Computational Discourse

Automated Essay Scoring with Discourse-Aware Neural Models

Farah Nadeem Huy Nguyen Yang Liu Mari Ostendorf

Karan Praharaj
MS student, CLASIC
University of Colorado Boulder

"Automated essay scoring can be *helpful*, but it is not without *flaws*… some very significant ones."

— Chee Wee Leong (Principal Research Engineer, ETS)

AES with Discourse-Aware Neural Models

An overview of our discussion today:

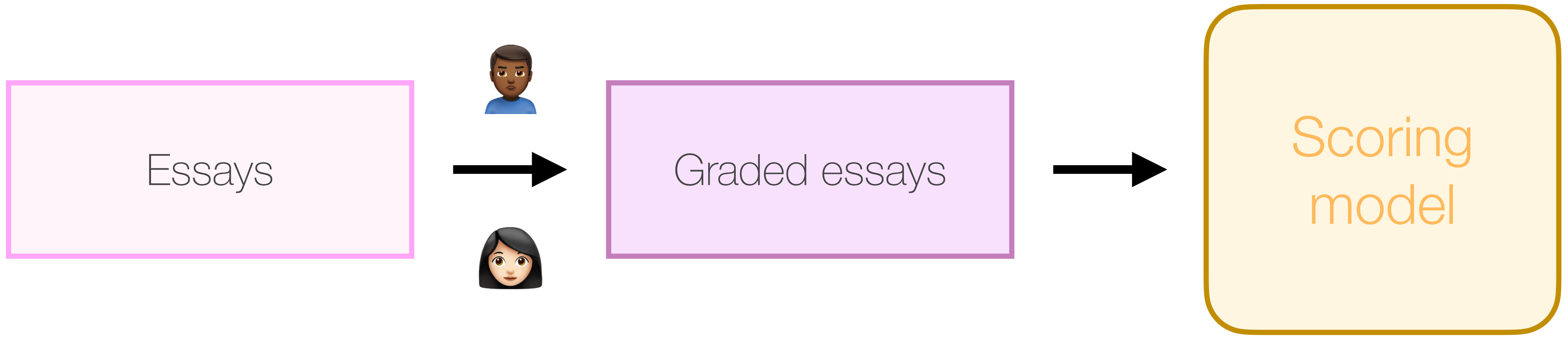
1. background, intro — lay of the land
2. methods — what was done
3. data — what was used
4. results — how well it fares
5. conclusion — what it shows

AES with Discourse-Aware Neural Models

1. background, intro — lay of the land
2. methods — what was done
3. data — what was used
4. results — how well it fares
5. conclusion — what it shows

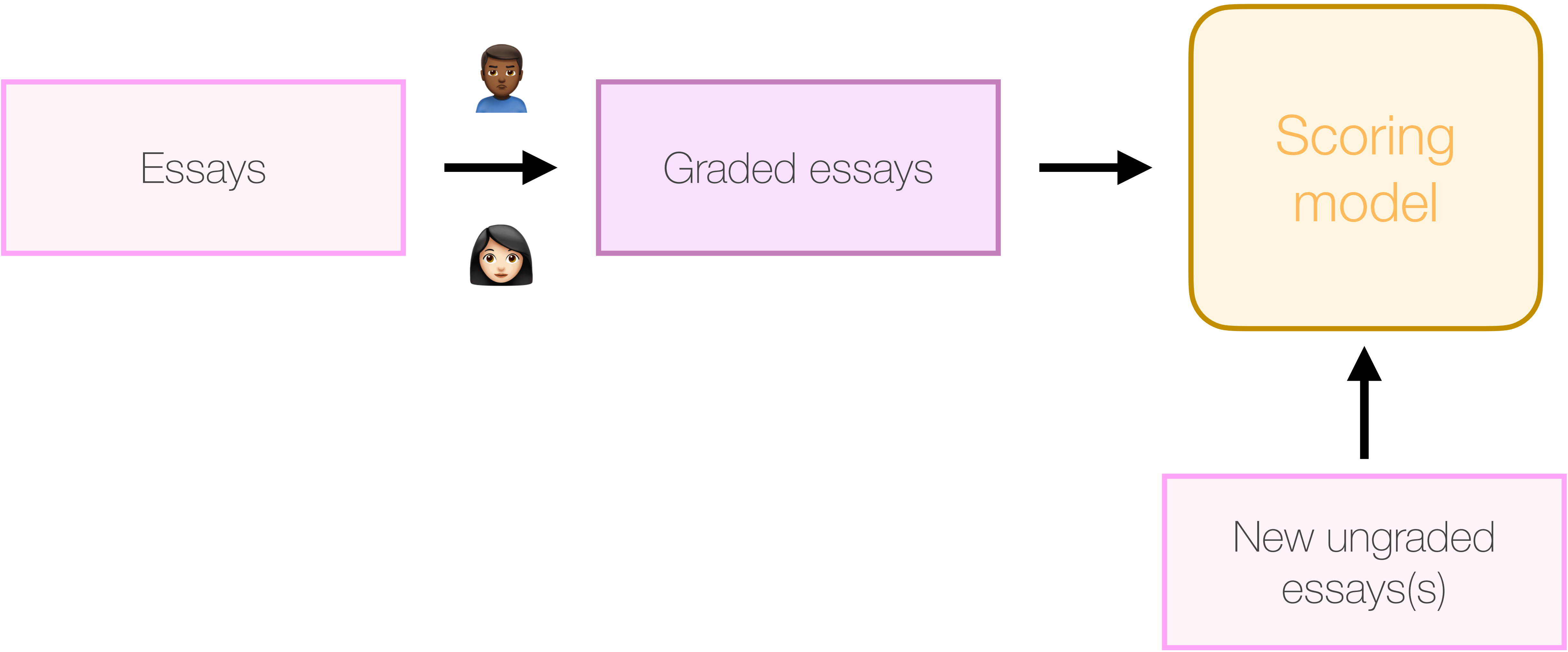
AES - Background and Intro

where we are, where we are headed



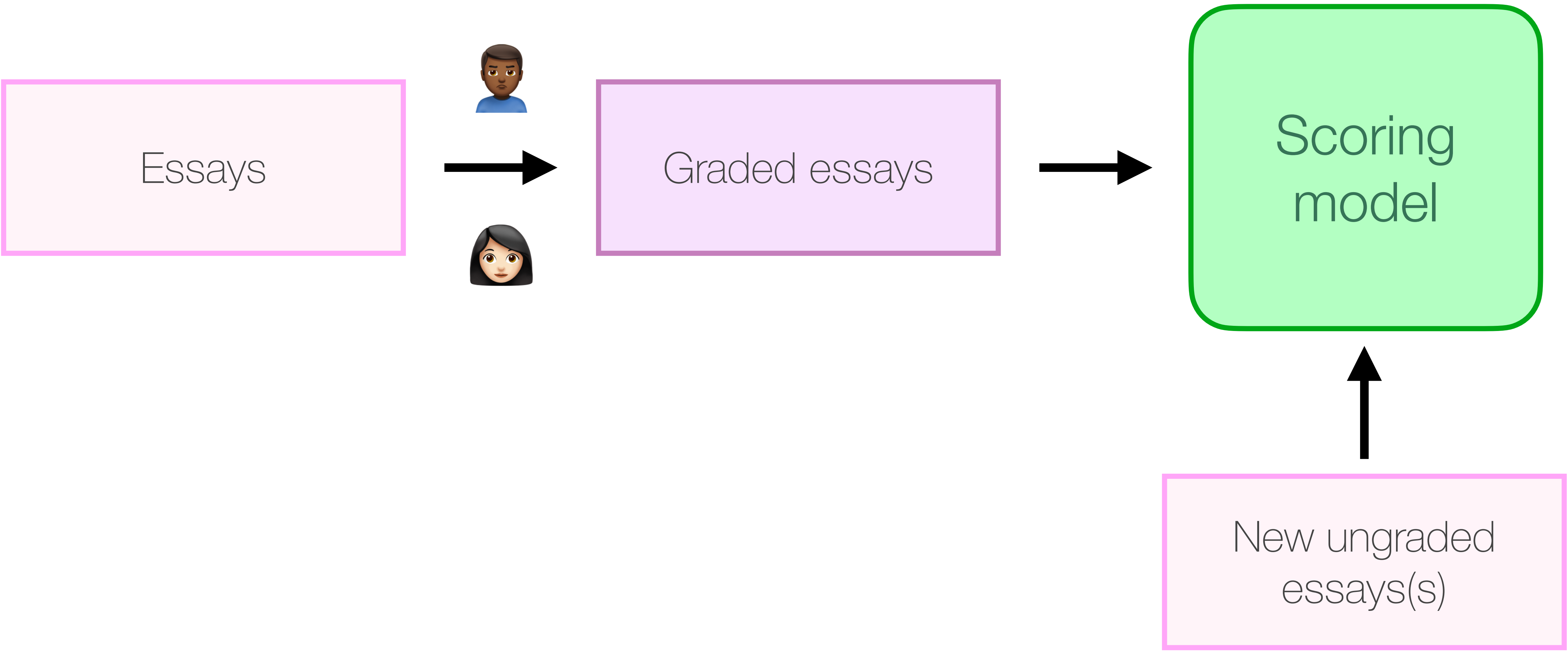
AES - Background and Intro

where we are, where we are headed



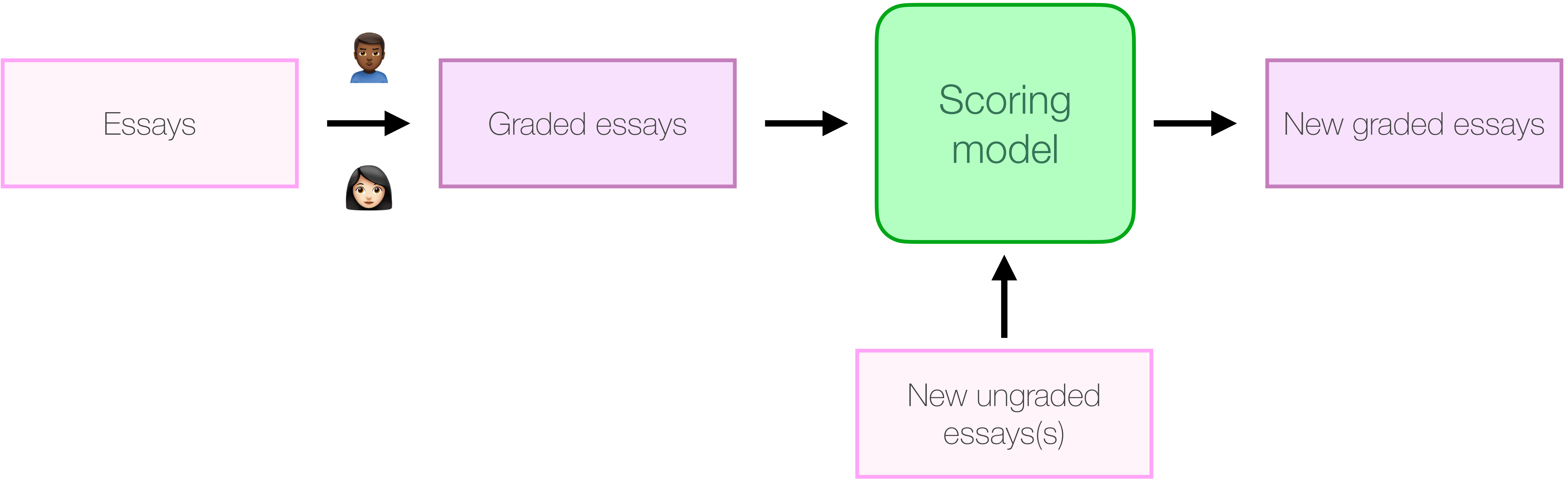
AES - Background and Intro

where we are, where we are headed



AES - Background and Intro

where we are, where we are headed



AES - Background and Intro

where we are, where we are headed

AES systems typically relied on hand-crafted features to predict essay quality.

But…high variability in essay types $\therefore$ not scalable!

Key: adapt to new types, automatic feature generation.

Enter: neural methods.

AES - Background and Intro

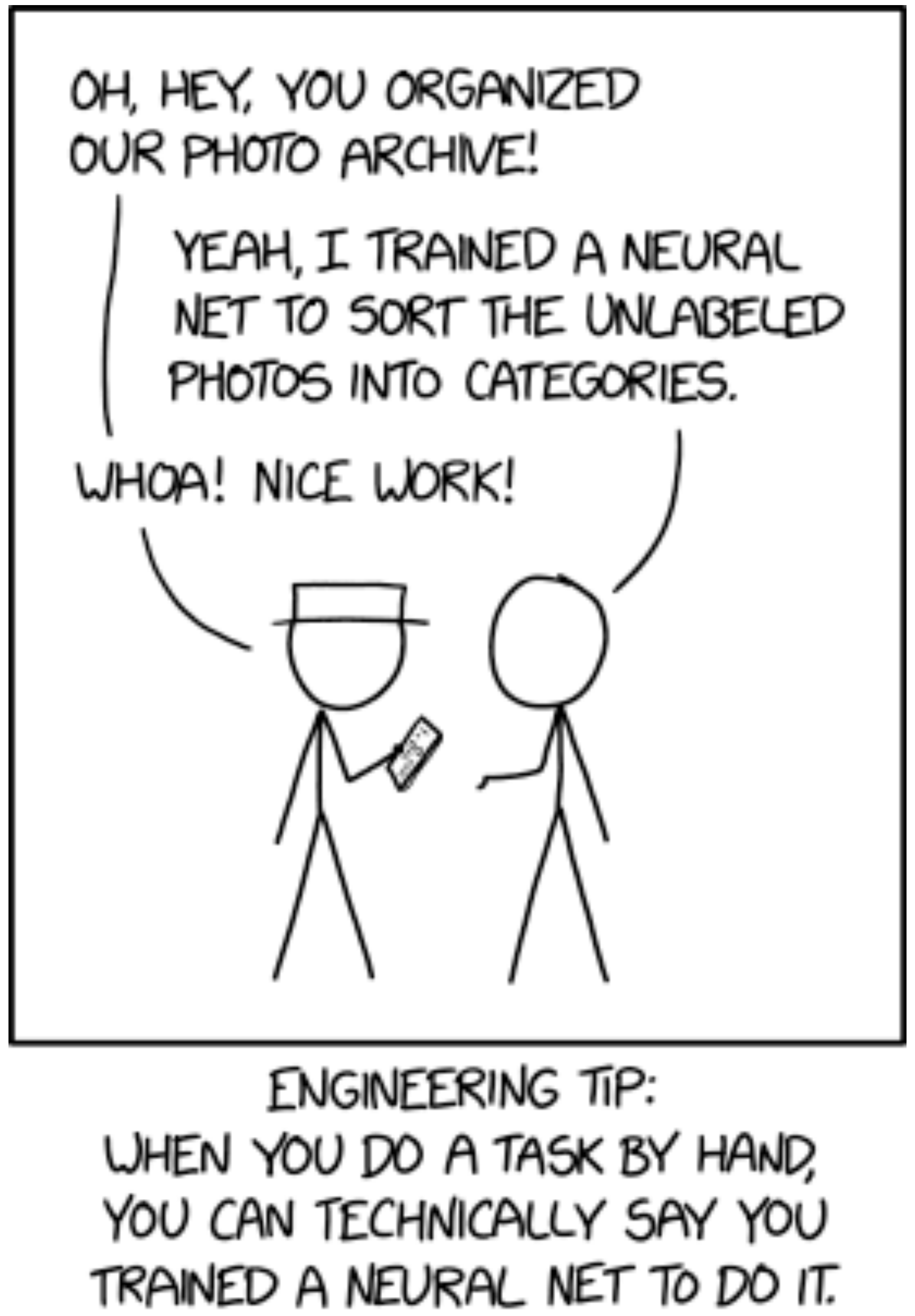
where we are, where we are headed

AES systems typically relied on hand-crafted features to predict essay quality.

But…high variability in essay types ∴ not scalable!

Key: adapt to new types, automatic feature generation.

Enter: neural methods.



AES - Background and Intro

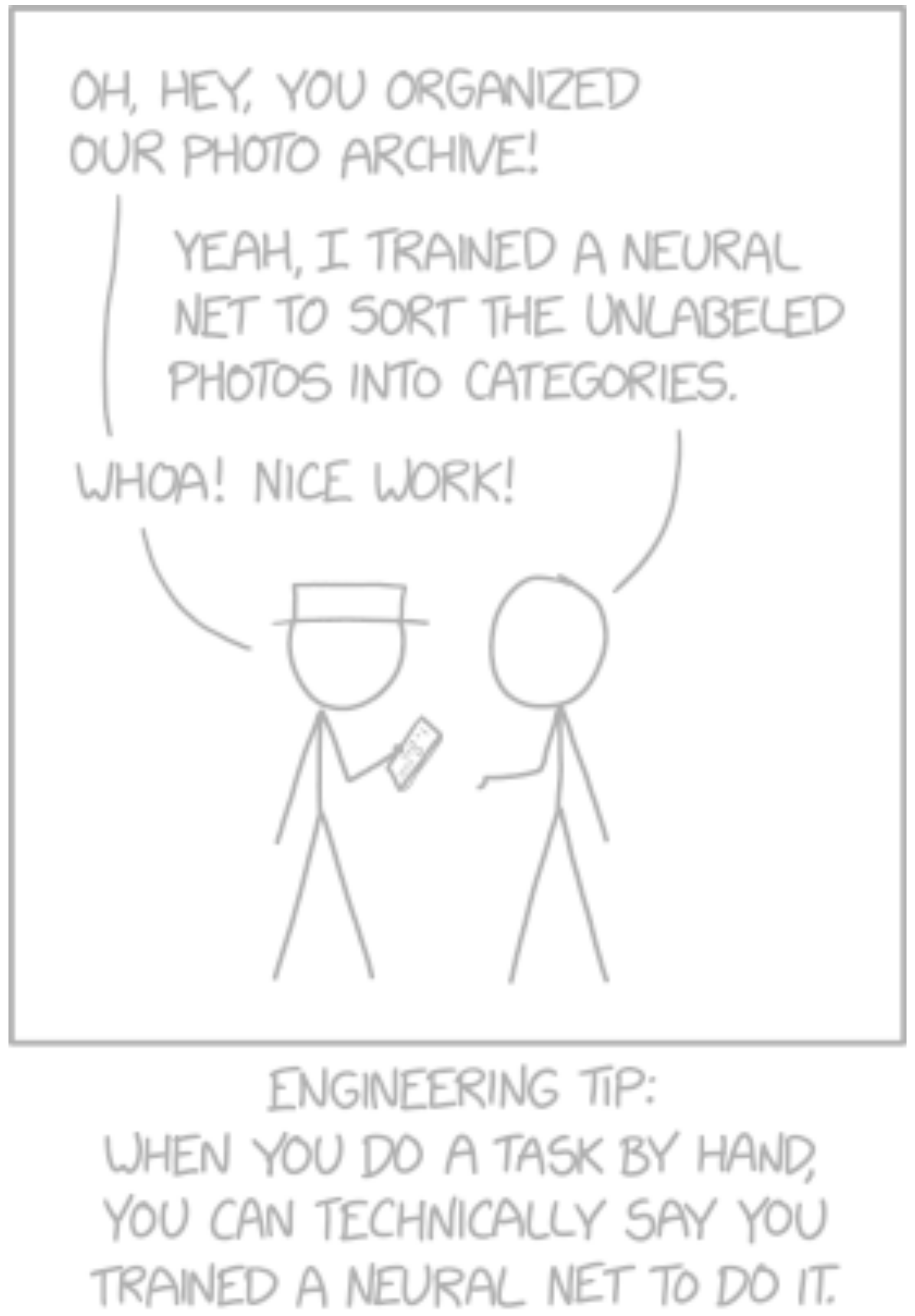
where we are, where we are headed

AES systems typically relied on hand-crafted features to predict essay quality.

But…high variability in essay types $\therefore$ not scalable!

Key: adapt to new types, automatic feature generation.

Enter: neural methods.



not true in our case.

AES - Background and Intro

where we are, where we are headed

This paper makes two contributions:

1. Discourse-aware structures and discourse-related pre-training boost performance.
2. Contextualized embeddings are not useful for tasks with small annotated training sets.

Moral of the story etc. …

Use a combination of neural models and hand-crafted features.

AES with Discourse-Aware Neural Models

1. background, intro — lay of the land
2. methods — what was done
3. data — what was used
4. results — how well it fares
5. conclusion — what it shows

AES with Discourse-Aware Neural Models

1. background, intro — lay of the land
2. methods — what was done
3. data — what was used
4. results — how well it fares
5. conclusion — what it shows

Method

Overall system: Map an essay to a vector. Pass it through for ordinal regression.

Main focus is on two LSTM-based neural models:

1. Hierarchical recurrent network with attention (HAN)
2. Bidirectional context with attention (BCA)

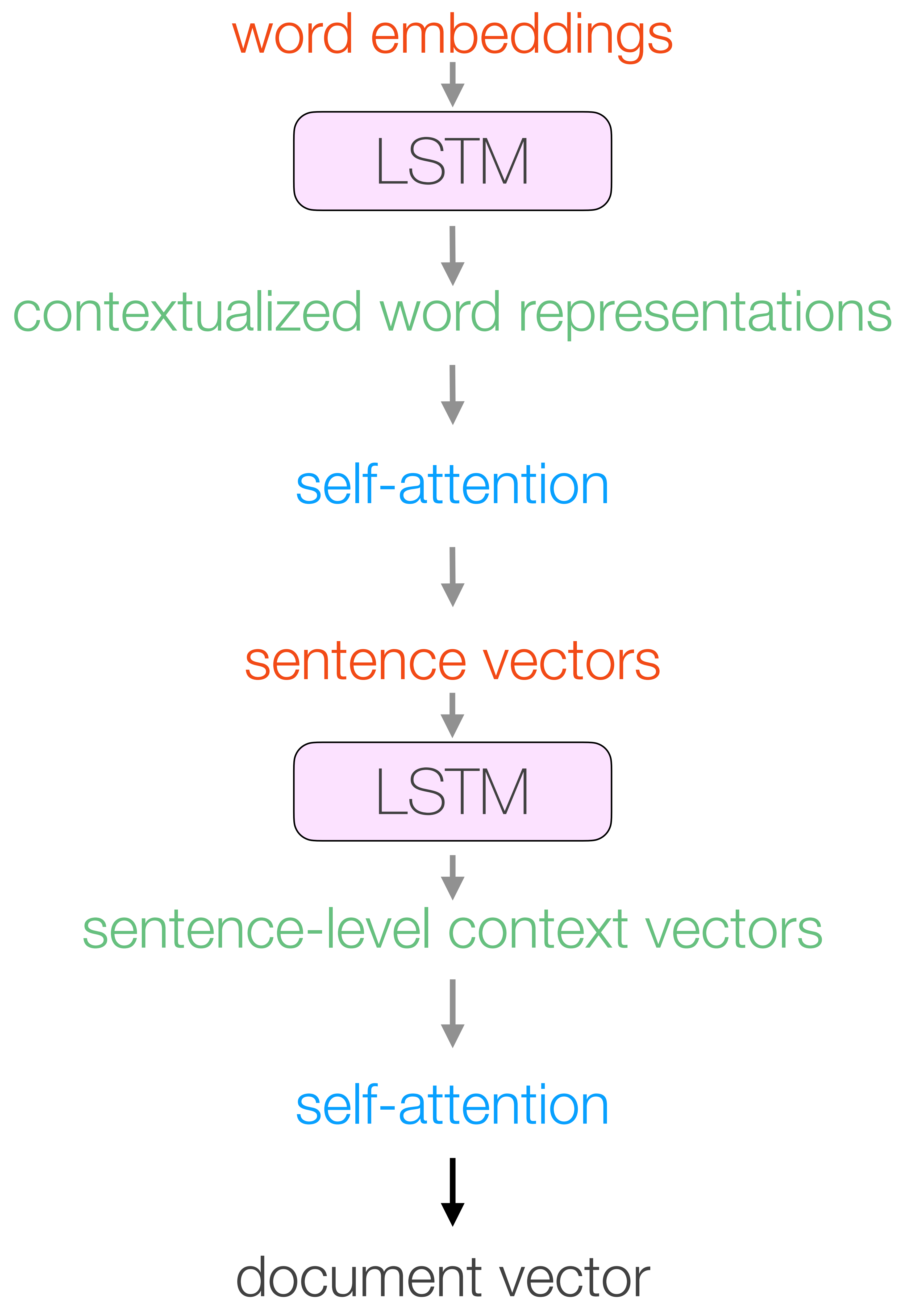
Hierarchical RNN (HAN)

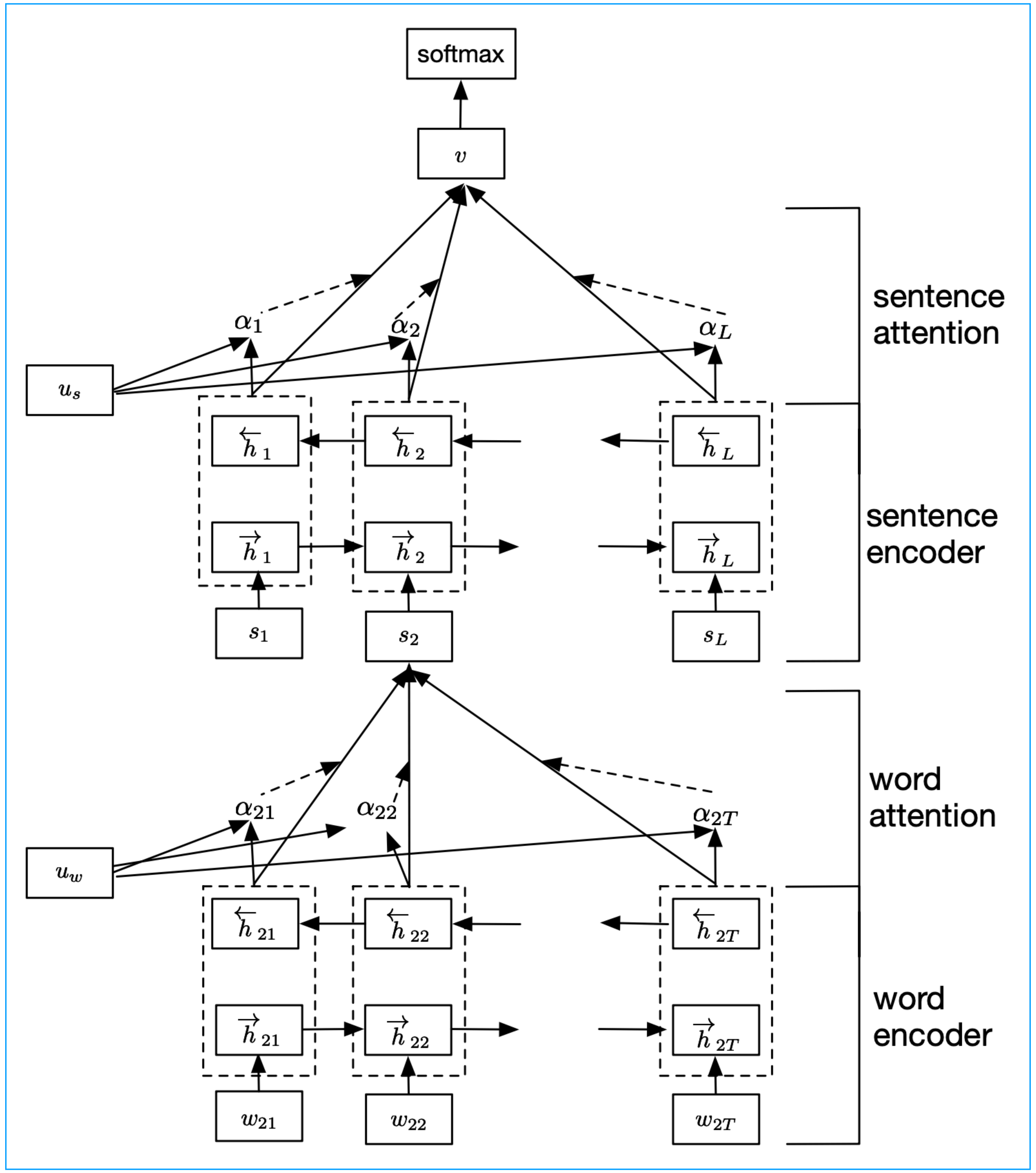
captures the hierarchical structure within a document using two LSTMs.

First layer — input: word embeddings output: contextualized word representations

Second layer — input: sentence vectors output: document vector

All of these elements are woven together by self-attention .


word embeddings
LSTM
contextualized word representations
self-attention
sentence vectors
LSTM
sentence-level context vectors
self-attention
document vector



Hierarchical Attention Network

Bidirectional context with attention (BCA)

Extends HAN to account for cross sentence dependencies.

Incorporates a look-back and look-ahead context vector using output from first LSTM.

Final word representation = LSTM output $\oplus$ look-back $\oplus$ look-ahead

This is used to create sentence vector using attention weights.

Auxiliary Training Tasks

Neural networks can make use of related tasks to improve performance.

This can be done via pre-training.

Pre-training tasks used:

1. Natural language inference (NLI)
2. Discourse marker prediction (DM)

Auxiliary Training Tasks

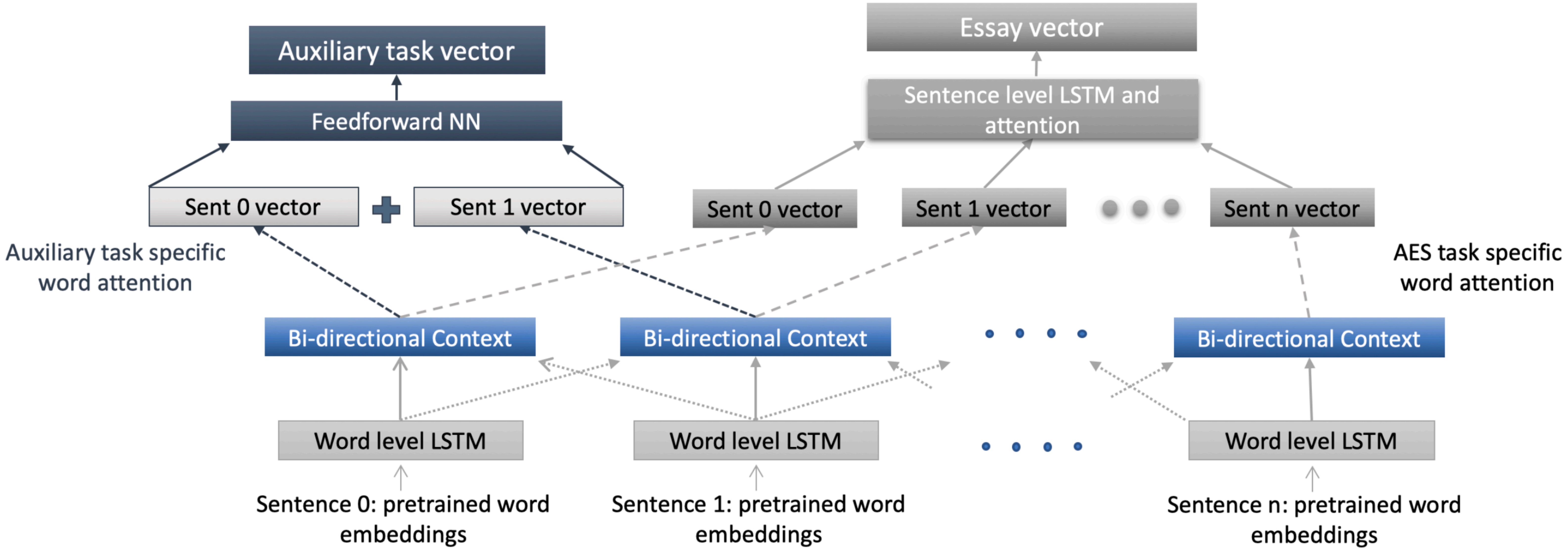
Neural networks can make use of related tasks to improve performance.

This can be done via pre-training.

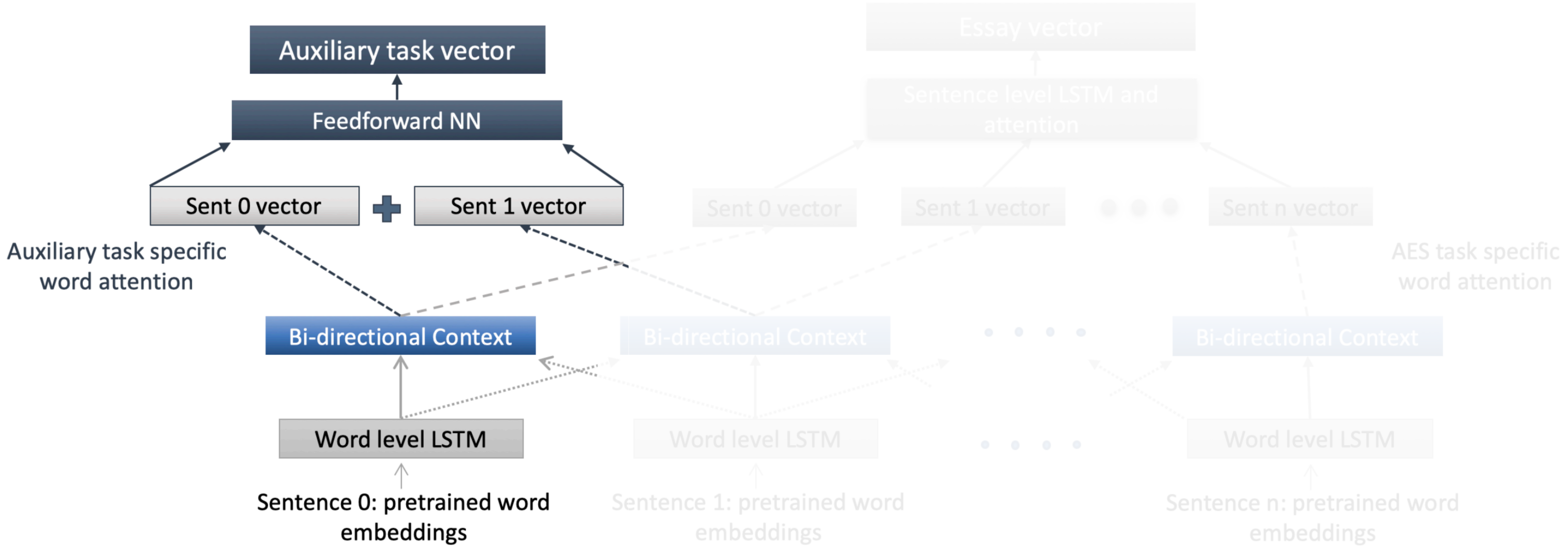
Pre-training tasks used:

1. Natural language inference (NLI)
2. Discourse marker prediction (DM)
3. Contextualized embeddings (MLM, NSP)

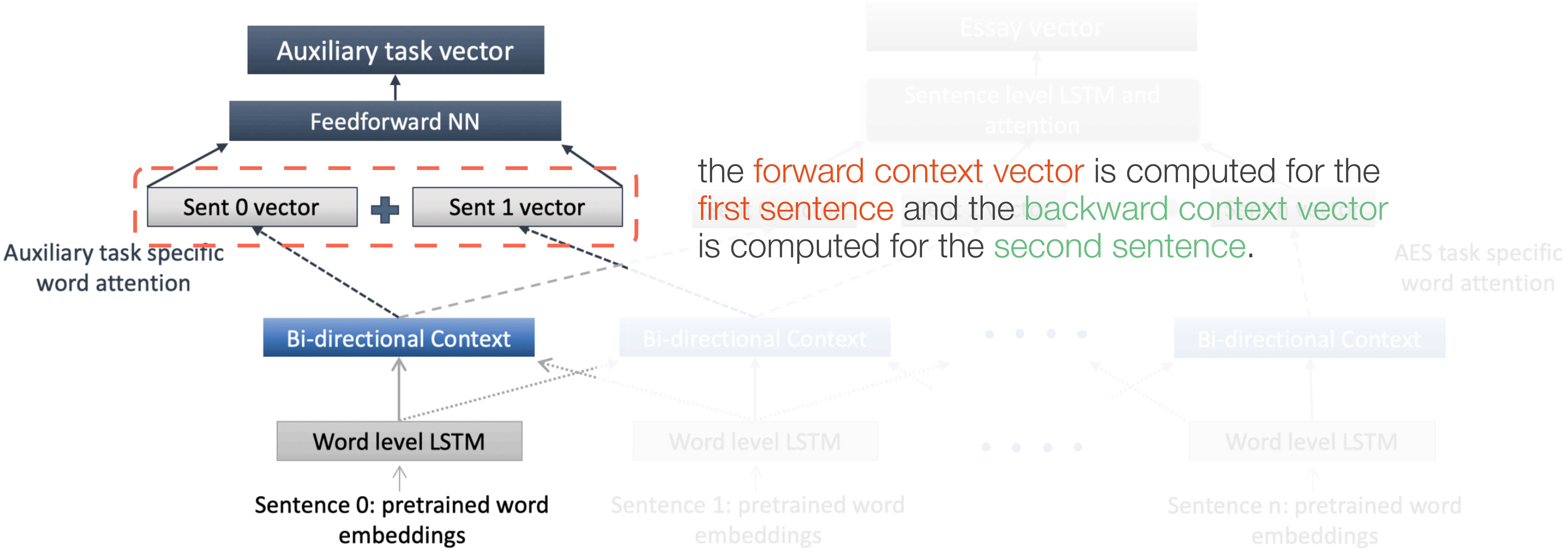
Auxiliary Training Tasks



Network structure for BCA with pretraining tasks

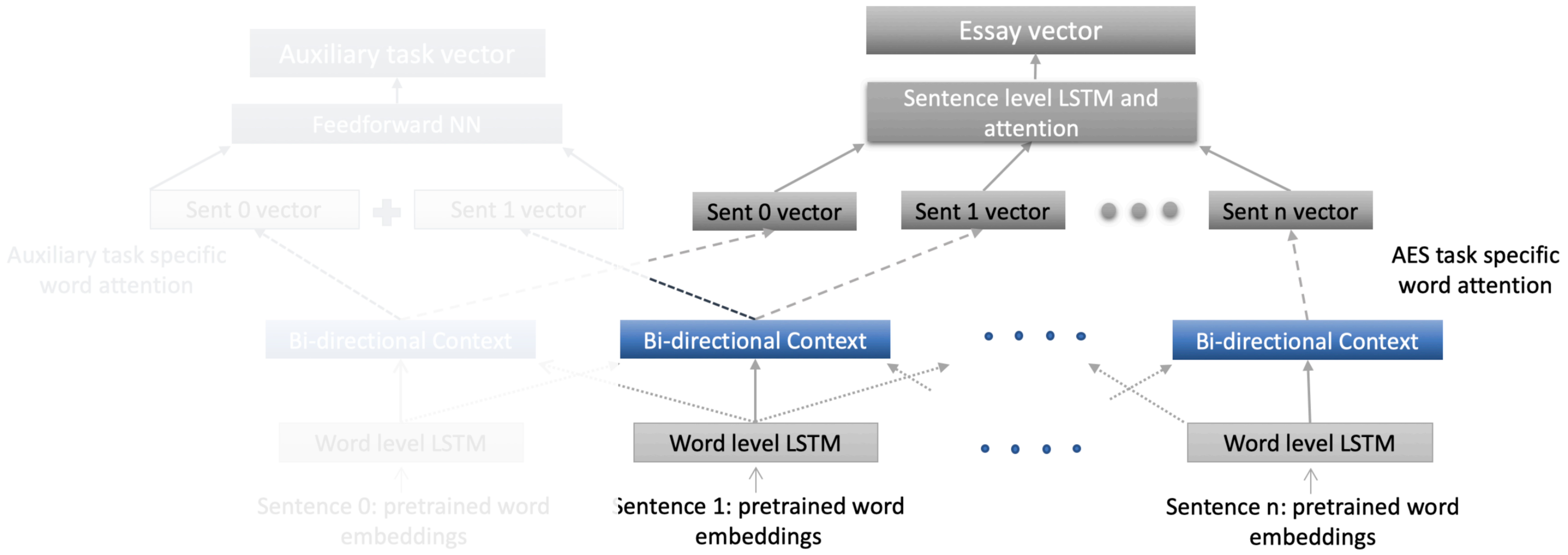


Network structure for BCA with pretraining tasks

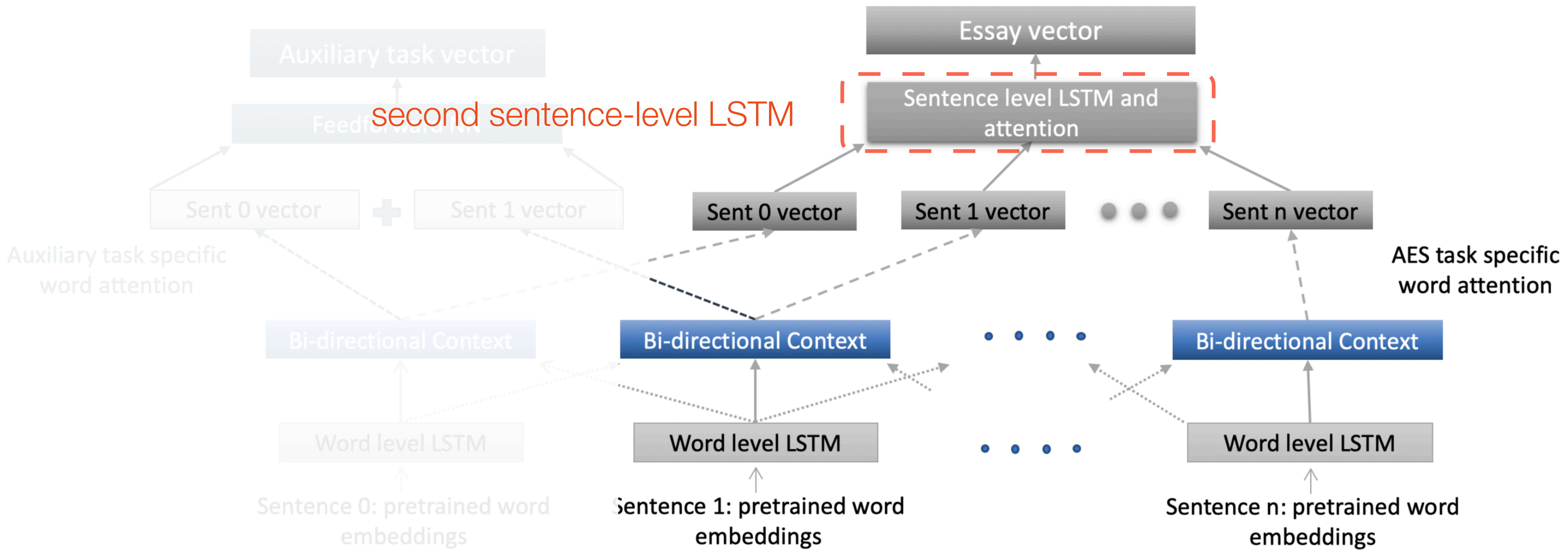


the forward context vector is computed for the first sentence and the backward context vector is computed for the second sentence.

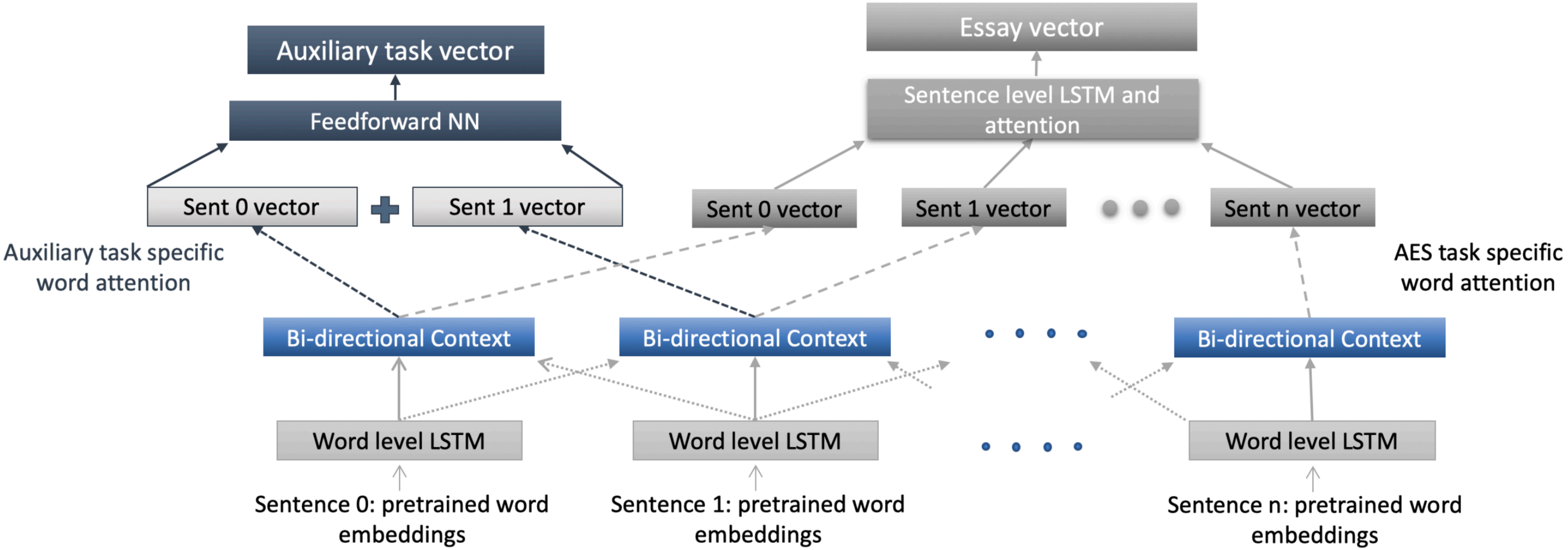
Network structure for BCA with pretraining tasks



Network structure for BCA with pretraining tasks



Network structure for BCA with pretraining tasks



Network structure for BCA with pretraining tasks

AES with Discourse-Aware Neural Models

1. background, intro — lay of the land
2. methods — what was done
3. data — what was used
4. results — how well it fares
5. conclusion — what it shows

AES with Discourse-Aware Neural Models

1. background, intro — lay of the land
2. methods — what was done
3. data — what was used
4. results — how well it fares
5. conclusion — what it shows

The Data

ETS Corpus of Non-Native Written English *(Source : LDC)*

- 12,100 TOEFL essays
- Essay scores classified as *high*, *medium*, *low*.
- 2 splits used:
  - Split 1 from LDC (11,000 train, 1100 test)
  - Split 2 by Klebanov et al. (6074 train, 2023 test)

The Data

ETS Corpus of Non-Native Written English *(Source : LDC)*

- 12,100 TOEFL essays
- Essay scores classified as *high*, *medium*, *low*.
- 2 splits used:
    - Split 1 from LDC (11,000 train, 1100 test)
    - Split 2 by Klebanov et al. (6074 train, 2023 test)

Data set	Essays	High	Medium	Low
Train/dev	11,000	3,835	5,964	1,202
Test	1,100	367	604	129
Train/dev	6,074	2,102	3,318	655
Test	2,023	700	1,101	222

The Data

ETS Corpus of Non-Native Written English *(Source : LDC)*

- 12,100 TOEFL essays
- Essay scores classified as *high*, *medium*, *low*.
- 2 splits used:
    - Split 1 from LDC (11,000 train, 1100 test)
    - Split 2 by Klebanov et al. (6074 train, 2023 test)

Data set	Essays	High	Medium	Low
Train/dev	11,000	3,835	5,964	1,202
Test	1,100	367	604	129
Train/dev	6,074	2,102	3,318	655
Test	2,023	700	1,101	222

The Data

Automated Student Assessment Prize (ASAP) Competition:

- To assess performance on smaller datasets
- First two sets chosen — persuasive essays
- No test sets provided. Results reported for 5-fold cross-validation.

Data set	Essays	Avg. len	Score range
1	1783	350	2-12
2	1800	350	1-6

The Data — Pre-training

NLI task : Stanford natural language inference (SNLI)
DM task : 13k free books from `smashwords.com`

Category	Number of samples
Idea justification	144022
Time relation	24600
Idea support	67223
Idea opposition	181949
Idea expansion	67800
Alternative	7203
Conclusion	88853
Negative samples	95450

Categories and data distribution for the discourse marker prediction task.

AES with Discourse-Aware Neural Models

1. background, intro — lay of the land
2. methods — what was done
3. data — what was used
4. results — how well it fares
5. conclusion — what it shows

AES with Discourse-Aware Neural Models

1. background, intro — lay of the land
2. methods — what was done
3. data — what was used
4. results — how well it fares
5. conclusion — what it shows

Results

Training configurations:

1. Training using only LDC essay data;
2. Pretraining with one task (NLI/DM) → training with essay data;
3. Pretraining the two aux. tasks (NLI-DM), followed by training with the essay data;
4. Training the BCA model with only the essay data, using static BERT token embeddings.

Results

Baselines:

1. Feature-based model : Gradient boosting — 33 argumentative features.
2. Neural baseline:
   - BERT sentence encoder
   - Universal sentence encoder (USE)

     (Not discourse aware)

   - These baselines are also hierarchical models: BERT-HAN and USE-HAN.

Results — LDC TOEFL essays

Model	Split 1	Split 2
Arg (Klebanov16)	-	0.344
Length (Klebanov16)	-	0.518
Arg + Len (Klebanov16)	-	0.540
Nguyen18	-	0.622
Feature baseline	0.659	0.642
USE-HAN	0.626	0.618
BERT-HAN	0.688	0.680
HAN	0.635	0.623
NLI-HAN	0.643	0.630
DM-HAN	0.651	0.654
NLI-DM-HAN	0.655	0.644
BCA	0.637	0.636
NLI-BCA	0.652	0.647
DM-BCA	0.661	0.661
NLI-DM-BCA	0.659	0.663
BERT-BCA	0.729	0.715

Results for the essay scoring task on LDC TOEFL corpus for both splits reported in QWK

Results — LDC TOEFL essays

Model	Split 1	Split 2
Arg (Klebanov16)	-	0.344
Length (Klebanov16)	-	0.518
Arg + Len (Klebanov16)	-	0.540
Nguyen18	-	0.622
Feature baseline	0.659	0.642
USE-HAN	0.626	0.618
BERT-HAN	0.688	0.680
HAN	0.635	0.623
NLI-HAN	0.643	0.630
DM-HAN	0.651	0.654
NLI-DM-HAN	0.655	0.644
BCA	0.637	0.636
NLI-BCA	0.652	0.647
DM-BCA	0.661	0.661
NLI-DM-BCA	0.659	0.663
BERT-BCA	0.729	0.715

All neural models beat previously reported results

Results for the essay scoring task on LDC TOEFL corpus for both splits reported in QWK

Results — LDC TOEFL essays

only two models that do not explicitly use discourse cues.

Model	Split 1	Split 2
Arg (Klebanov16)	-	0.344
Length (Klebanov16)	-	0.518
Arg + Len (Klebanov16)	-	0.540
Nguyen18	-	0.622
Feature baseline	0.659	0.642
USE-HAN	0.626	0.618
BERT-HAN	0.688	0.680
HAN	0.635	0.623
NLI-HAN	0.643	0.630
DM-HAN	0.651	0.654
NLI-DM-HAN	0.655	0.644
BCA	0.637	0.636
NLI-BCA	0.652	0.647
DM-BCA	0.661	0.661
NLI-DM-BCA	0.659	0.663
BERT-BCA	0.729	0.715

Results for the essay scoring task on LDC TOEFL corpus for both splits reported in QWK

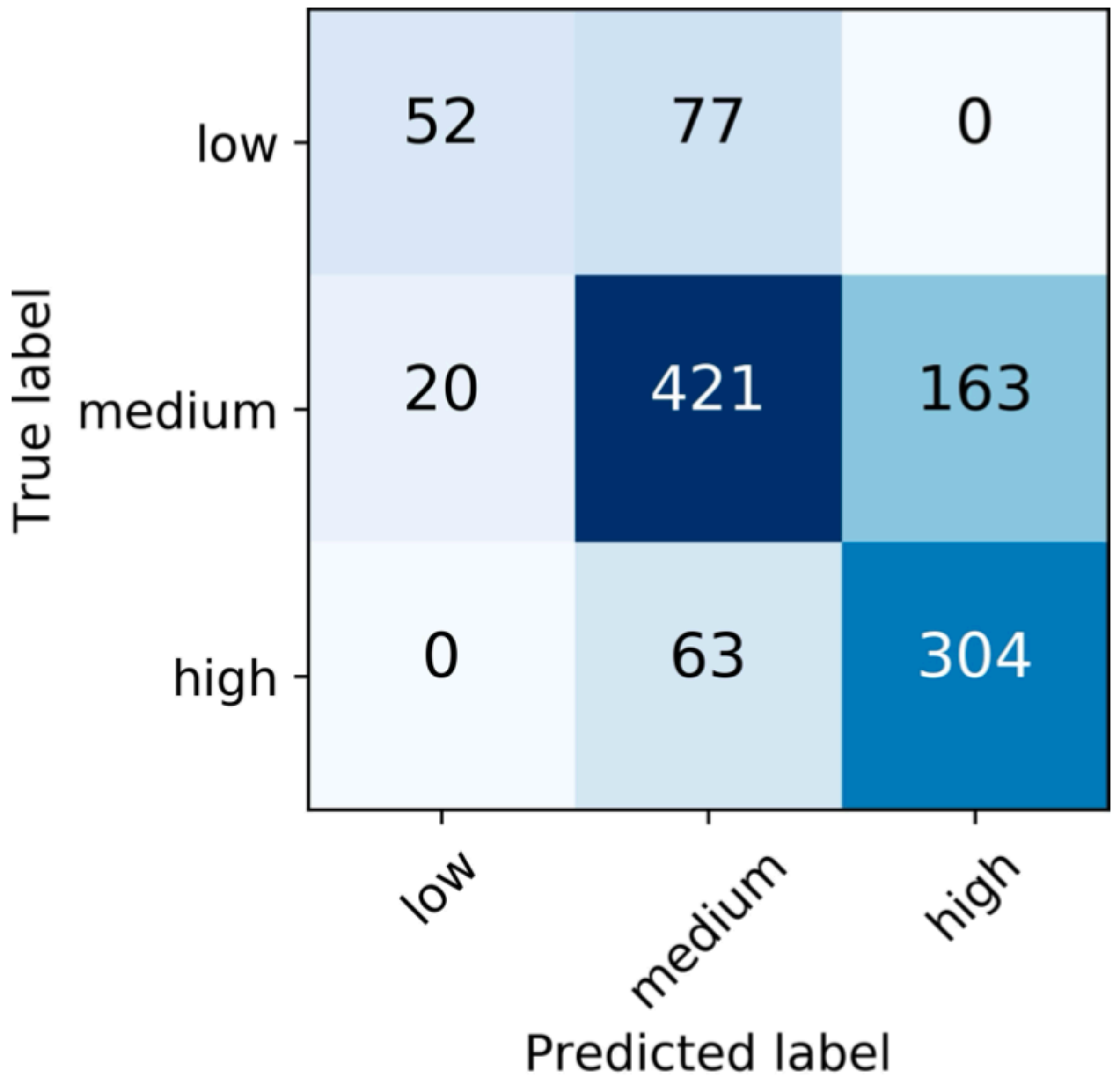
Results — LDC TOEFL essays

Model	Split 1	Split 2
Arg (Klebanov16)	-	0.344
Length (Klebanov16)	-	0.518
Arg + Len (Klebanov16)	-	0.540
Nguyen18	-	0.622
Feature baseline	0.659	0.642
USE-HAN	0.626	0.618
BERT-HAN	0.688	0.680
HAN	0.635	0.623
NLI-HAN	0.643	0.630
DM-HAN	0.651	0.654
NLI-DM-HAN	0.655	0.644
BCA	0.637	0.636
NLI-BCA	0.652	0.647
DM-BCA	0.661	0.661
NLI-DM-BCA	0.659	0.663
BERT-BCA	0.729	0.715

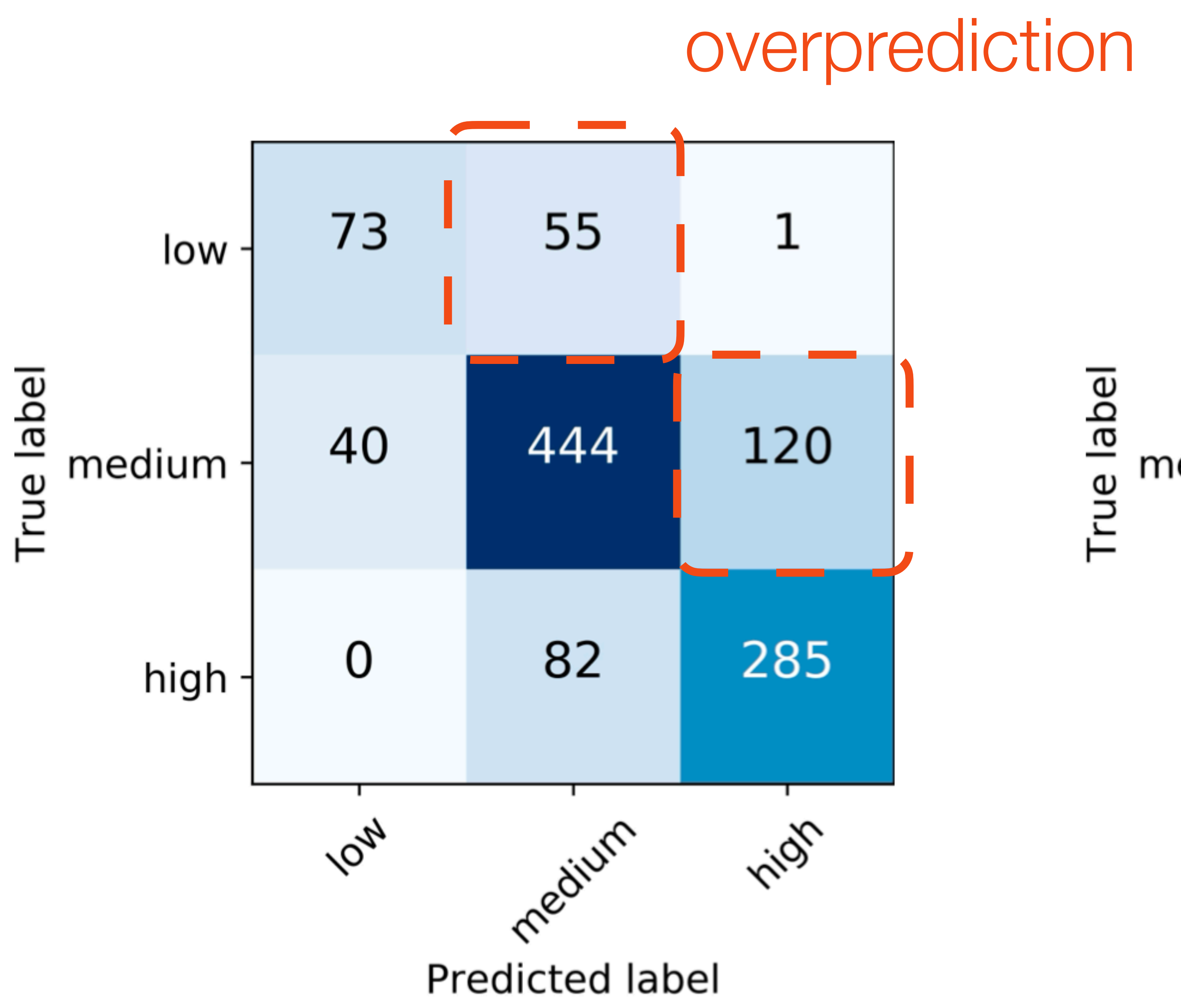
Combining contextualized word embeddings with cross-sentence attention gives best results.

Results for the essay scoring task on LDC TOEFL corpus for both splits reported in QWK

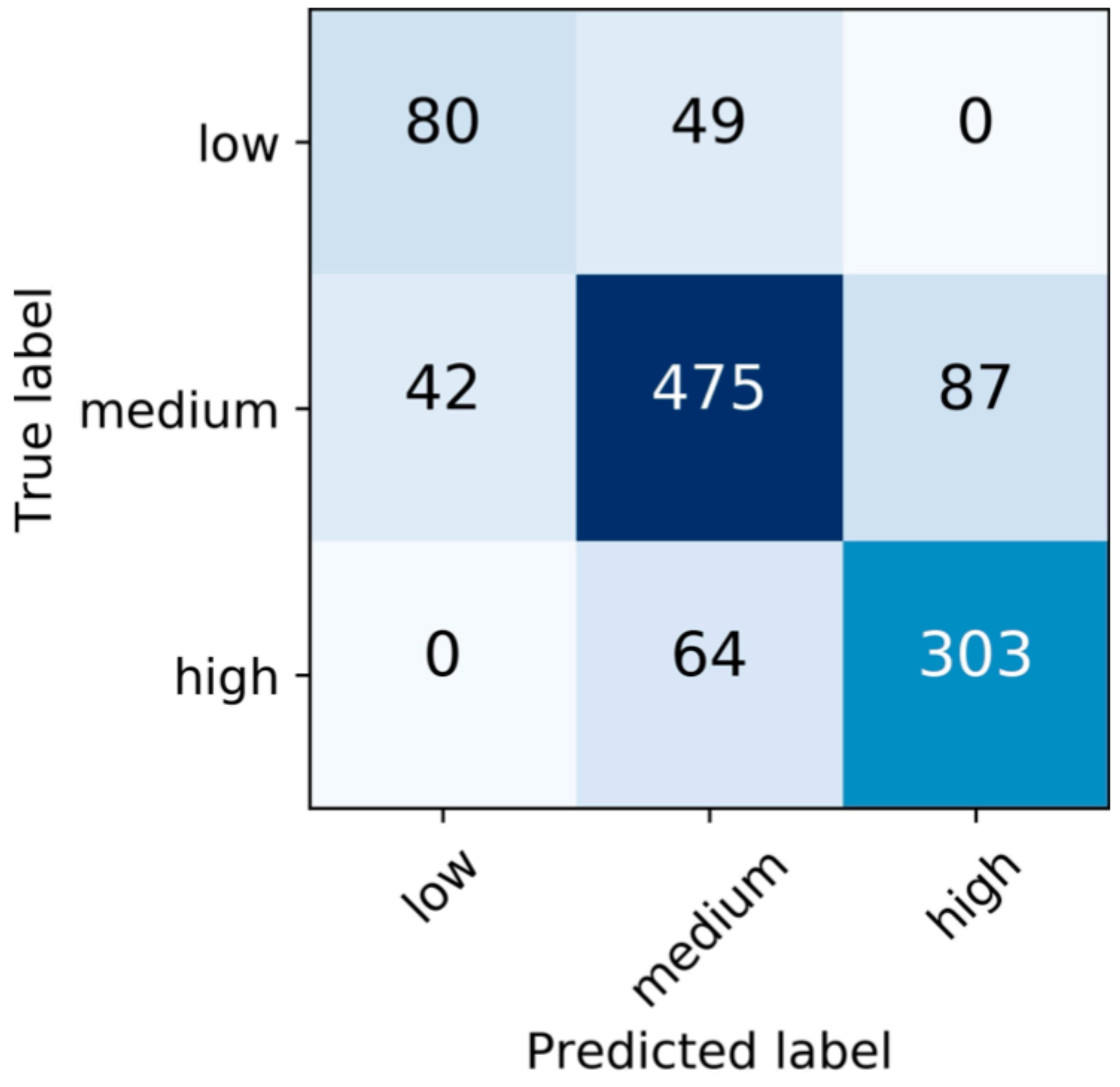
Results — LDC TOEFL essays



USE-HAN baseline



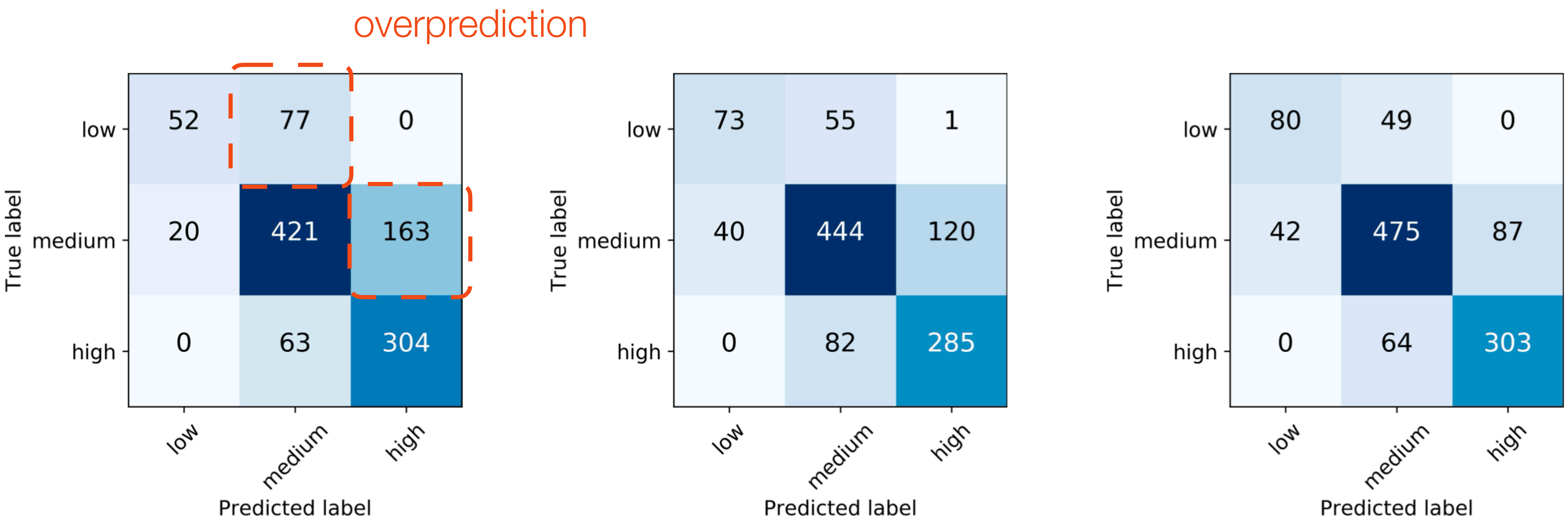
DM-BCA



BERT-BCA

Confusion Matrices for the USE-HAN baseline vs. the best neural models on LDC TOEFL split 1

Results — LDC TOEFL essays



USE-HAN baseline

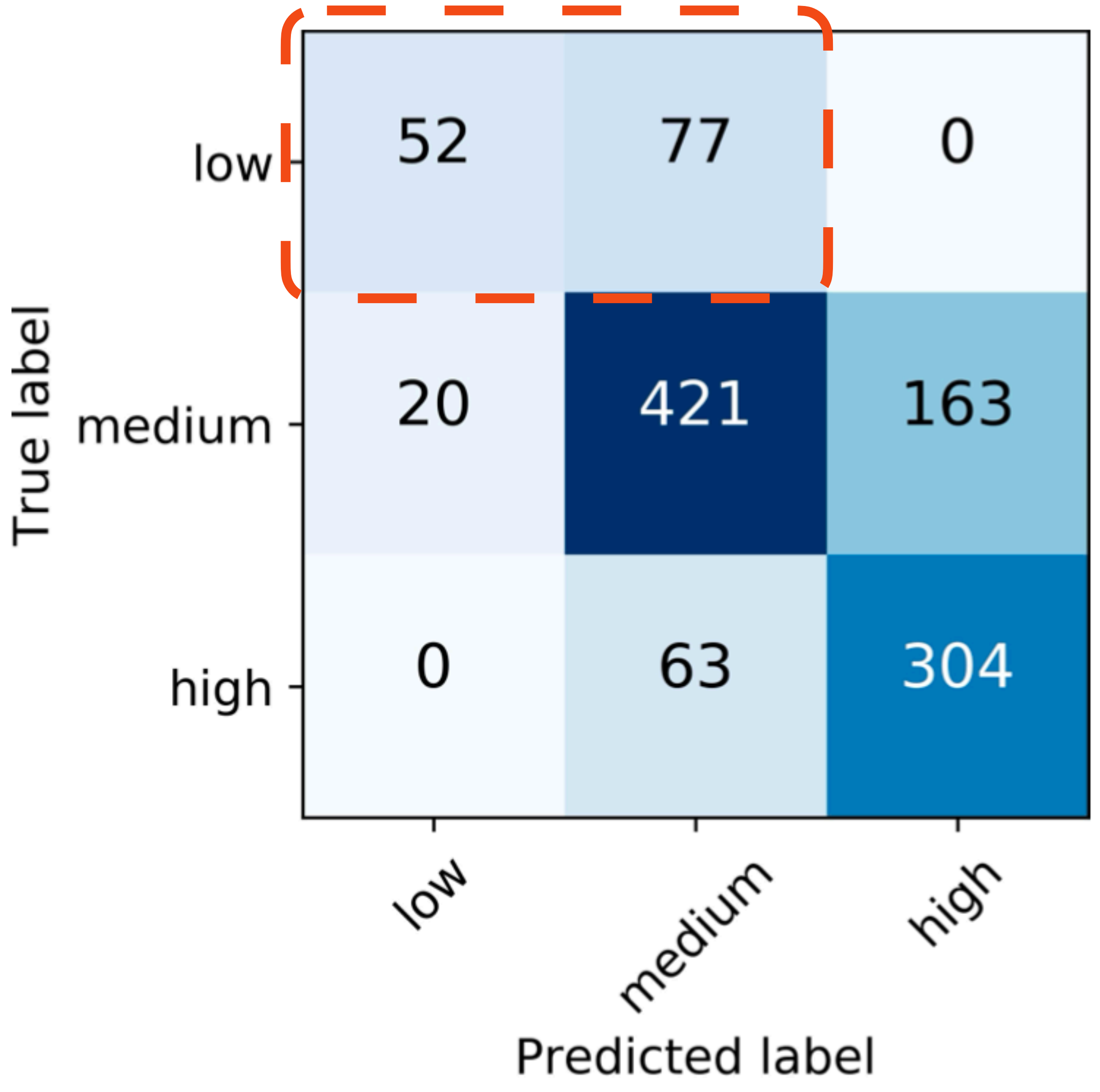
DM-BCA

BERT-BCA

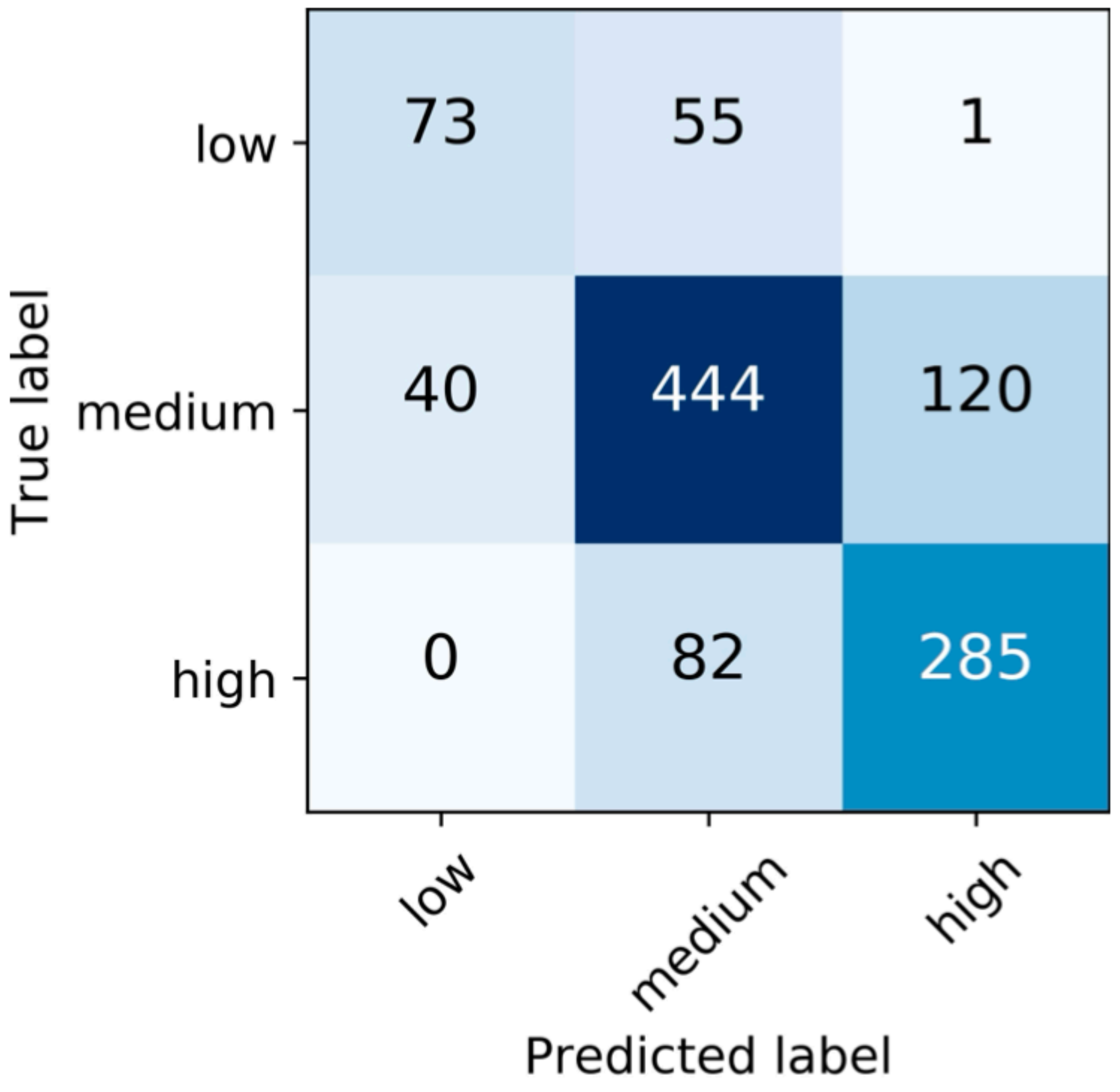
Confusion Matrices for the USE-HAN baseline vs. the best neural models on LDC TOEFL split 1

Results — LDC TOEFL essays

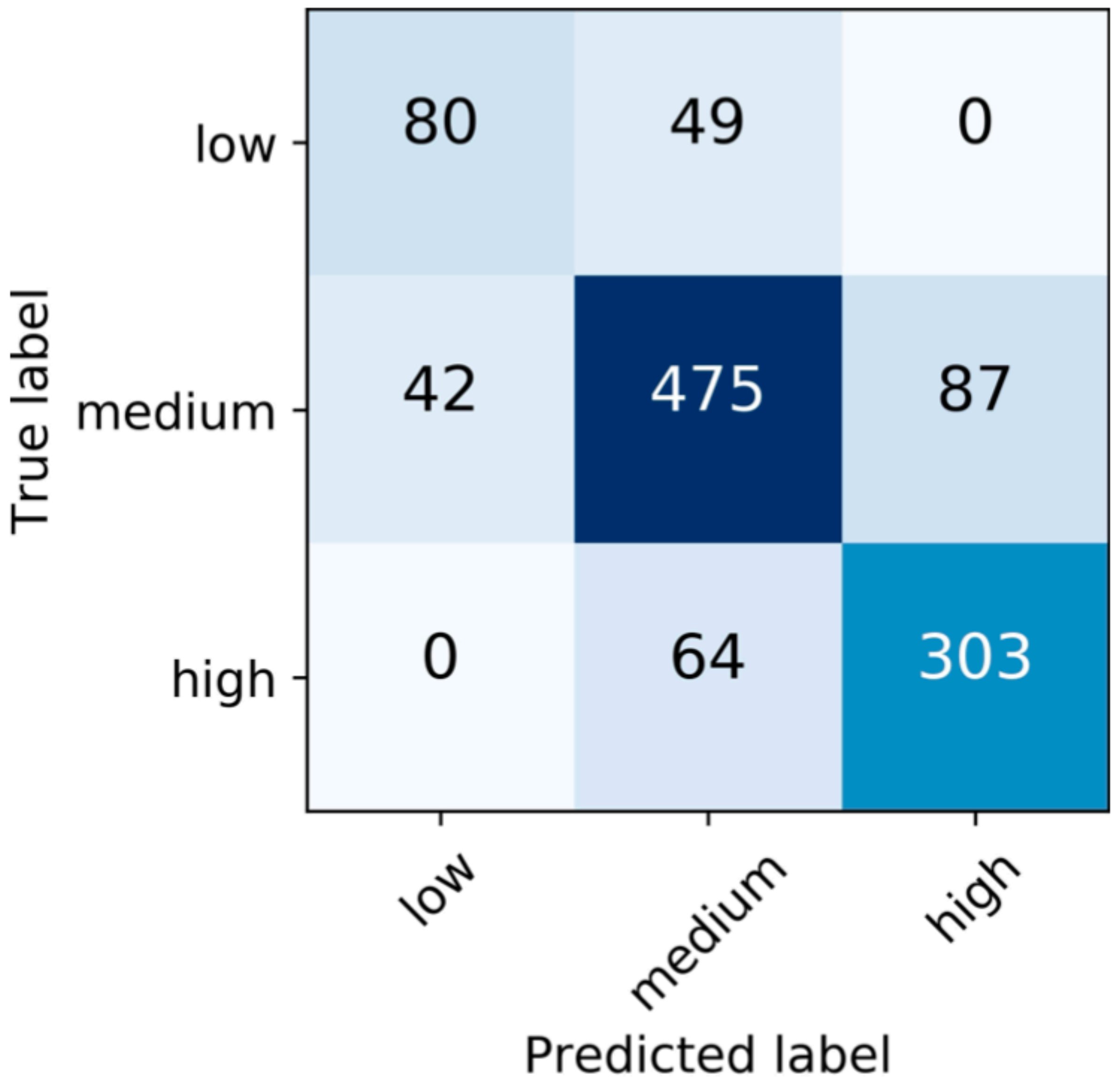
poor recall for “low” overprediction



USE-HAN baseline



DM-BCA



BERT-BCA

Confusion Matrices for the USE-HAN baseline vs. the best neural models on LDC TOEFL split 1

Results — ASAP Essays

Feature-based models outperform neural models. 🤔

Model	ASAP 1	ASAP 2
TSLF (Liu 2019)	0.852	0.736
Feature baseline	0.833	0.692
BERT-HAN	0.748	0.627
NLI-DM-BCA	0.800	0.671
NLI-DM-BCA+features	0.840	0.711

Results for the essay scoring task for ASAP sets 1 and 2 reported in QWK.

Results — ASAP Essays

Model	ASAP 1	ASAP 2
TSLF (Liu 2019)	0.852	0.736
Feature baseline	0.833	0.692
BERT-HAN	0.748	0.627
NLI-DM-BCA	0.800	0.671
NLI-DM-BCA+features	0.840	0.711

Results for the essay scoring task for ASAP sets 1 and 2 reported in QWK.

Small gains for combination model

Try a more sophisticated combination?

AES with Discourse-Aware Neural Models

An overview of our discussion today:

1. background, intro — lay of the land
2. methods — what was done
3. data — what was used
4. results — how well it fares
5. conclusion — what it shows

AES with Discourse-Aware Neural Models

An overview of our discussion today:

1. background, intro — lay of the land
2. methods — what was done
3. data — what was used
4. results — how well it fares
5. conclusion — what it shows

The Bottom Line : Learning and Takeaways

A neural model with cross-sentence dependencies + discourse-based training task = performance boost on feature-based SOTA.

The NLI task does not contribute much.

Using pre-trained BERT tokens can boost performance (even more!) on TOEFL data. NSP a silent contributor presumably.

For ASAP, neural models underperform feature-based systems.

Best results are achieved with a model that uses combination of hand-crafted + neural representation.

The Bottom Line : Learning and Takeaways

A neural model with cross-sentence dependencies + discourse-based training task = performance boost on feature-based SOTA.

The NLI task does not contribute much.

Using pre-trained BERT tokens can boost performance (even more!) on TOEFL data. NSP a silent contributor presumably.

For ASAP, neural models underperform feature-based systems.

Best results are achieved with a model that uses combination of hand-crafted + neural representation.

The Bottom Line : Learning and Takeaways

A neural model with cross-sentence dependencies + discourse-based training task = performance boost on feature-based SOTA.

The NLI task does not contribute much.

Using pre-trained BERT tokens can boost performance (even more!) on TOEFL data. NSP a silent contributor presumably.

For ASAP, neural models underperform feature-based systems.

Best results are achieved with a model that uses combination of hand-crafted + neural representation.

The Bottom Line : Learning and Takeaways

A neural model with cross-sentence dependencies + discourse-based training task = performance boost on feature-based SOTA.

The NLI task does not contribute much.

Using pre-trained BERT tokens can boost performance (even more!) on TOEFL data. NSP a silent contributor presumably.

For ASAP, neural models underperform feature-based systems.

Best results are achieved with a model that uses combination of hand-crafted + neural representation.

The Bottom Line : Learning and Takeaways

A neural model with cross-sentence dependencies + discourse-based training task = performance boost on feature-based SOTA.

The NLI task does not contribute much.

Using pre-trained BERT tokens can boost performance (even more!) on TOEFL data. NSP a silent contributor presumably.

For ASAP, neural models underperform feature-based systems.

Best results are achieved with a model that uses combination of hand-crafted + neural representation.

The Bottom Line : Learning and Takeaways

A neural model with cross-sentence dependencies + discourse-based training task = performance boost on feature-based SOTA.

The NLI task does not contribute much.

Using pre-trained BERT tokens can boost performance (even more!) on TOEFL data. NSP a silent contributor presumably.

For ASAP, neural models underperform feature-based systems.

Best results are achieved with a model that uses combination of hand-crafted + neural representation.

Thank you.

General Resources

visualisingdata.com/resources/

D3 (js), matplotlib (python), seaborn (python), ggplot (R, python)

Storytelling with data:

https://www.amazon.com/Storytelling-Data-Visualization-Business-Professionals/dp/1119002257/ref=nodl_

Caveats to data visualization:

https://www.data-to-viz.com/caveats.html

Randal Olson's matplotlib tips:

http://www.randalolson.com/2014/06/28/how-to-make-beautiful-data-visualizations-in-python-with-matplotlib/

Colors

coolors.co, palettable.io (custom color palettes)

jiffyclub.github.io/palettable (colors in Python)

colororacle.org (color blind test app)

ianstormtaylor.com/design-tip-never-use-black

Science as Art

http://worrydream.com/ScientificCommunicationAsSequentialArt/

r2d3.us/visual-intro-to-machine-learning-part-1

r-graph-gallery.com/portfolio/data-art/

Thank you.

Slides inspired by:

Sam Way (Spotify)
Prof. Dan Larremore (CU Boulder)
Prof. Aaron Clauset (CU Boulder)