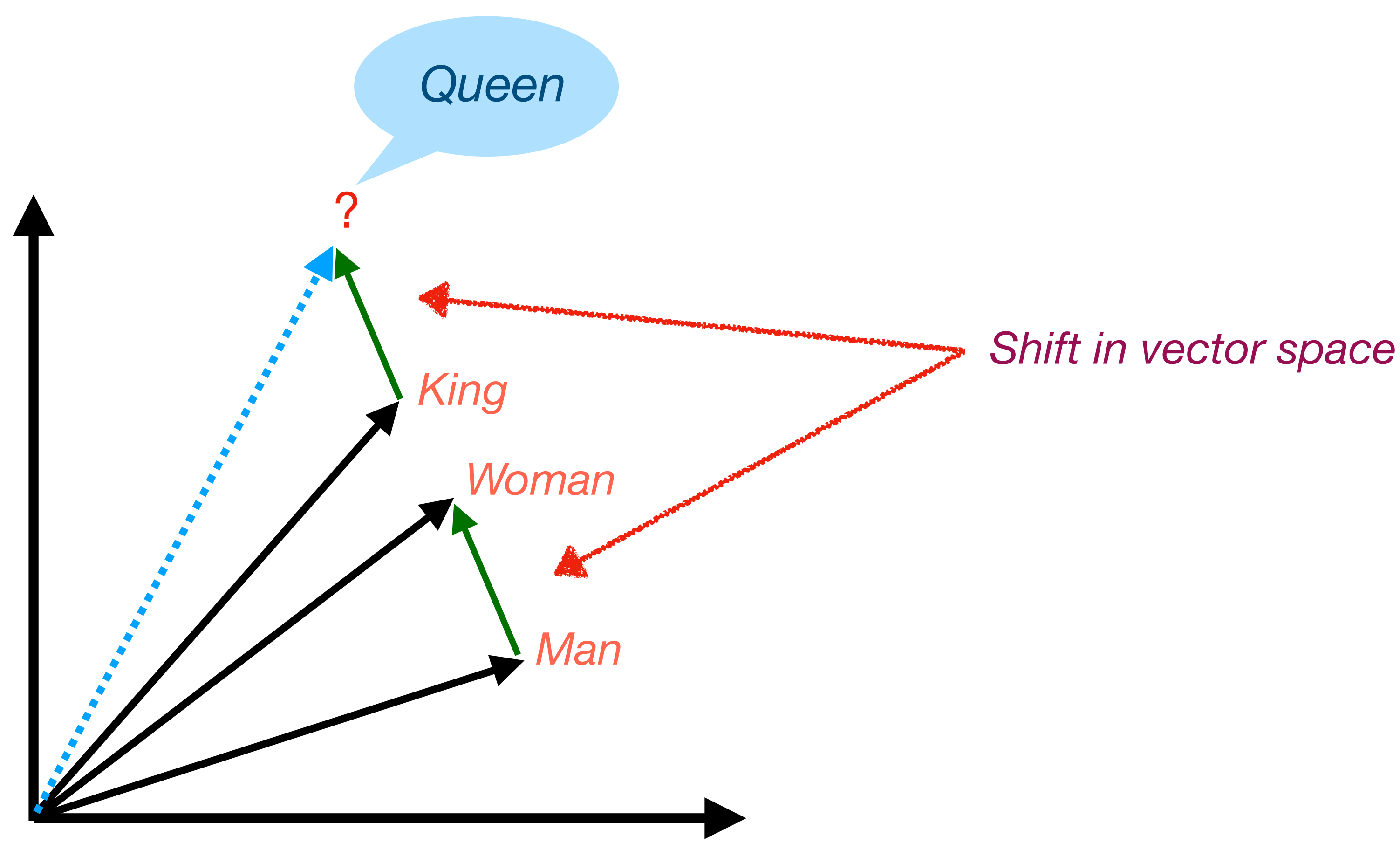
Gender Bias in BERT

Numeric Representations of Words

Word embedding: high-dimensional yet compact meaningful vector representations of words.

A meaningful vector representations of words can give answers to —

(Woman - Man) + King = ?



Gender Bias in Word Representations

Word-word co-occurrence

Dataset

Bias inherit

Male

Female

Gender Bias in Word Representations

Word-word co-occurrence

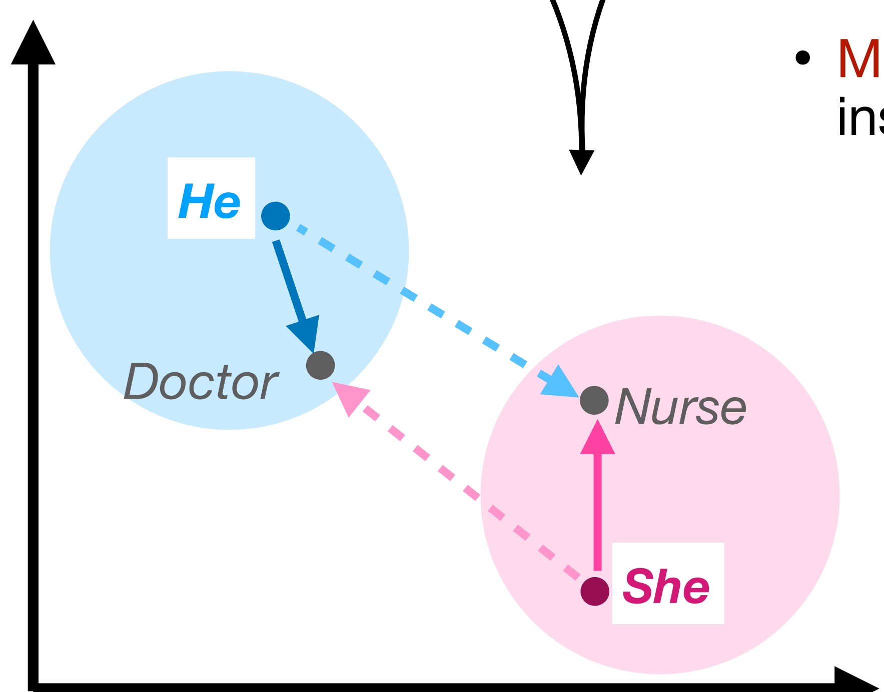
Dataset

Bias inherit

Male

Female

Word vectors



- Model inherit bias from the dataset. For instance, unequal distances:

$$He - Doctor \quad < \quad She - Doctor$$

$$He - Nurse \quad > \quad She - Nurse$$

Do you notice gender colour stereotype here?

Problem Formulation

Since it is hard to find and remove all the gender stereotypes from the dataset, we focus on identify biases inherited by the models trained on such datasets.

- Define: When do we call a system is gender biased?
- Quantify: How to measure/quantify the gender bias?
- Solve: How to gender debias?

Measure Bias

How to measure the bias?

- Compute the difference between similarities of a gender neutral word (eg., Doctor) from gender-specific words (eg. Male and Female).

Similarity score: Cosine angle, Euclidean distance.

- Compute the difference between prediction scores when a gender-specific word is switched to a word specifying some other gender.

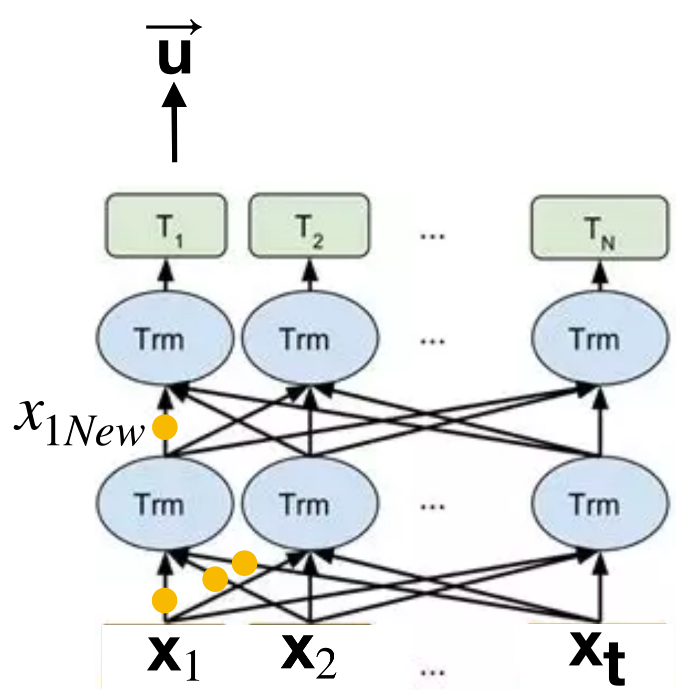
- Some other way that you may figure out in future…

Contextualised Word Embedding

- BERT: Bidirectional Encoder Representations from Transformers (BERT) is popular provides contextualised word vectors. Consider two sentences:

S1: Singapore is a nice country.
S2: Singapore university of technology and design.

$\vec{u}$ = BERT(S1) (vector of Singapore in S1)
$\vec{v}$ = BERT(S2) (vector of Singapore in S2)



Singapore is a nice country.

Contextualised Word Embedding

- BERT: Bidirectional Encoder Representations from Transformers (BERT) is popular provides contextualised word vectors. Consider two sentences:

S1: <u>Singapore</u> is a nice country.
S2: <u>Singapore</u> university of technology and design.

$\vec{u}$ = BERT(S1) (vector of Singapore in S1)
$\vec{v}$ = BERT(S2) (vector of Singapore in S2)

We find…

- $Cosine(\vec{u}, \vec{v}) \neq 1$
- $\vec{u} - \vec{v} \neq 0$

Even though $\vec{u}$ and $\vec{v}$ are vectors for the same word <u>Singapore</u>, they are different because of the context.

Measure Bias in Contextualised Embedding

- One way to quantify gender bias in BERT is by observing predictions of downstream models utilising it as an underlying language model.

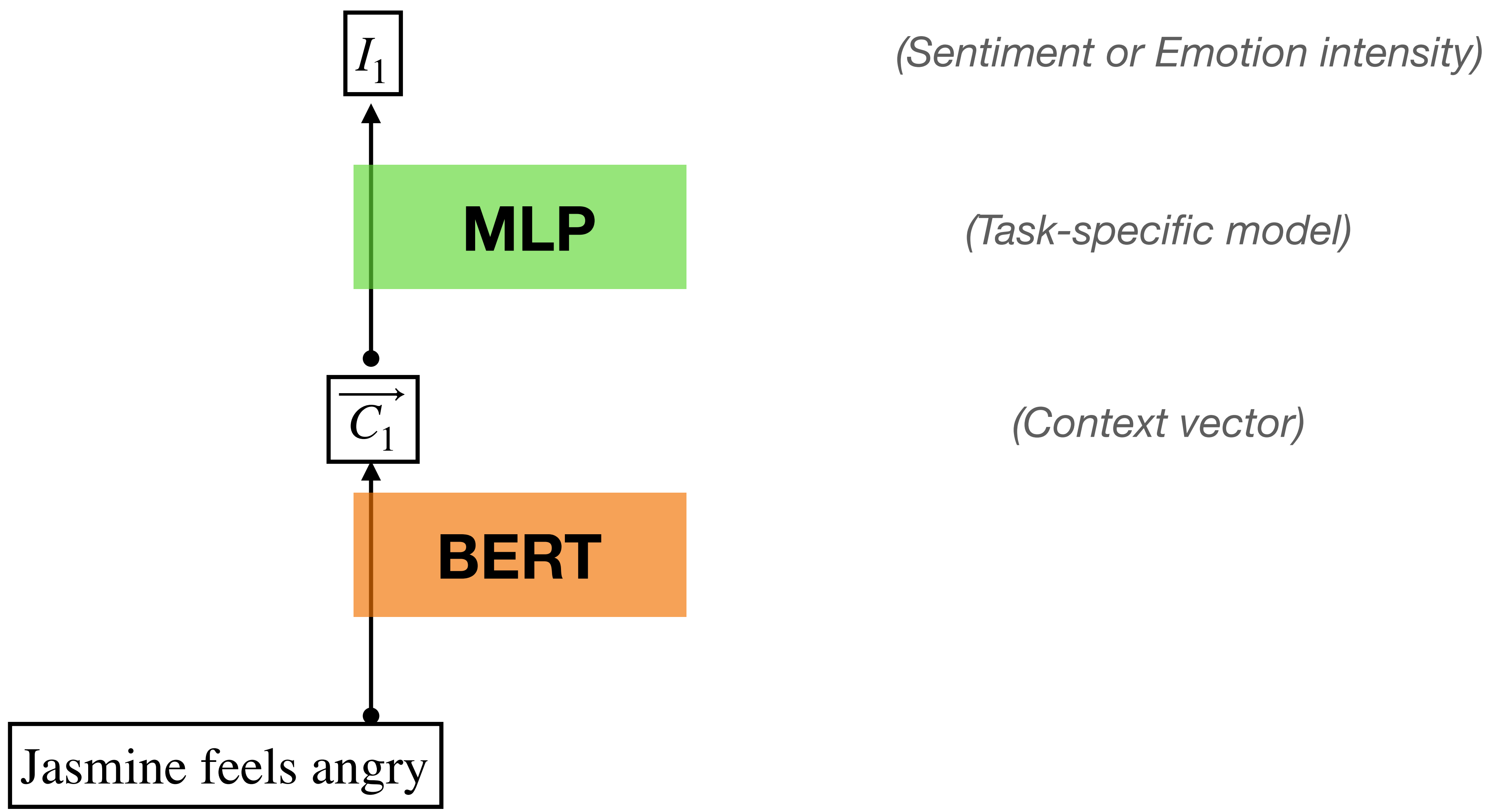
Downstream tasks we consider:

Given a sentence such as "Jasmine feels angry", predict

1) Intensity of <u>emotion</u>

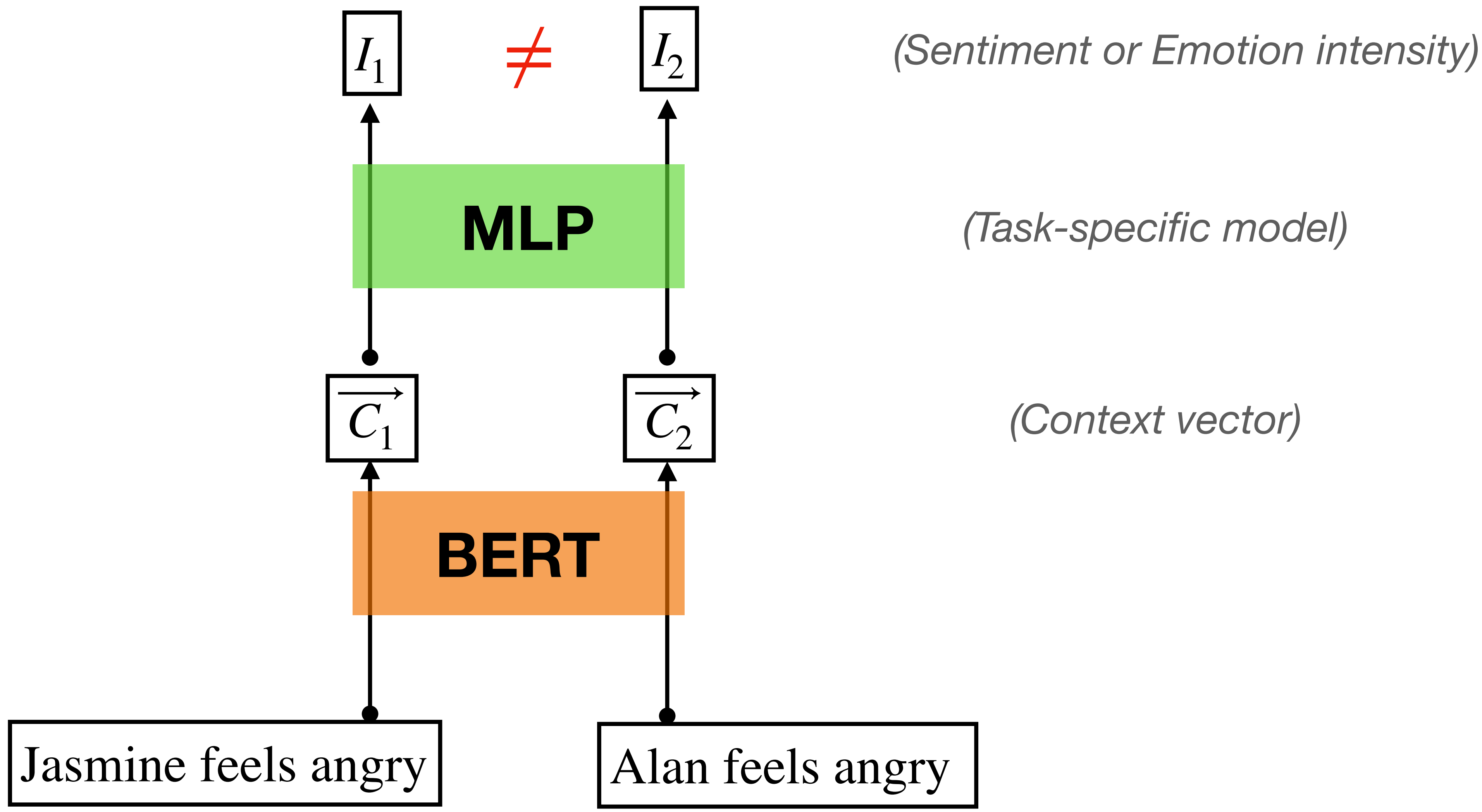
2) Intensity of <u>sentiment</u>

Measure Bias in Contextualised Embedding



Note: The MLP is kept simple (1-hidden layer) to prevent learning it's own bias.

Measure Bias in Contextualised Embedding

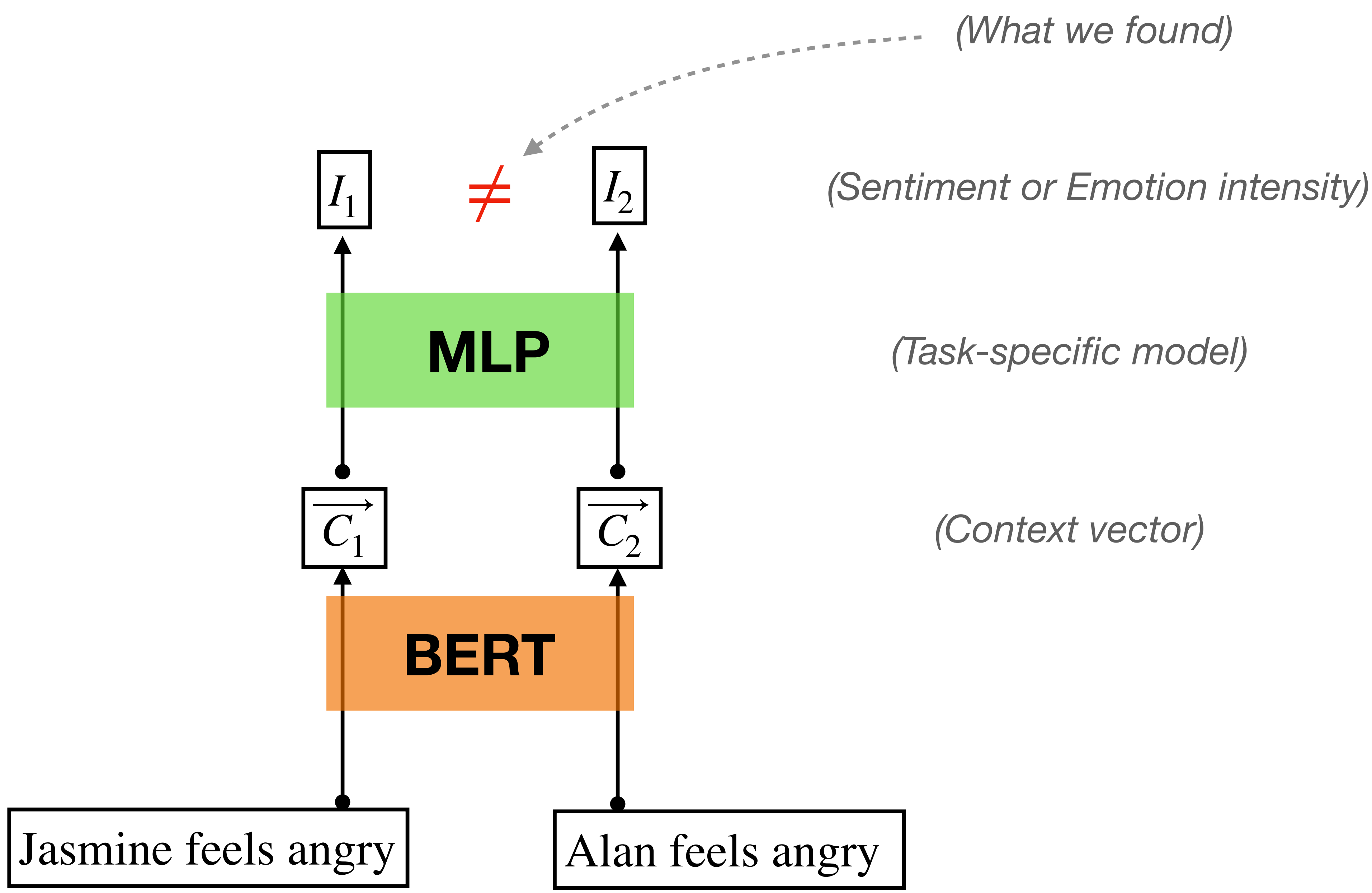


An ideal model is supposed to assign equal intensity scores.

BERT-MLP does not follow that!

Note: The MLP is kept simple (1-hidden layer) to prevent learning it's own bias.

Measure Bias in Contextualised Embedding



An ideal model is supposed to assign equal intensity scores.

BERT-MLP does not follow that!

Note: The MLP is kept simple (1-hidden layer) to prevent learning it's own bias.

Solve: Gender Debiasing Contextualised Embedding

We identify a subspace where BERT encodes gender information.

Approach:

Step1: We collect a set of *n*-gender word pairs such as:

$$(f_i, m_i) \in \{(Queen, King), (She, He), \ldots (Woman, Man)\}$$

(The words in a gender pair are gender opposite of each other based on their common usage.)

Solve: Gender Debiasing Contextualised Embedding

We identify a subspace where BERT encodes gender information.

Approach:

Step1: We collect a set of *n*-gender word pairs such as:

$$(f_i, m_i) \in \{(Queen, King), (She, He), \ldots (Woman, Man)\}$$

(The words in a gender pair are gender opposite of each other based on their common usage.)

Step2: Feed them to BERT and obtain corresponding word embeddings.

$\vec{u}_i$ is word vector for f_i and $\vec{v}_i$ is word vector for m_i .

Solve: Gender Debiasing Contextualised Embedding

We identify a subspace where BERT encodes gender information.

Approach:

Step1: We collect a set of *n*-gender word pairs such as:

$$(f_i, m_i) \in \{(Queen, King), (She, He), \ldots (Woman, Man)\}$$

(The words in a gender pair are gender opposite of each other based on their common usage.)

Step2: Feed them to BERT and obtain corresponding word embeddings.

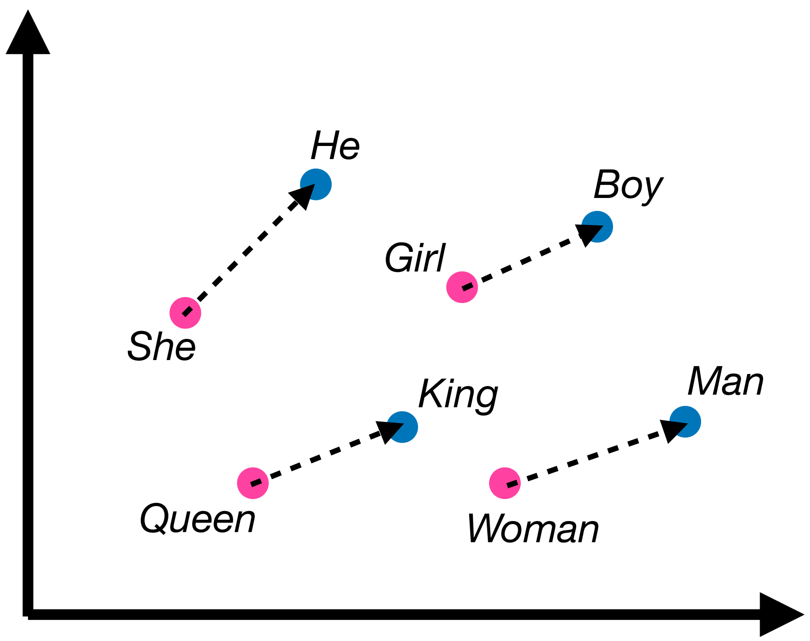
$\vec{u}_i$ is word vector for f_i and $\vec{v}_i$ is word vector for m_i .

Step3: Obtain a set of difference vectors $\{\vec{d}_1, \ldots, \vec{d}_n\}$, where $\overrightarrow{d_i} = (\vec{v}_i - \vec{u}_i)$

(Each difference vector shows shift from female to male in vector space.)

Solve: Gender Debiasing Contextualised Embedding

Step4: We have a set of *n*-difference vectors $\{\vec{d}_1, \ldots, \vec{d}_n\}$, each vector shows a shift in gender direction.



*Questions: **Can** we used it to find the directions (subspace) that encode gender information? i.e., do we know any algorithm that can help uncover the principal directions which explains the most spread of vectors in $\{\vec{d}_1, \ldots, \vec{d}_n\}$?*

PCA Recap

Remember: First principal component 1_{st} PC corresponds to the direction that best fits the data, 2_{nd} PC is best fit direction orthogonal to 1_{st} PC.

(Best fit implies the direction of a line that minimises the average square distance from data points to the line)

Difference vectors

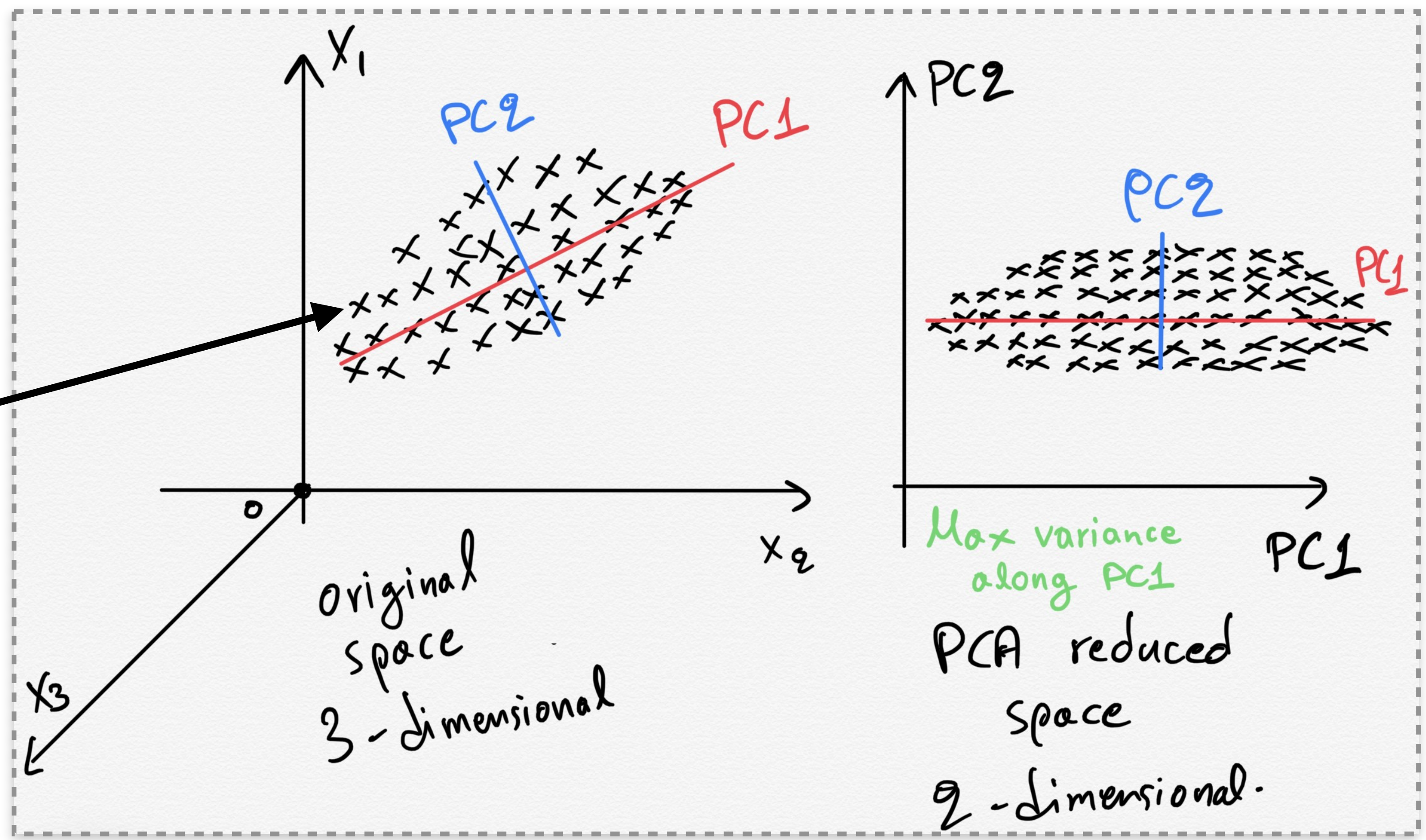


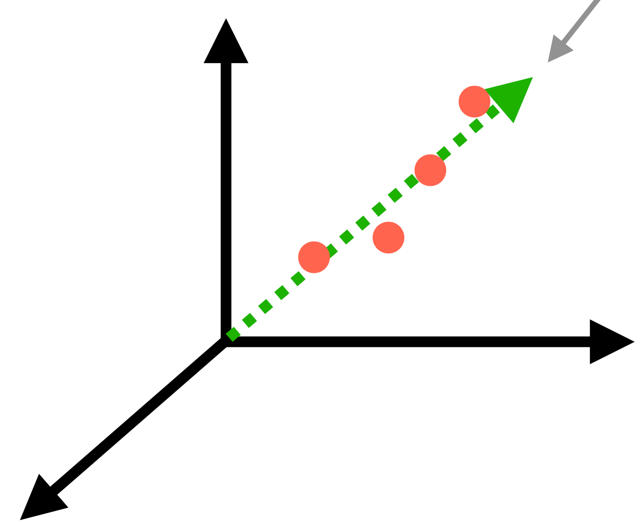
Image Source

Solve: Gender Debiasing Contextualised Embedding

Step5: Compute PCs. The 1_{st} PC covers explains most variance in the difference vectors. Hence, we consider the gender subspace to be 1-dimensional with axis identified by 1_{st} PC.

(in 768-dimensional space of word vectors)

Something like this



Black line denotes an axis of the original BERT vector space and are many, just for illustration shown three axes

Step6: Removing obtained gender direction during inference:

From each word vector produced by BERT, we remove the component in PC1. This will reduce the gender-specific information from the BERT word embedding.

ALGORITHM 1: Extracting layer-wise principal component in Gender subspace.

Input : - Strings pair (S_m, S_f), which differ only in gender-specific words.

Output : - P=Layer-wise principal component set $\{P_0, \ldots, P_{12}\}$.

$W_{tf} \leftarrow \texttt{Tokenize}(S_f)$ /* WP Tokenization */

$W_{tm} \leftarrow \texttt{Tokenize}(S_m)$ /* WP Tokenization */

$u_0 \leftarrow \texttt{Layer}_0(W_{tf})$ /* Context-independent input vectors for S_f */

$v_0 \leftarrow \texttt{Layer}_0(W_{tm})$ /* Context-independent input vectors for S_m */

$D_0 \leftarrow (v_0 - u_0)$ /* Difference vector */

$P_0 \leftarrow \texttt{PCA}(D_0)$ /* PC with maximum EV */

for $j \leftarrow [1, 2, \ldots, 12]$ **do**

$\quad u^*_{j-1} \leftarrow \texttt{Proj}_{\perp P_{j-1}}(u_{j-1})$ /* Perpendicular projection */

$\quad v^*_{j-1} \leftarrow \texttt{Proj}_{\perp P_{j-1}}(v_{j-1})$

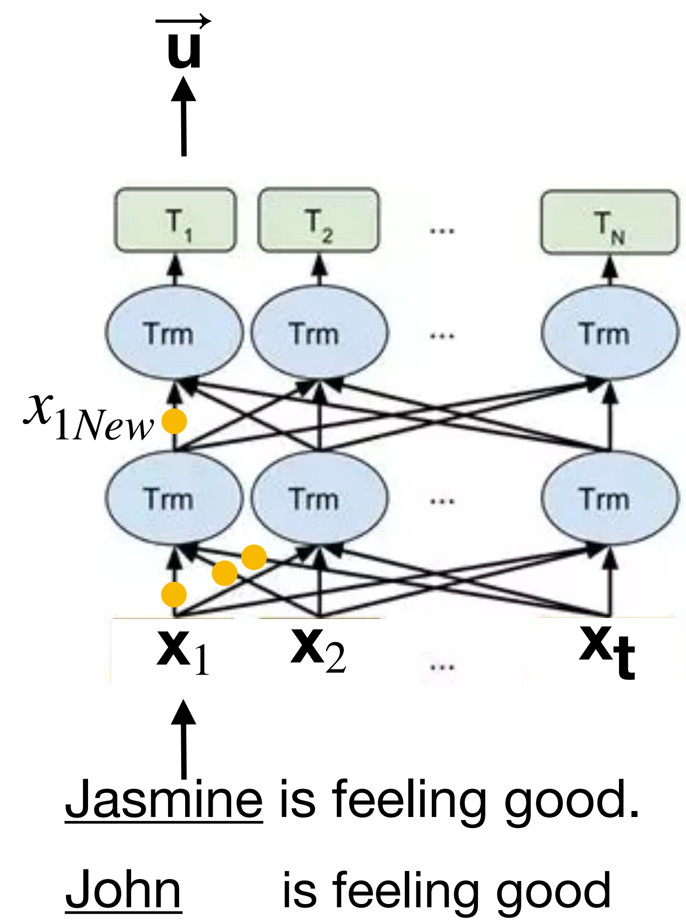
$\quad u_j \leftarrow \texttt{Layer}_j(u^*_{j-1})$

$\quad v_j \leftarrow \texttt{Layer}_j(v^*_{j-1})$

$\quad D_j \leftarrow (v_j - u_j)$ /* Difference vector */ x_{1New}

$\quad P_j \leftarrow \texttt{PCA}(D_j)$

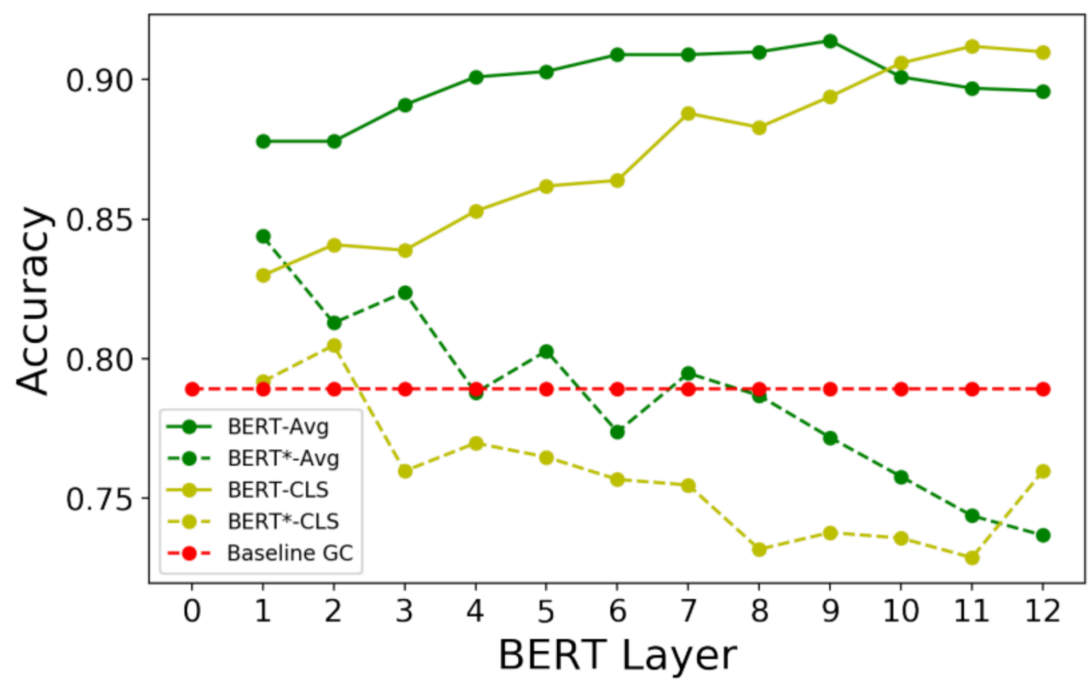
end



Algorithm: Iterative computation of principal components.

Emotion	Emotion Intensity						Valence Intensity					
	BERT			BERT^{De}			BERT			BERT^{De}		
	Pearson	$\Delta_{F\uparrow-M\downarrow}$	$\Delta_{M\uparrow-F\downarrow}$	Pearson	$\Delta_{F\uparrow-M\downarrow}$(%d)	$\Delta_{M\uparrow-F\downarrow}$(%d)	Pearson	$\Delta_{F\uparrow-M\downarrow}$	$\Delta_{M\uparrow-F\downarrow}$	Pearson	$\Delta_{F\uparrow-M\downarrow}$(%d)	$\Delta_{M\uparrow-F\downarrow}$(%d)
Joy	0.666	0.0396	0.0402	0.660	0.0143(↓**63.9**)	0.0143(↓**64.4**)	0.659	0.0346	0.0376	0.670	0.0209(↓**39.5**)	0.0138(↓**63.3**)
Fear	0.581	0.0202	0.0244	0.593	0.0152(↓**24.7**)	0.0158(↓**35.2**)		0.0263	0.0244		0.0156(↓**40.6**)	0.0123(↓**49.5**)
Sadness	0.615	0.0380	0.0138	0.604	0.0178(↓**58.9**)	0.0097(↓**29.7**)		0.0272	0.0205		0.0153(↓**43.7**)	0.0118(↓**42.4**)
Anger	0.627	0.0074	0.0316	0.626	0.0121(↑63.5)	0.0149(↓**52.8**)		0.0219	0.0198		0.0130(↓**40.6**)	0.0119(↓**39.8**)

Table 2: Final-layer ($layer_{12}$) of BERT and BERT^{De} equity evaluation of the five-intensity regression models. %d refers to the percentage change in Δ values. The p-values for 1) Emotion intensity models: {anger}≤ 0.05 ($\geq$0.70*); {joy, fear, sad} $\leq$ 0.20 ($\geq$0.70*). 2) The valence intensity model (emotion-wise p-values): {anger}≤ 0.05 ($\geq$0.75*); {joy, fear, sad}≤ 0.20 ($\geq$0.70*), where values with * denotes BERT^{De}-based MLP regressor.



x_t

Figure 4: BERT-CLS and BERT^{De}-CLS denote MLP accuracy using $layer_k$ (x-axis) vectors in I_1 setting. Similarly, BERT-Avg and BERT^{De}-Avg refer to the I_2 setting. Switching from BERT to BERT^{De}, we see a significant drop in MLP gender-classification performance in both I_1 and I_2 inputs cases.