Kinder, the empathetic writing assistant

“Kindness is the language which the deaf can hear and the blind can see.” – Mark Twain

MIDS Capstone Summer 2025
Andy Guinto, Diego Moss,
Priyanka Upadhyay

UC Berkeley School of Information

Image generated with Perplexity

The problem and our impact

Our vision is to improve people's well-being by making everyday communication over online and mobile apps more empathetic.

Studies Show a 40% Decline in Empathy Among College Students

Is Technology Destroying Communication in the Workplace?

Angry by design: toxic communication and technical architectures

Greater use of virtual technology leads to toxic communications

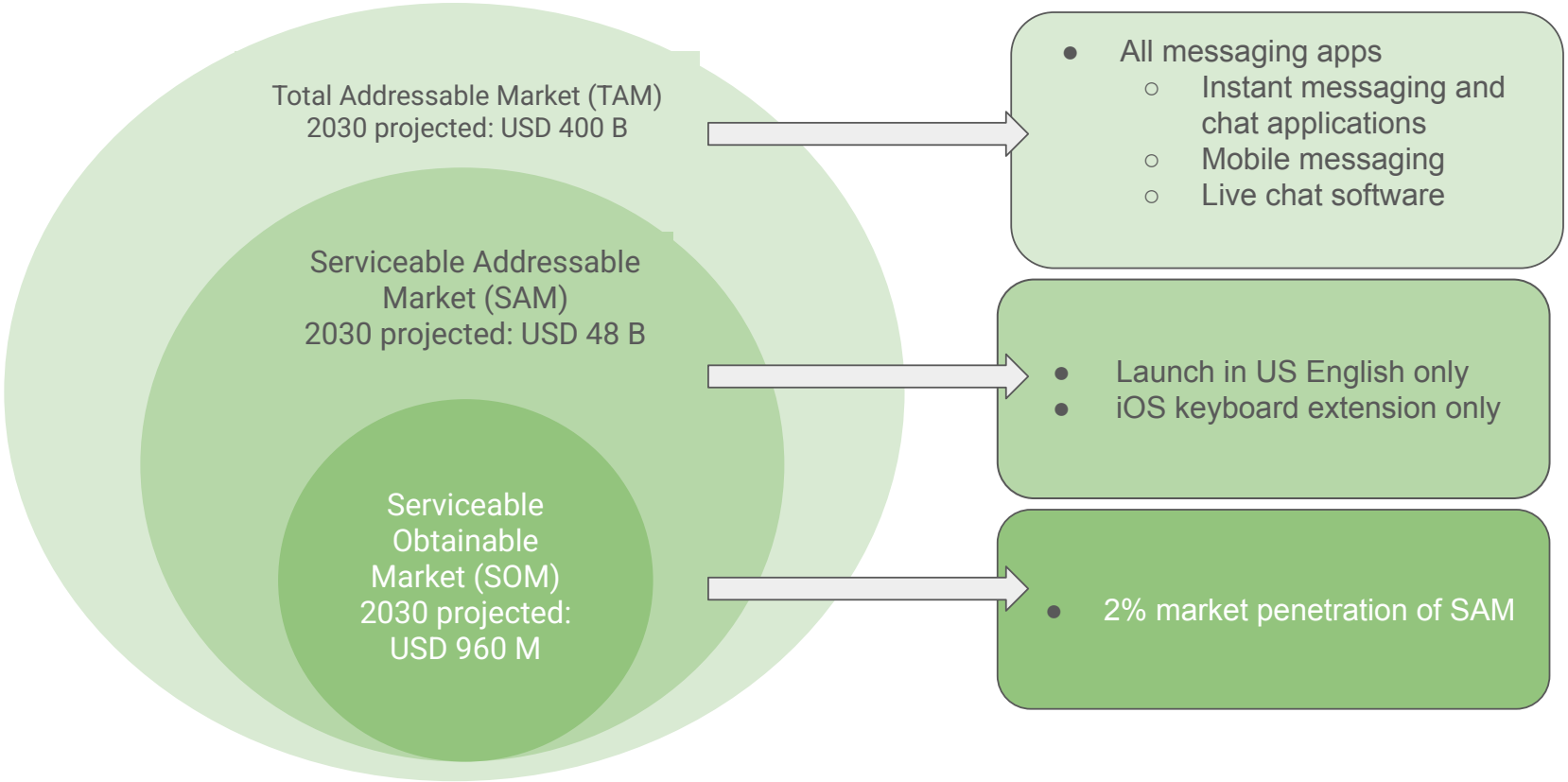
How Social Media Platforms Can Contribute to Dehumanizing Other People

More Social Media Use, Lower Empathy

How technology is harming our ability to feel empathy

UMich study: https://news.umich.edu/empathy-college-students-don-t-have-as-much-as-they-used-to/

The opportunity



https://cropink.com/gen-z-social-media-usage-statistics;https://www.verifiedmarketresearch.com/product/instant-messaging-chat-software-market/; https://www.cognitivemarketresearch.com/mobile-messaging-app-market-report

Kinder’s unique market positioning

Value Proposition	Kinder iOS	Grammarly	Inbuilt app assistants	Chatgpt/Claude/Grok
Grammar and writing style	✗	✓	✓	✓
Focus on empathetic language	✓	✗	✗	✓
In app use with instant rephrasing	✓	✓	✓	✗
Personal preference settings (such as empathy meter) 🤍💙💛🧡❤️	✓	✗	✗	✗
Requires user authentication	✓	✓	✓	✗
Collects personal, device, or usage data	✗	✓	✓	✓

Our test users love Kinder already!

What users liked

- They deeply relate to the problem
- Can use Kinder right within the messaging app of their choice

"It's like a filter for my emotions — in the best way", UX Designer, San Jose

"Kinder makes me feel confident that I'm coming across the way I want to", Student, Los Angeles

"Kinder changed how I talk to my team. helped me soften my feedback without losing clarity", Head of Data & AI, San Francisco

User privacy and ethical concerns is our priority

What are some of users' questions

- No Personal, device or usage data stores
- On device processing of data only
- T&C for iOS keyboard extension
- MVP US English only
- Nuances of empathy by culture and demographics - human data annotation and survey for train our models

"Different cultures or may have nuances in what is considered empathetic"

"Will my messages or other personal data be stored?"

"My idea of empathetic communication may vary in personal vs professional setting"

"I might speak to my Mom differently from my friend and still consider both empathetic communication"

"English language usage may vary in countries other than USA"

Demo

Empathy Defined

Empathy has three components according to psychology experts:

1. Perspective Taking[1]
2. Emotional Mimicking[2]
3. Connection between the Self and Other[3]

We want to improve empathy in online communications by non-intrusively…

→ Notifying the user when they say something that is unempathetic

→ Showing the user how much a message might cause a negative emotional reaction

→ Offer an empathetic rephrased message to prevent disconnecting with someone

1. Aragona, M. (2016)
2. Cortina, M. (2021).
3. Vilardaga, R. (2009).

Data Used and EDA

What Data are we using?

(Un)empathetic Classification

Subreddits of insults and compliments

Roastme
Toastme
FreeCompliments

Human annotated online messages from data annotation teams from both Surge-ai and Google research

“[‘NAME]’ Why are you being so sensitive”

“If awkward was a person.”

"I'm sorry that happened. Some people can be jerks sometimes. That's not your fault.”

“First off, I love your hair and choice of glasses”

What Data are we using?

(Un)empathetic Classification

Message Rephrasing

Instruction Tuning

with help from ***LLama 3.1 405b Instruct*** for synthetic data in empathetic rephrasing.

And empathetic dialogues on facebook provided by ***AI at Meta***

Meta

Data Annotation for Empathy Classifier

-2 -1 0 1 2

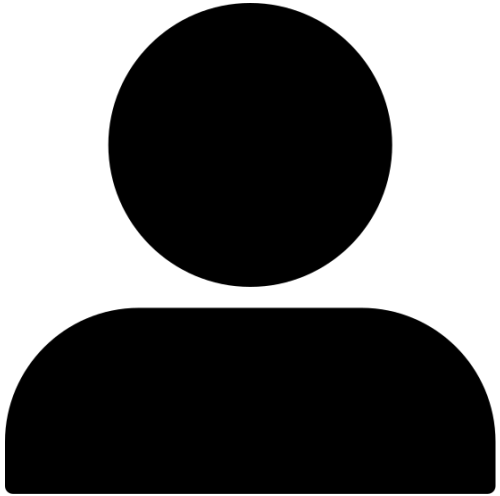
Empathetic and unempathetic messages are roughly equal in:

- message length,
- word variety,
- emojis
- and topics covered!

-2 -1 0 1 2

Extra Data from Human Respondents

Human Evaluations as Ground Truth



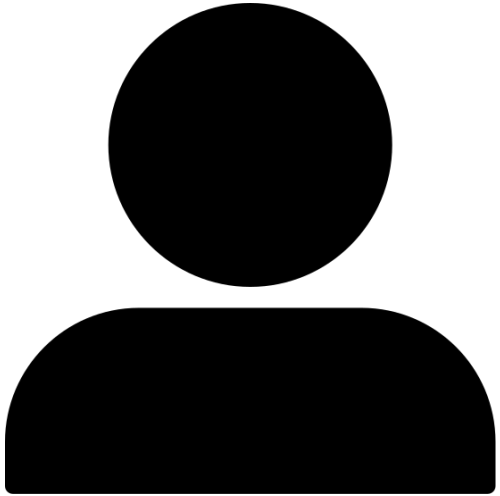
Why are you being so sensitive about this!?

Imagine you received the above message from a friend. Fill in the sentence “I would feel…”

-2	-1	0	1	2
Very Bad	Bad	Neutral	Good	Very Good

Extra Data from Human Respondents

Human Evaluations as Ground Truth



Why are you being so sensitive about this!?

Imagine you received the above message from a friend. Which of the below ways of rephrasing the message is more empathetic

Can you help me understand why this bothers you?

What about this is making you feel this way?

Data Pipelines

Online sources and annotated data from google and surge-ai

Further data collection on human labels to messages

Classifying
(Un)Empathetic Messages

LLM Re-annotation

Empathy Classifier Training

Rephrasing
Unempathetic Messages

Meta

Empathy Rephraser Training

Synthetic data generation for instruction tuning using LLama

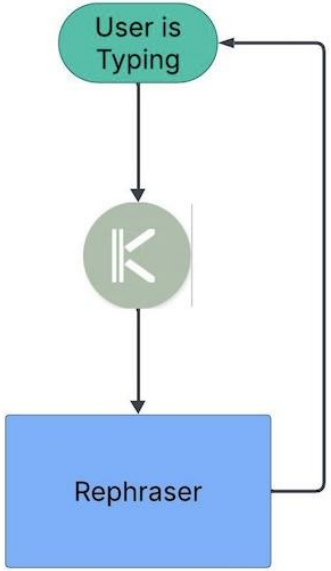
Dialogues from facebook exchanges

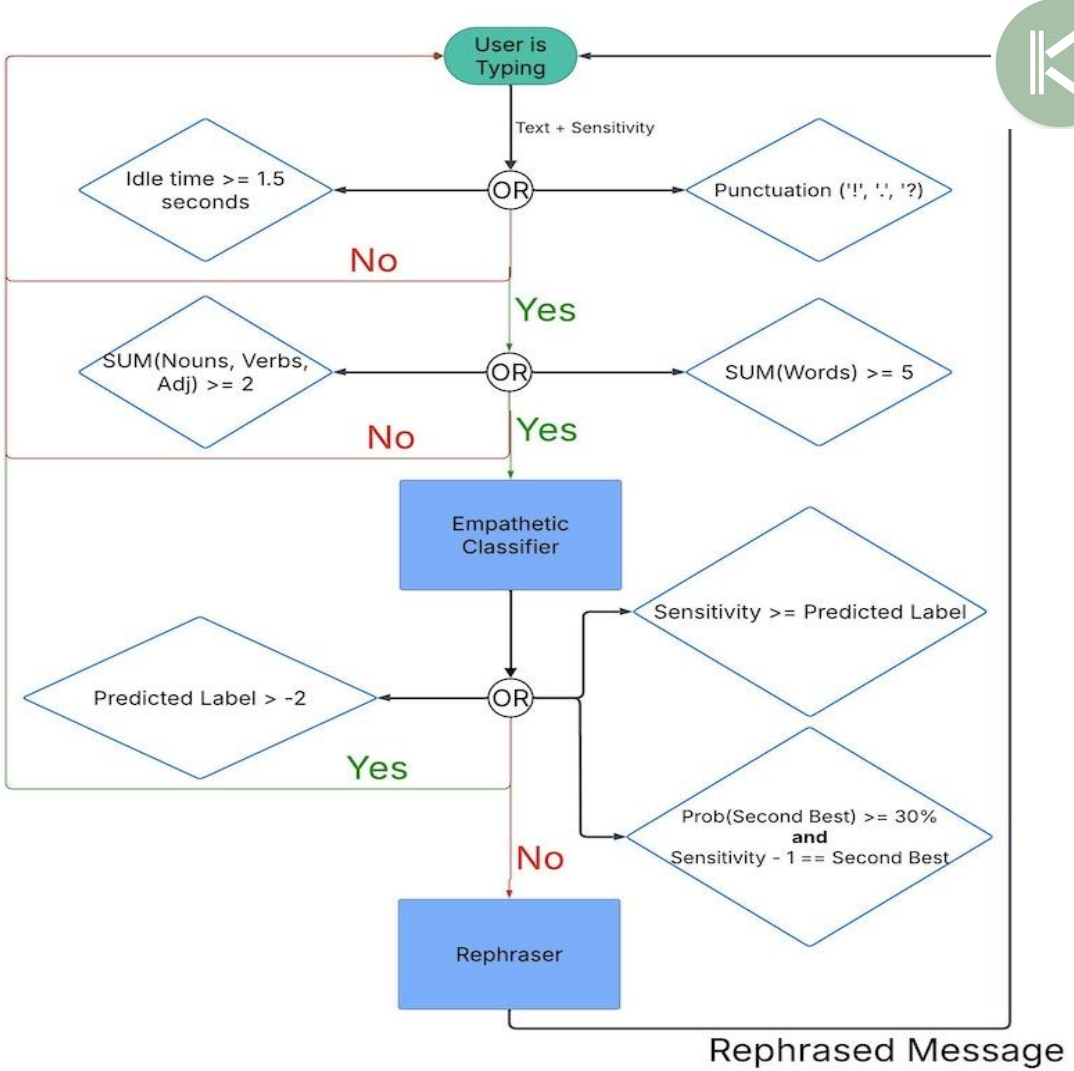
Using human qualitative testing for choosing better rephrasing

Modeling and Architecture

Logical Flow

- Classifier Trigger?
- Rephraser Trigger(s)?
- Rephraser Evaluation?
- Constraints
  - Capped at **128** tokens





Model Training

- Classifier
  - Fine tuned pardonmyai-tiny (https://huggingface.co/tarekziade/pardonmyai-tiny)
  - Avoid Catastrophic Forgetting Problem
    - Learning Rate: 0.00002
    - Warmup Ratio: 0.1
  - Loss: Tolerance Based KL-Divergence

- Rephraser
  - Instruction tuned (https://huggingface.co/distilbert/distilgpt2)
  - Shrink Model
    - Removed unused chars
    - Resized Embeddings Layer
  - Lean towards deterministic results
    - Temperature is set to 0.3
    - Max 2 ngrams
  - Loss: Cross Entropy
  - New special token in Tokenizer: [‘NAME’]

```
Tokens: ['[CLS]', 'damn', '[NAME]', 'and', 'john', 'are', 'best', 'friends', '.', '[SEP]']
Cosine similarity between [NAME] and 'John': 0.9739507436752319
```

Rephraser - Instruction Tuning

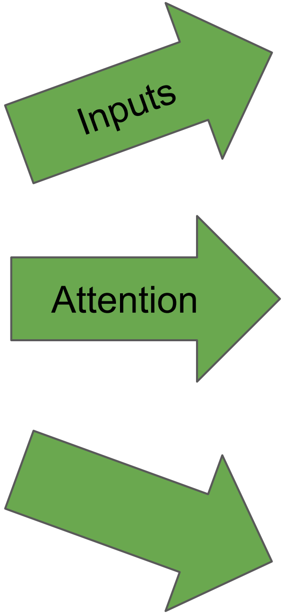
Inputs

```
tensor([21017, 46486,    25,   198,  6207, 11840,   589,   428,   284,  2128,
          517,   795,  8071,  6587,    25,   198,  2504,   338,   407,   616,
         1917,    13,   198,   198, 21017, 18261,    25,   198,    40,  1833,
          345,   821,  6476,   257,  4427,    11,   475,   314,  1101,   407,
          287,   257,  2292,   284,  1011,   319,   326,  5798,    13,   220,
        50256, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257])
```

Masked!

```
### Instruction:
Rephrase this to sound
more empathetic:
That's not my problem.

### Response:
I understand you're
facing a challenge, but
I'm not in a position
to take on that
responsibility.
<|endoftext|>
```




```
tensor([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
        1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
        1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
        0, 0, 0, 0, 0, 0, 0, 0])
```

```
tensor([ -100,  -100,  -100,  -100,  -100,  -100,  -100,  -100,  -100,  -100,
         -100,  -100,  -100,  -100,  -100,  -100,  -100,  -100,  -100,  -100,
         -100,  -100,  -100,   198, 21017, 18261,    25,   198,    40,  1833,
          345,   821,  6476,   257,  4427,    11,   475,   314,  1101,   407,
          287,   257,  2292,   284,  1011,   319,   326,  5798,    13,   220,
        50256, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257,
        50257, 50257, 50257, 50257, 50257, 50257, 50257, 50257])
```

Tolerant Boosted Kullback-Leibler Divergence Function

(1)

Label	Soft Labels (KL Divergence with Center Weight = 0.8, Tolerance = 1)
-2	[**0.8**, **0.2,** 0, 0, 0]
-1	[**0.1**, **0.8**, **0.1**, 0, 0]
0	[0, **0.1**, **0.8**, **0.1**, 0]
1	[0, 0, **0.1**, **0.8**, **0.1**]
2	[0, 0, 0, **0.8**, **0.2**]

- **Per_Label_Soft_Label_Loss[i] =** - (log(Predicted_Prob[i])) * Soft_Label[i]
- **Soft_Labels_KL_Loss** = SUM(Per_Label_Soft_Label_Loss)

---

(2)

- Boost Survey labeled Data
- Decrease AI re-annotated data
- Leave original data alone
- Confidence >= 50% only (LaaJ)

Text	Survey	Conf	Survey Boost
"They are bad for your body"	0	0.8	0
"Just go for a walk, you'll feel better"	1	1.0	1.0

- **Annotation_Weight** = Conf + Survey * (Survey Boost - Conf)

---

(3)

Label	Inverse Frequency (IF)	Boost (SB)
-2	0.4255	1.0
-1	0.9749	1.5
0	1.7428	2.2
1	1.0585	1.5
2	0.7984	1.0

- **Label_Weight** = Inverse Frequency * Boost

(4)

- **Per_Sample_Loss** = *Soft_Labels_KL_Loss* (1) * *Annotation_Weights* (2) * *Label Weights* (3)
- **Normalized_Batch_Loss** = SUM(All Per_Sample_Loss) / Batch Size

Basic Example:

True	Pred	Probabilities	Loss
0	1	[0.05, **0.1**, **0.35**, **0.4**, 0.1]	~3.56

(1) - (~~log(0.05) * 0~~ + log(0.1) * 0.1 + log(0.35) * 0.8 + log(0.4) * 0.1 + ~~log(0.1) * 0~~)
(2) 0.8 + ~~0 * (0 - 0.8)~~
(3) 1.7428 * 2.2
(4) 1.162 (1) * 0.8 (2) * 3.833 (3)

Tolerant Boosted Kullback-Leibler Divergence Function

(1)

Let:

- $t_\ell \in \{0,1,2,3,4\}$: the true class index for example ℓ, mapped from semantic labels $\{-2,-1,0,1,2\}$
- $\mathbf{T}_\ell = [T_{\ell,0}, T_{\ell,1}, \ldots, T_{\ell,4}]$: the soft target distribution over class indices for example ℓ

Soft Target Distribution (Center-Weighted)
Let $\alpha \in (0,1)$ denote the center weight, e.g., $\alpha = 0.8$. Then:

$$T_{\ell,i} = \begin{cases} \alpha, & \text{if } i = t_\ell \\ \frac{1-\alpha}{Z_\ell}, & \text{if } |i - t_\ell| = 1 \text{ and } i \in \{0,1,2,3,4\} \\ 0, & \text{otherwise} \end{cases} \quad \text{for } i \in \{0,1,2,3,4\}$$

$$\mathrm{KL}_\ell = \sum_{i=0}^{4} T_{\ell,i} \cdot \log\left(\frac{T_{\ell,i}}{P_{\ell,i}}\right)$$

---

(2)

Let:

- $w_\ell^{\text{conf}} \in [0.5, 1.0]$: confidence-based weight for example ℓ, filtered to include only predictions with at least 50% confidence
- $\delta_\ell \in \{0,1\}$: 1 if survey, 0 if not
- $w_\ell^{\text{survey}} \in \mathbb{R}^+$: fixed weight for survey-based labels ($w_\ell^{\text{survey}} = 1.0$)

Per-sample Weight:

$$w_\ell^{\text{final}} = w_\ell^{\text{conf}} + \delta_\ell \cdot \left(w_\ell^{\text{survey}} - w_\ell^{\text{conf}}\right)$$

---

(3)

Let:

- $\mathbf{b} = [1.0, 1.5, 2.2, 1.5, 1.0]$: class-specific boost factors
- $\mathbf{f} = [f_0, f_1, \ldots, f_{C-1}]$: inverse frequency weights per class
- $y_\ell \in \{0, \ldots, C-1\}$: the true class index for example ℓ

Boosted Inverse Frequency (BIF):

$$\mathrm{BIF}_\ell = \mathbf{b}_{y_\ell} \cdot \mathbf{f}_{y_\ell}$$

---

(4)

$$\mathcal{L}_{\text{batch}} = \frac{1}{N} \sum_{\ell=1}^{N} \mathrm{KL}_\ell \cdot \mathrm{BIF}_\ell \cdot w_\ell^{\text{final}}$$

Tolerant Boosted Kullback-Leibler Divergence Function

KL Divergence

Center Weight: 80%
Tolerance: 1

(Penalize predicted neighbors less)

×

Annotation Weights

Human Surveys

Developer Annotated

AI Re-annotated (LaaJ)

(Weights by Data type)

×

BIF

Label 0

Labels {-1, 1}

Labels {-2, 2}

(Weights by Sample Size)

= LOSS

(+/-) 1 Tolerant Evaluation

- **Baseline**
  - **Loss:** Categorical Cross Entropy
  - **Weighting:** Inverse Frequency

- **Final Model**
  - **Loss:** Tolerance-Boosted Kullback-Leibler
  - **Weighting:** Boosted Inverse Frequency

Text	True Label
"They are bad for your body"	-1
"Just go for a walk, you'll feel better"	-2

Label	±1 Precision (B | F)	±1 Recall (B | F)	±1 F1-Score (B | F)
-2	0.91 | **0.93**	0.64 | **0.76**	0.75 | **0.84**
-1	0.88 | **0.91**	0.57 | **0.89**	0.69 | **0.90**
0	0.60 | **0.61**	0.88 | **0.93**	0.72 | **0.74**
1	0.68 | **0.90**	0.90 | **0.92**	0.77 | **0.91**
2	0.76 | **0.85**	**0.86** | 0.76	0.80 | **0.81**

(+/-) 1 Tolerant Evaluation

Let each class label $c \in \{0, 1, 2, 3, 4\}$ corresponds to semantic labels $\{-2, -1, 0, 1, 2\}$, such that:

- TP_c: predicted label is within ± 1 of the true label c
- FP_c: predicted label is not within ± 1 of c, but was predicted as c
- FN_c: true label is c, but prediction was not within ± 1

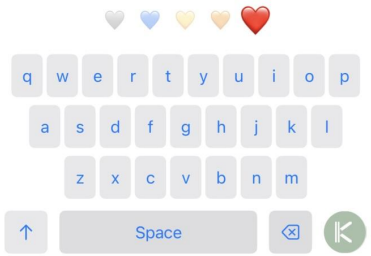
Per-class metrics:

$$\mathrm{Precision}_c = \frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FP}_c} \quad , \quad \mathrm{Recall}_c = \frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FN}_c}$$

$$\mathrm{F1}_c = \frac{2 \cdot \mathrm{Precision}_c \cdot \mathrm{Recall}_c}{\mathrm{Precision}_c + \mathrm{Recall}_c}$$

iOS Architecture

- CoreML
- SwiftUI
  - *Generic Keyboard*

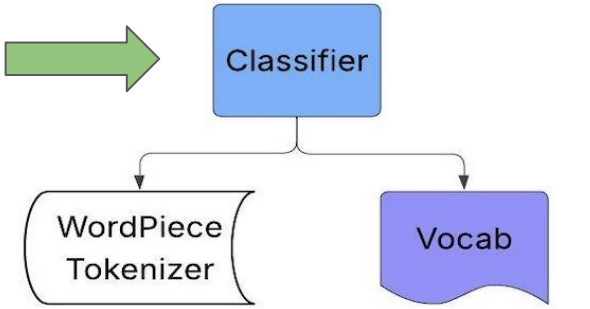


  - *Sensitivity Settings*

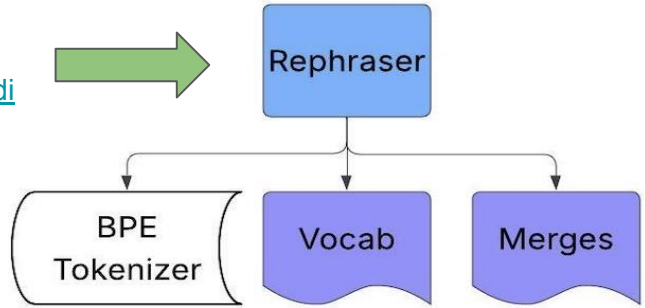
  - *Additional Rephrase Button*

★ Reminder

Fine-tuned
pardonmyai-tiny
https://huggingface.co/tarekziade/pardonmyai-tiny



Instruction-tuned
distilgpt2
https://huggingface.co/distilbert/distilgpt2



Model Quantization (Oversimplified)

1.) Filter Task via HuggingFace Exporters

$ python -m exporters.coreml –model=”<model>” –feature <task> <Output Model>

2.) Reduce output

```
class LogitsOnlyWrapper(torch.nn.Module):
    def __init__(self, base_model):
        super().__init__()
        self.base_model = base_model

    def forward(self, input_ids, attention_mask):
        logits = self.base_model(input_ids=input_ids, attention_mask=attention_mask).logits
        return logits[:, -1, :]
```

3.) Convert to Core ML

```
coreml_model = ct.convert(
    traced_model,
    source="pytorch",
    inputs=[
        ct.TensorType(name="input_ids", shape=(1, ct.RangeDim(1, 128)), dtype=np.int32),
        ct.TensorType(name="attention_mask", shape=(1, ct.RangeDim(1, 128)), dtype=np.int32)
    ],
    outputs=["logits"],
    convert_to="neuralnetwork",
    minimum_deployment_target=ct.target.iOS14,
    compute_units=ct.ComputeUnit.ALL
)
```

4.) Quantize even further

```
quantized_model = ct.models.neural_network.quantization_utils.quantize_weights(
    coreml_model,
    nbits=8,
    quantization_mode="linear"
)
```

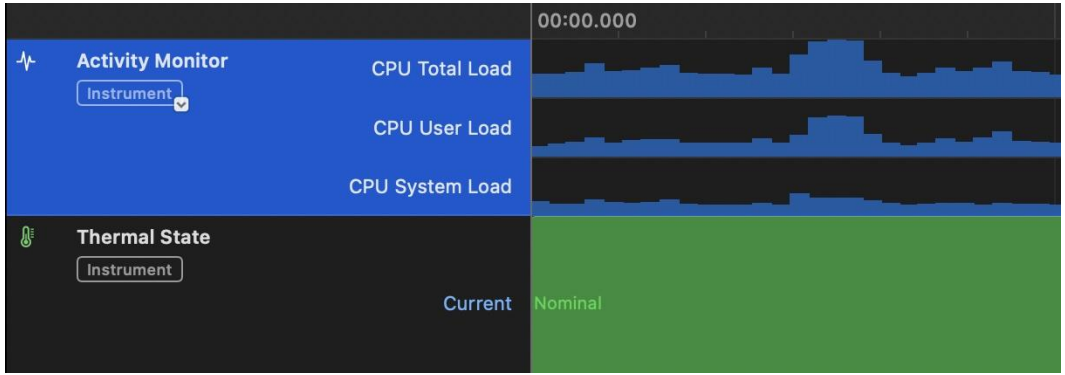
5.) Compile

$ xcrun coremlc compile <Model>

Model	Best Size (mb)
Classifier	28
Rephraser	~100-150

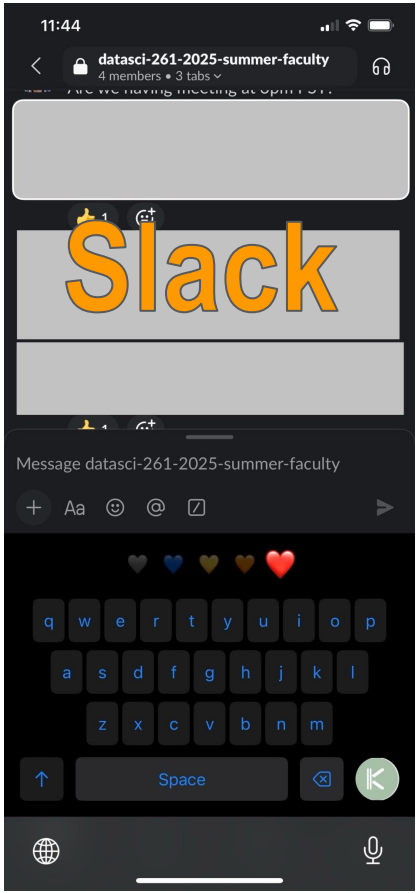
Instrumentation and Profiling

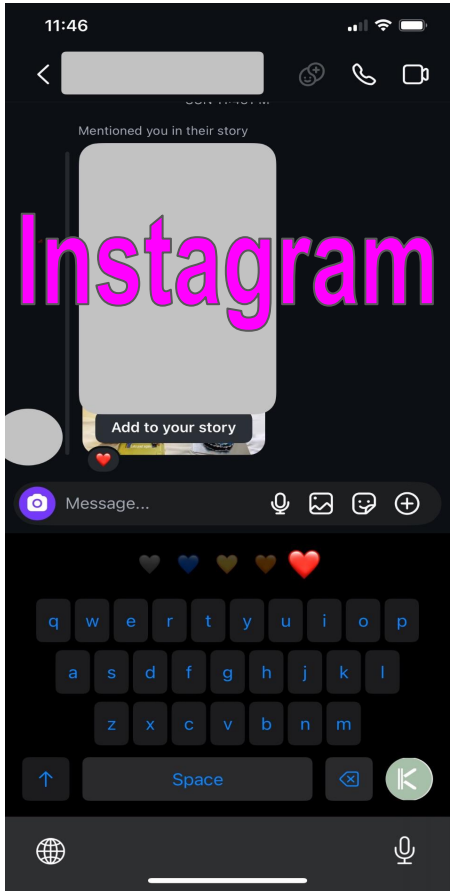
- Device Tested
  - iPhone 16 Pro Max
  - iOS 16+
- Hardware Requirements
  - CPU Only - Any device that supports iOS 14
  - ANE/GPU - iPhone 11+
- Evaluation Time
- Thermal looks good!

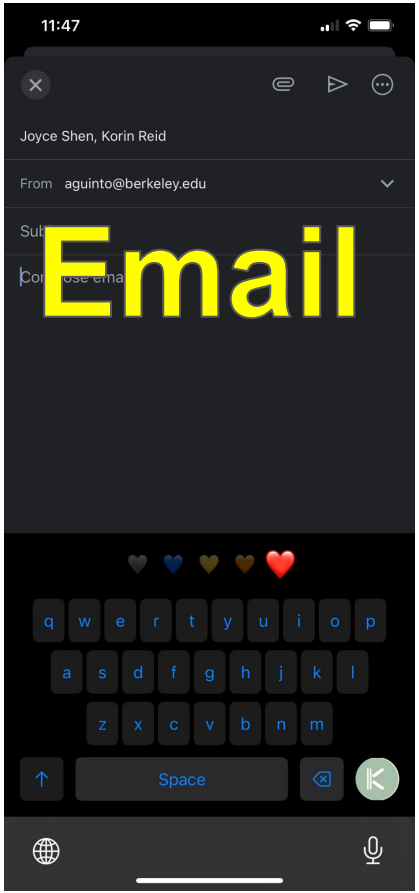


Hardware	Classifier Inference(s)	Rephraser Inference (s)
CPU	0.20546603202819824	5.127754092216492
ANE/GPU	0.16854095458984375	4.187222957611084
ANE/GPU with Padding	0.19244205951690674	7.1593159437179565

Device Installation - Cross Application, Cross Device, etc.







What's Next?

Algorithm Development

- More surveys and re-annotations for training
- Find optimal parameters for the Tolerant-Boosted KL Divergence Function
  - Boost values, Center Weight, and Survey weights have room for experimentation
- Update evaluation metric to include (+/-) 1 IFF probability >= Threshold

Application Specific

- Address demographic and cultural bias as part of loss and evaluation
- Address UI Edge cases
- Integrate onto Android

Academic Endeavors

- Fully Homomorphic Encryption (FHE) and Reinforcement learning from Human Feedback (RHLF)
- Submit at an NLP (or Cryptography) conference

Team Kinder Pledge

“We aim to empower users to communicate more empathetically by" offering real-time and human-backed rephrasing suggestions.”

*“We aim to make empathetic rephrasing more widely **available** without consumers purchasing additional hardware.”*

*“We aim to preserve user **privacy** by ensuring that only the user can view their own data in plaintext.”*

Q&A

Appendix

T&C Page 1

Kinder – Terms and

Last updated: 6/30/2025

Please read these Terms and Conditions (“Terms”) carefully before downloading or using Kinder, the iOS keyboard-extension that offers real-time, AI-powered empathy suggestions (“App,” “Kinder,” “we,” “our,” or “us”). By installing or using Kinder you acknowledge that you have read, understood, and agree to be bound by these Terms and by our separate Privacy Policy, which is incorporated by reference.

1. Acceptance of the Terms

- If you do not agree with these Terms, do not install or use Kinder.
- We may update these Terms periodically. Material changes will be communicated through the App or by other reasonable means, and continued use constitutes acceptance of the revised Terms.

2. Service Description

Kinder analyzes text you type inside the keyboard to (i) flag language that may be perceived as unkind, and (ii) suggest alternative, more empathetic phrasing. Processing is performed primarily on-device; however, if you opt-in, certain prompts may be sent to cloud models for enhanced accuracy.

3. Eligibility

You must be at least 13 years old and legally able to form a binding contract. If you are under 18, you affirm that you have parental or guardian consent.

4. License Grant & Intellectual Property

We grant you a personal, revocable, non-exclusive, non-transferable license to use Kinder solely for its intended purpose on iOS devices you own or control. All intellectual-property rights in Kinder and its content remain ours or our licensors’.

5. User Responsibilities & Restrictions

- Do not reverse-engineer, decompile, or create derivative works of Kinder.
- Do not use Kinder to generate or transmit unlawful, harassing, defamatory, or hateful content.
- You remain solely responsible for the messages you send, even when you accept Kinder’s suggestions.

6. Privacy & California-Specific Disclosures

6.1 Categories of Personal Information Collected

T&C Page 2

6.2 Notice at Collection (Cal. Civ. Code §
At or before the point of collection we inform you of the categories of any personal information and the purposes for which they are used. This notice is provided within these Terms and the in-app onboarding screen.

6.3 California Consumer Privacy Act (CCPA/CPRA) Rights
If you reside in California, you have the right to:

- Know the categories and specific pieces of personal information we collect, use, or disclose;
- Request deletion of personal information (with statutory exceptions);
- Correct inaccurate personal information;
- Opt-out of the “sale” or “sharing” of personal information (Kinder does not sell or share your data for cross-context behavioral advertising);
- Limit the use of sensitive personal information; and
- Be free from discrimination for exercising CCPA rights.

You may exercise these rights in-app or by emailing privacy@kinder.ai.

6.4 CalOPPA – Online Privacy Protection

- Kinder maintains a conspicuously posted, accessible Privacy Policy that discloses our data practices, the effective date, and how we notify users of material changes.
- “Do-Not-Track” Signals: Browsers may transmit DNT requests. We currently do not track users for advertising and therefore do not respond differently to DNT signals; this is disclosed as required by CalOPPA.

6.5 Data Retention & Deletion
We retain on-device data only while necessary for core functionality, and cloud-processed data is deleted after completion of inference. We apply least-retention and surprise-minimization principles recommended by the California AG for mobile apps.

7. AI Output & Disclaimers

- Suggestions are generated statistically and may be inaccurate or inappropriate in certain contexts; you should review and modify any text before sending.
- Kinder does not provide legal, medical, or psychological advice.

8. Third-Party Services
The App may interface with Apple systems and optional cloud-AI providers. Your use of third-party services is subject to their separate terms and

Acknowledgements and additional resources

The Kinder team extends gratitude to:

- **UC Berkeley MIDS Program** for providing the capstone framework and academic support
- **Data Annotation Partners**: Surge AI and Google Research teams for high-quality human annotations
- **Open Source Community**:
  - Meta AI for the Empathetic Dialogues dataset
  - Hugging Face for model hosting and tools
  - Reddit communities for authentic conversational data
- **Technical Infrastructure**:
  - Apple CoreML team for mobile AI optimization tools
  - LLaMA and InstructLab for synthetic data generation capabilities
- **User Testing Participants** who provided invaluable feedback during development

Github with the code
https://github.com/NLPwned/Empathetic_Datasets_210_Capstone/tree/main/

Project Website
https://priyau.wixsite.com/writekindly

Human Survey Template Doc
https://github.com/NLPwned/Empathetic_Datasets_210_Capstone/blob/main/Empathic_Communication_-_Responses.docx

DISCOVER

HAVE AI LANGUAGE MODELS ACHIEVED THEORY OF MIND? [DISCOVER MAGAZINE]

New Technology > Robots

AI Has Suddenly Evolved to Achieve Theory of Mind

In a stunning development, a neural network now has the intuitive skills of a 9-year-old.

Theory of Mind AI: Bringing Human Cognition to Machines

Neil Sahota Published: July 29, 2024 Artificial Intelligence

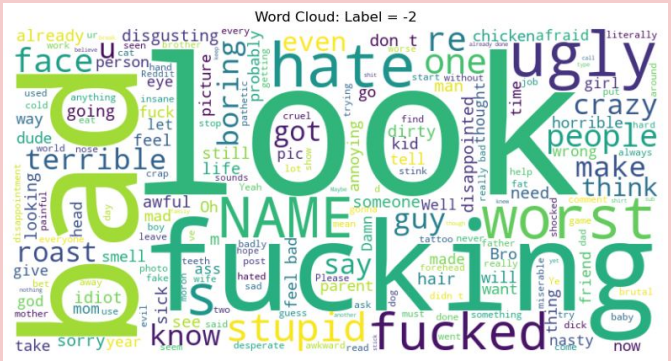
Rubinet al., (2025)
Strachan, J.W.A., Albergo, D., Borghini, G. *et al (2014)*

EDA - Our data is looking good!

Negative and positive messages are equivalent in:

Message Length	($p < .001$)
Emoji Usage	($p < .01$)
Word Variety	($p < .001$)
Relevant Topics	($p < .05$)

Very Negative Class



Very Positive Class

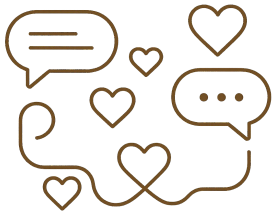
We are operating in a blue ocean

	Integrated within messaging apps	Standalone Apps
Grammar, polish, and writing style	Grammarly	Inbuilt AI writing assistant in messaging apps (such as LinkedIn Messenger)
Option for empathetic writing	Kinder	ChatGPT Claude Grok

Our value proposition

All users of any ios messaging app. ios keyboard extension and can work right in messaging apps

As users type their message, will detect and flag words or phrases that don't read very kindly and get instant rephrasing suggestions



Want to write kindly with confidence, without overthinking

Empathy meter: personal preference

Images generated on canva and Perplexity