `bert` (Devlin et al., 2019), and `xlnet` (Yang et al., 2019).
- These models are equipped with large pre-trained lexical embeddings, and they apply multiple self-attention heads to all input words.
- `bert` processes input strings without regard to sequence, in a massively parallel way, which permits it to efficiently identify large numbers of co-occurrence dependency patterns among the words of a string.

Predicting Sentence Acceptability with Deep Neural Language Models

- In addition to `lstm` and `tdlm` LALPS experiment with three transformer language models.
- These are `gpt2` (Radford et al., 2019), `bert` (Devlin et al., 2019), and `xlnet` (Yang et al., 2019).
- These models are equipped with large pre-trained lexical embeddings, and they apply multiple self-attention heads to all input words.
- `bert` processes input strings without regard to sequence, in a massively parallel way, which permits it to efficiently identify large numbers of co-occurrence dependency patterns among the words of a string.

Predicting Sentence Acceptability with Deep Neural Language Models

- In addition to `lstm` and `tdlm` LALPS experiment with three transformer language models.
- These are `gpt2` (Radford et al., 2019), `bert` (Devlin et al., 2019), and `xlnet` (Yang et al., 2019).
- These models are equipped with large pre-trained lexical embeddings, and they apply multiple self-attention heads to all input words.
- `bert` processes input strings without regard to sequence, in a massively parallel way, which permits it to efficiently identify large numbers of co-occurrence dependency patterns among the words of a string.

Sentence Probabilities for Transformers

- `lstm` and `gpt2` are unidirectional, and so they can be used to compute the probability of a sentence left to right, according to the formula $\overrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i | w_{<i})$.
- `bert` is bidirectional, and predicts words for both their left and right contexts.
- It requires the formula $\overleftrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i | w_{<i}, w_{>i})$.
- This equation does not yield true probabilities, as its values do not to sum to 1 (normalising these values to genuine probabilities is intractable).
- Instead these values provide confidence scores of likelihood.
- `xlnet` can be applied either unidirectionally or bidirectionally.

Sentence Probabilities for Transformers

- `lstm` and `gpt2` are unidirectional, and so they can be used to compute the probability of a sentence left to right, according to the formula $\overrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i | w_{<i})$.
- `bert` is bidirectional, and predicts words for both their left and right contexts.
- It requires the formula $\overleftrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i | w_{<i}, w_{>i})$.
- This equation does not yield true probabilities, as its values do not to sum to 1 (normalising these values to genuine probabilities is intractable).
- Instead these values provide confidence scores of likelihood.
- `xlnet` can be applied either unidirectionally or bidirectionally.

Sentence Probabilities for Transformers

- `lstm` and `gpt2` are unidirectional, and so they can be used to compute the probability of a sentence left to right, according to the formula $\overrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i|w_{<i})$.
- `bert` is bidirectional, and predicts words for both their left and right contexts.
- It requires the formula $\overleftrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i|w_{<i}, w_{>i})$.
- This equation does not yield true probabilities, as its values do not to sum to 1 (normalising these values to genuine probabilities is intractable).
- Instead these values provide confidence scores of likelihood.
- `xlnet` can be applied either unidirectionally or bidirectionally.

Sentence Probabilities for Transformers

- `lstm` and `gpt2` are unidirectional, and so they can be used to compute the probability of a sentence left to right, according to the formula $\overrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i|w_{<i})$.
- `bert` is bidirectional, and predicts words for both their left and right contexts.
- It requires the formula $\overleftrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i|w_{<i}, w_{>i})$.
- This equation does not yield true probabilities, as its values do not to sum to 1 (normalising these values to genuine probabilities is intractable).
- Instead these values provide confidence scores of likelihood.
- `xlnet` can be applied either unidirectionally or bidirectionally.

Sentence Probabilities for Transformers

- `lstm` and `gpt2` are unidirectional, and so they can be used to compute the probability of a sentence left to right, according to the formula $\overrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i | w_{<i})$.
- `bert` is bidirectional, and predicts words for both their left and right contexts.
- It requires the formula $\overleftrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i | w_{<i}, w_{>i})$.
- This equation does not yield true probabilities, as its values do not to sum to 1 (normalising these values to genuine probabilities is intractable).
- Instead these values provide confidence scores of likelihood.
- `xlnet` can be applied either unidirectionally or bidirectionally.

Sentence Probabilities for Transformers

- `lstm` and `gpt2` are unidirectional, and so they can be used to compute the probability of a sentence left to right, according to the formula $\overrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i|w_{<i})$.
- `bert` is bidirectional, and predicts words for both their left and right contexts.
- It requires the formula $\overleftrightarrow{P}(s) = \prod_{i=0}^{|s|} P(w_i|w_{<i}, w_{>i})$.
- This equation does not yield true probabilities, as its values do not to sum to 1 (normalising these values to genuine probabilities is intractable).
- Instead these values provide confidence scores of likelihood.
- `xlnet` can be applied either unidirectionally or bidirectionally.

Language Model Architectures

Model	Configuration			Training Data			
	Architecture	Encoding	#Param.	Casing	Size	Tokenisation	Corpora
lstm	RNN	Unidir.	60M	Uncased	0.2GB	Word	Wikipedia
tdlm	RNN	Unidir.	80M	Uncased	0.2GB	Word	Wikipedia
gpt2	Transformer	Unidir.	340M	Cased	40GB	BPE	WebText
$\texttt{bert}_{\texttt{cs}}$	Transformer	Bidir.	340M	Cased	13GB	WordPiece	Wikipedia, BookCorpus
$\texttt{bert}_{\texttt{ucs}}$	Transformer	Bidir.	340M	Uncased	13GB	WordPiece	Wikipedia, BookCorpus
xlnet	Transformer	Hybrid	340M	Cased	126GB	Sentence-Piece	Wikipedia, BookCorpus, Giga5 ClueWeb, Common Crawl

Acceptability Scoring Measures

Acc. Measure	Equation
LogProb	$\log P_m(s)$
Mean LP	$\dfrac{\log P_m(s)}{\lvert s \rvert}$
PenLP	$\dfrac{\log P_m(s)}{((5+\lvert s \rvert)/(5+1))^{\alpha}}$
NormLP	$-\dfrac{\log P_m(s)}{\log P_u(s)}$
SLOR	$\dfrac{\log P_m(s) - \log P_u(s)}{\lvert s \rvert}$

$P(s)$ is the sentence probability, computed using either the uni-prob or bi-prob formula, depending on the model, $P_u(s)$ is the sentence probability estimated by a unigram language model, and $\alpha = 0.8$.

Upper Bounds on Model Performance

- LALPS compute two human performance estimates to serve as upper bounds on the accuracy of a model.
- ub_1 is LCL's one-vs-rest annotator correlation, where LALPS select a random annotator's rating, and compare it to the mean rating of the rest, using Pearson's r.
- They repeat this for a large number of trials (1000) to get a robust estimate of the mean correlation.
- ub_2 is a half-vs-half annotator correlation, where for each sentence they randomly split the annotators into two groups, and compare the mean ratings between the groups.

Upper Bounds on Model Performance

- LALPS compute two human performance estimates to serve as upper bounds on the accuracy of a model.
- $\mathtt{ub}_1$ is LCL's one-vs-rest annotator correlation, where LALPS select a random annotator's rating, and compare it to the mean rating of the rest, using Pearson's *r*.
- They repeat this for a large number of trials (1000) to get a robust estimate of the mean correlation.
- $\mathtt{ub}_2$ is a half-vs-half annotator correlation, where for each sentence they randomly split the annotators into two groups, and compare the mean ratings between the groups.

Upper Bounds on Model Performance

- LALPS compute two human performance estimates to serve as upper bounds on the accuracy of a model.
- $\mathtt{ub}_1$ is LCL's one-vs-rest annotator correlation, where LALPS select a random annotator's rating, and compare it to the mean rating of the rest, using Pearson's *r*.
- They repeat this for a large number of trials (1000) to get a robust estimate of the mean correlation.
- $\mathtt{ub}_2$ is a half-vs-half annotator correlation, where for each sentence they randomly split the annotators into two groups, and compare the mean ratings between the groups.

Upper Bounds on Model Performance

- LALPS compute two human performance estimates to serve as upper bounds on the accuracy of a model.
- $\mathtt{ub}_1$ is LCL's one-vs-rest annotator correlation, where LALPS select a random annotator's rating, and compare it to the mean rating of the rest, using Pearson's *r*.
- They repeat this for a large number of trials (1000) to get a robust estimate of the mean correlation.
- $\mathtt{ub}_2$ is a half-vs-half annotator correlation, where for each sentence they randomly split the annotators into two groups, and compare the mean ratings between the groups.

Performance Filtered and Unfiltered for Outliers

- LALPS present model performance for the annotation sets in which outlier ratings ($\geq$ 2 standard deviation) have been removed.
- This filtering does not significantly affect the model accuracy scores, but it does increase the simulated human upper bound correlations.
- For completeness they present the upper bound correlations for both outlier filtered (ub_1,ub_2) and outlier unfiltered ($\mathrm{ub}_1^{\varnothing}$,$\mathrm{ub}_2^{\varnothing}$) test sets.

Performance Filtered and Unfiltered for Outliers

- LALPS present model performance for the annotation sets in which outlier ratings ($\geq$ 2 standard deviation) have been removed.
- This filtering does not significantly affect the model accuracy scores, but it does increase the simulated human upper bound correlations.
- For completeness they present the upper bound correlations for both outlier filtered (ub_1,ub_2) and outlier unfiltered ($\mathrm{ub}_1^{\varnothing}$,$\mathrm{ub}_2^{\varnothing}$) test sets.

Performance Filtered and Unfiltered for Outliers

- LALPS present model performance for the annotation sets in which outlier ratings ($\geq$ 2 standard deviation) have been removed.
- This filtering does not significantly affect the model accuracy scores, but it does increase the simulated human upper bound correlations.
- For completeness they present the upper bound correlations for both outlier filtered ($\mathtt{ub}_1$,$\mathtt{ub}_2$) and outlier unfiltered ($\mathtt{ub}_1^{\varnothing}$,$\mathtt{ub}_2^{\varnothing}$) test sets.

Model Performance: Null Context

Rtg Encod.		Model	*LogProb*	*Mean LP*	*PenLP*	*NormLP*	*SLOR*
$h^{\varnothing}$	Unidir.	$\texttt{lstm}^{\varnothing}$	0.29	0.42	0.42	0.52	**0.53**
		$\texttt{lstm}^{+}$	0.30	0.49	0.45	0.61	**0.63**
		$\texttt{tdlm}^{\varnothing}$	0.30	0.49	0.45	0.60	**0.61**
		$\texttt{tdlm}^{+}$	0.30	0.50	0.45	0.59	**0.60**
		$\texttt{gpt2}^{\varnothing}$	0.33	0.34	**0.56**	0.38	0.38
		$\texttt{gpt2}^{+}$	0.38	0.59	0.58	**0.63**	0.60
		$\texttt{xlnet}^{\varnothing}_{\texttt{uni}}$	0.31	0.42	0.51	0.51	**0.52**
		$\texttt{xlnet}^{+}_{\texttt{uni}}$	0.36	0.56	0.55	0.61	**0.61**
	Bidir.	$\texttt{bert}^{\varnothing}_{\texttt{cs}}$	0.51	0.54	**0.63**	0.55	0.53
		$\texttt{bert}^{+}_{\texttt{cs}}$	0.53	0.63	**0.67**	0.64	0.60
		$\texttt{bert}^{\varnothing}_{\texttt{ucs}}$	0.59	0.63	**0.70**	0.63	0.60
		$\texttt{bert}^{+}_{\texttt{ucs}}$	0.60	0.68	**0.72**	0.67	0.63
		$\texttt{xlnet}^{\varnothing}_{\texttt{bi}}$	0.52	0.51	**0.66**	0.53	0.53
		$\texttt{xlnet}^{+}_{\texttt{bi}}$	0.57	0.65	**0.73**	0.66	0.65
	—	$\texttt{ub}_1$/ $\texttt{ub}^{\varnothing}_1$			0.75 / 0.66		
		$\texttt{ub}_2$/ $\texttt{ub}^{\varnothing}_2$			0.92 / 0.88		

Model Performance: Real Context

Rtg	Encod.	Model	*LogProb*	*Mean LP*	*PenLP*	*NormLP*	*SLOR*
h^{+}	Unidir.	$\texttt{lstm}^{\varnothing}$	0.29	0.44	0.43	**0.52**	**0.52**
		$\texttt{lstm}^{+}$	0.31	0.51	0.46	**0.62**	**0.62**
		$\texttt{tdlm}^{\varnothing}$	0.30	0.50	0.45	**0.59**	**0.59**
		$\texttt{tdlm}^{+}$	0.30	0.50	0.46	**0.58**	**0.58**
		$\texttt{gpt2}^{\varnothing}$	0.32	0.33	**0.56**	0.36	0.37
		$\texttt{gpt2}^{+}$	0.38	0.60	0.59	**0.63**	0.60
		$\texttt{xlnet}^{\varnothing}_{\texttt{uni}}$	0.30	0.42	0.50	0.49	**0.51**
		$\texttt{xlnet}^{+}_{\texttt{uni}}$	0.35	0.56	0.55	**0.60**	**0.61**
	Bidir.	$\texttt{bert}^{\varnothing}_{\texttt{cs}}$	0.49	0.53	**0.62**	0.54	0.51
		$\texttt{bert}^{+}_{\texttt{cs}}$	0.52	0.63	**0.66**	0.63	0.58
		$\texttt{bert}^{\varnothing}_{\texttt{ucs}}$	0.58	0.63	**0.70**	0.63	0.60
		$\texttt{bert}^{+}_{\texttt{ucs}}$	0.60	0.68	**0.73**	0.67	0.63
		$\texttt{xlnet}^{\varnothing}_{\texttt{bi}}$	0.51	0.50	**0.65**	0.52	0.53
		$\texttt{xlnet}^{+}_{\texttt{bi}}$	0.57	0.65	**0.74**	0.65	0.65
	—	$\texttt{ub}_1$/ $\texttt{ub}_1^{\varnothing}$			0.73 / 0.66		
		$\texttt{ub}_2$/ $\texttt{ub}_2^{\varnothing}$			0.92 / 0.89		

Model Performance: Random Context

Rtg	Encod.	Model	*LogProb*	*Mean LP*	*PenLP*	*NormLP*	*SLOR*
h^-	Unidir.	$\texttt{lstm}^{\varnothing}$	0.28	0.44	0.43	**0.50**	**0.50**
		$\texttt{lstm}^{-}$	0.27	0.41	0.40	**0.47**	**0.47**
		$\texttt{tdlm}^{\varnothing}$	0.29	0.52	0.46	**0.59**	0.58
		$\texttt{tdlm}^{-}$	0.28	0.49	0.44	**0.56**	0.55
		$\texttt{gpt2}^{\varnothing}$	0.32	0.34	**0.55**	0.35	0.35
		$\texttt{gpt2}^{-}$	0.30	0.42	**0.51**	0.44	0.41
		$\texttt{xlnet}^{\varnothing}_{\texttt{uni}}$	0.30	0.44	**0.51**	0.49	0.49
		$\texttt{xlnet}^{-}_{\texttt{uni}}$	0.29	0.40	**0.49**	0.46	0.46
	Bidir.	$\texttt{bert}^{\varnothing}_{\texttt{cs}}$	0.48	0.53	**0.62**	0.53	0.49
		$\texttt{bert}^{-}_{\texttt{cs}}$	0.49	0.52	**0.61**	0.51	0.47
		$\texttt{bert}^{\varnothing}_{\texttt{ucs}}$	0.56	0.61	**0.68**	0.60	0.56
		$\texttt{bert}^{-}_{\texttt{ucs}}$	0.56	0.58	**0.66**	0.57	0.53
		$\texttt{xlnet}^{\varnothing}_{\texttt{bi}}$	0.49	0.48	**0.62**	0.49	0.48
		$\texttt{xlnet}^{-}_{\texttt{bi}}$	0.50	0.51	**0.64**	0.51	0.50
	—	$\texttt{ub}_1$/ $\texttt{ub}^{\varnothing}_1$			0.75 / 0.68		
		$\texttt{ub}_2$/ $\texttt{ub}^{\varnothing}_2$			0.92 / 0.88		

Modelling Experiment Results

- The bidirectional models significantly outperform the unidirectional models across all three context types, when *PenLP*, rather than *SLOR* is the scoring function.
- This suggests that large lexical embeddings and bidirectional context training render normalisation by word frequency unnecessary.
- Model architecture rather than size is the decisive factor governing performance.
- $\texttt{bert}_{\texttt{ucs}}$ approaches estimated individual human performance, as specified by $\texttt{ub}_1$, and surpasses it for $\texttt{ub}_1^{\varnothing}$, on the the prediction of sentence acceptability task.

Modelling Experiment Results

- The bidirectional models significantly outperform the unidirectional models across all three context types, when *PenLP*, rather than *SLOR* is the scoring function.
- This suggests that large lexical embeddings and bidirectional context training render normalisation by word frequency unnecessary.
- Model architecture rather than size is the decisive factor governing performance.
- $\texttt{bert}_{\texttt{ucs}}$ approaches estimated individual human performance, as specified by $\texttt{ub}_1$, and surpasses it for $\texttt{ub}_1^{\varnothing}$, on the the prediction of sentence acceptability task.

Modelling Experiment Results

- The bidirectional models significantly outperform the unidirectional models across all three context types, when *PenLP*, rather than *SLOR* is the scoring function.
- This suggests that large lexical embeddings and bidirectional context training render normalisation by word frequency unnecessary.
- Model architecture rather than size is the decisive factor governing performance.
- $\texttt{bert}_{\texttt{ucs}}$ approaches estimated individual human performance, as specified by $\texttt{ub}_1$, and surpasses it for $\texttt{ub}_1^{\varnothing}$, on the the prediction of sentence acceptability task.

Modelling Experiment Results

- The bidirectional models significantly outperform the unidirectional models across all three context types, when *PenLP*, rather than *SLOR* is the scoring function.
- This suggests that large lexical embeddings and bidirectional context training render normalisation by word frequency unnecessary.
- Model architecture rather than size is the decisive factor governing performance.
- $\texttt{bert}_{\texttt{ucs}}$ approaches estimated individual human performance, as specified by $\texttt{ub}_1$, and surpasses it for $\texttt{ub}_1^{\varnothing}$, on the the prediction of sentence acceptability task.

Controlling for MT Bias in the Test Set

- One might suggest that round trip MT introduces a systematic bias into the types of infelicities that appear in the LALPS test set, which could influence the performance of their models.
- To control for such a possible bias they test the bidirectional transformers, with *PenLP*, on the test set of linguists' examples that LCL extract from Adger's (2003) syntax textbook.
- LCL construct this set by randomly selecting 50 good, and 50 ill-formed English sentences from the full list of examples, and crowd source annotating them for acceptability.

Controlling for MT Bias in the Test Set

- One might suggest that round trip MT introduces a systematic bias into the types of infelicities that appear in the LALPS test set, which could influence the performance of their models.
- To control for such a possible bias they test the bidirectional transformers, with *PenLP*, on the test set of linguists' examples that LCL extract from Adger's (2003) syntax textbook.
- LCL construct this set by randomly selecting 50 good, and 50 ill-formed English sentences from the full list of examples, and crowd source annotating them for acceptability.

Controlling for MT Bias in the Test Set

- One might suggest that round trip MT introduces a systematic bias into the types of infelicities that appear in the LALPS test set, which could influence the performance of their models.
- To control for such a possible bias they test the bidirectional transformers, with *PenLP*, on the test set of linguists' examples that LCL extract from Adger's (2003) syntax textbook.
- LCL construct this set by randomly selecting 50 good, and 50 ill-formed English sentences from the full list of examples, and crowd source annotating them for acceptability.

Bidirectional Transformer Performance on Linguists' Examples

- The three bidirectional model scores, with *PenLP*, are: $\texttt{gpt2}$ = 0.45, $\texttt{bert}_{\texttt{cs}}$ = 0.53, and $\texttt{xlnet}_{\texttt{bi}}$ = 0.58.
- While these scores are lower than those for the round trip MT test sets, they indicate a strong correlation with human judgments.
- It is important to note that they are achieved for an out of domain task.
- The models are trained on naturally occurring text, but they are tested on artificially constructed examples.
- The linguists' examples are, in general, much shorter (less than 7 words) than the sentences in the training corpora.

Bidirectional Transformer Performance on Linguists' Examples

- The three bidirectional model scores, with *PenLP*, are: `gpt2` = 0.45, $\texttt{bert}_{\texttt{cs}}$ = 0.53, and $\texttt{xlnet}_{\texttt{bi}}$ = 0.58.
- While these scores are lower than those for the round trip MT test sets, they indicate a strong correlation with human judgments.
- It is important to note that they are achieved for an out of domain task.
- The models are trained on naturally occurring text, but they are tested on artificially constructed examples.
- The linguists' examples are, in general, much shorter (less than 7 words) than the sentences in the training corpora.

Bidirectional Transformer Performance on Linguists' Examples

- The three bidirectional model scores, with *PenLP*, are: `gpt2` = 0.45, $\texttt{bert}_{\texttt{cs}}$ = 0.53, and $\texttt{xlnet}_{\texttt{bi}}$ = 0.58.
- While these scores are lower than those for the round trip MT test sets, they indicate a strong correlation with human judgments.
- It is important to note that they are achieved for an out of domain task.
- The models are trained on naturally occurring text, but they are tested on artificially constructed examples.
- The linguists' examples are, in general, much shorter (less than 7 words) than the sentences in the training corpora.

Bidirectional Transformer Performance on Linguists' Examples

- The three bidirectional model scores, with *PenLP*, are: `gpt2` = 0.45, $\texttt{bert}_{\texttt{cs}}$ = 0.53, and $\texttt{xlnet}_{\texttt{bi}}$ = 0.58.
- While these scores are lower than those for the round trip MT test sets, they indicate a strong correlation with human judgments.
- It is important to note that they are achieved for an out of domain task.
- The models are trained on naturally occurring text, but they are tested on artificially constructed examples.
- The linguists' examples are, in general, much shorter (less than 7 words) than the sentences in the training corpora.

Bidirectional Transformer Performance on Linguists' Examples

- The three bidirectional model scores, with *PenLP*, are: $\texttt{gpt2}$ = 0.45, $\texttt{bert}_{\texttt{cs}}$ = 0.53, and $\texttt{xlnet}_{\texttt{bi}}$ = 0.58.
- While these scores are lower than those for the round trip MT test sets, they indicate a strong correlation with human judgments.
- It is important to note that they are achieved for an out of domain task.
- The models are trained on naturally occurring text, but they are tested on artificially constructed examples.
- The linguists' examples are, in general, much shorter (less than 7 words) than the sentences in the training corpora.

Conclusions

- Processing context induces a cognitive load for humans, which creates a compression effect on the distribution of acceptability ratings.
- If the context is relevant to the sentence, a discourse coherence effect uniformly boosts sentence acceptability.
- Bidirectional neural language models outperform unidirectional models on the sentence acceptability prediction task.
- The best bidirectional model approaches estimated individual human performance on this task.

Conclusions

- Processing context induces a cognitive load for humans, which creates a compression effect on the distribution of acceptability ratings.
- If the context is relevant to the sentence, a discourse coherence effect uniformly boosts sentence acceptability.
- Bidirectional neural language models outperform unidirectional models on the sentence acceptability prediction task.
- The best bidirectional model approaches estimated individual human performance on this task.

Conclusions

- Processing context induces a cognitive load for humans, which creates a compression effect on the distribution of acceptability ratings.
- If the context is relevant to the sentence, a discourse coherence effect uniformly boosts sentence acceptability.
- Bidirectional neural language models outperform unidirectional models on the sentence acceptability prediction task.
- The best bidirectional model approaches estimated individual human performance on this task.

Conclusions

- Processing context induces a cognitive load for humans, which creates a compression effect on the distribution of acceptability ratings.
- If the context is relevant to the sentence, a discourse coherence effect uniformly boosts sentence acceptability.
- Bidirectional neural language models outperform unidirectional models on the sentence acceptability prediction task.
- The best bidirectional model approaches estimated individual human performance on this task.