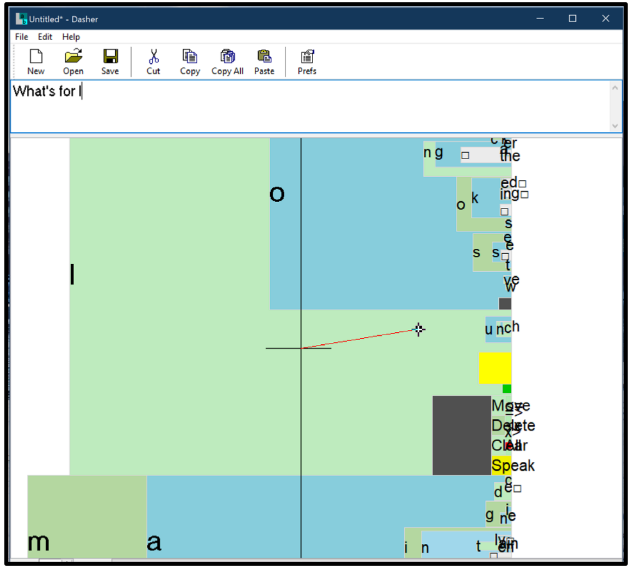
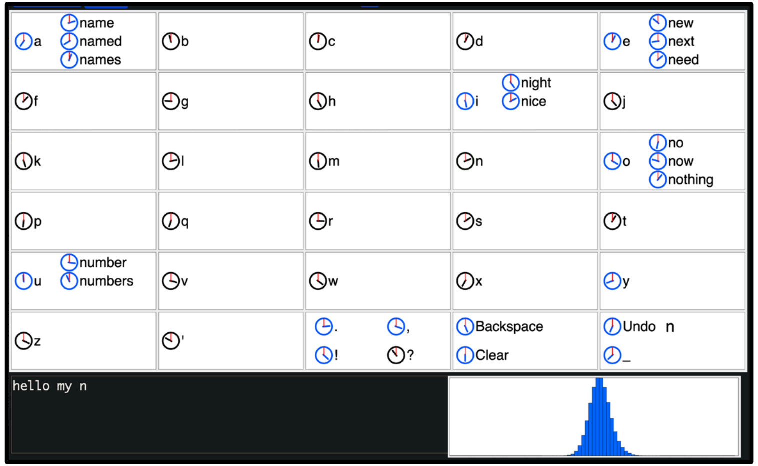
Language Modeling and Predictive Text Entry for AAC

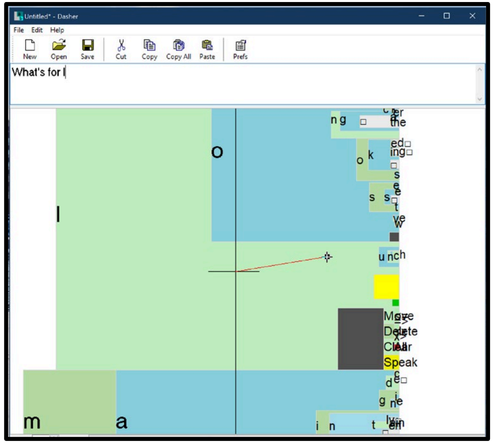


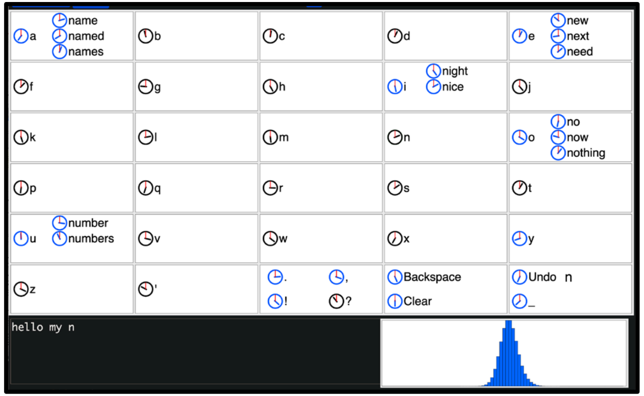


Keith Vertanen
vertanen@mtu.edu
https://keithv.com
Michigan Tech

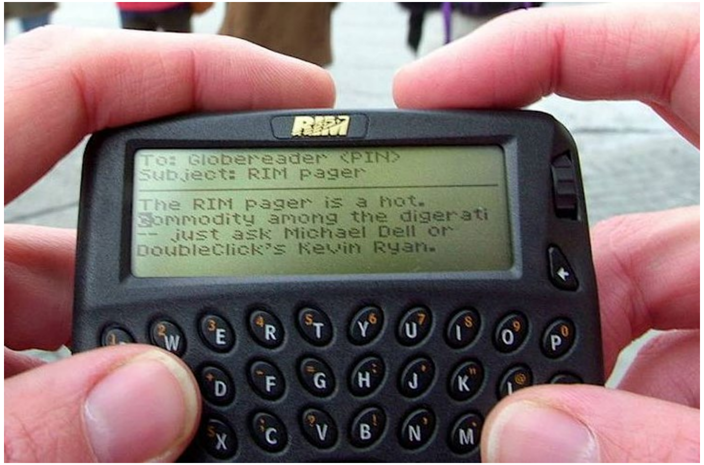
Overview

- **Language modeling**
  - N-gram language models
    - Motivating example: Auto-correcting keyboard
  - Dasher interface
  - Adapting to a user's text
  - Neural network language models
  - Selecting training data
- **Word predictions**
  - Impact of different design choices
  - Single switch interface: Nomon
- **Future directions**

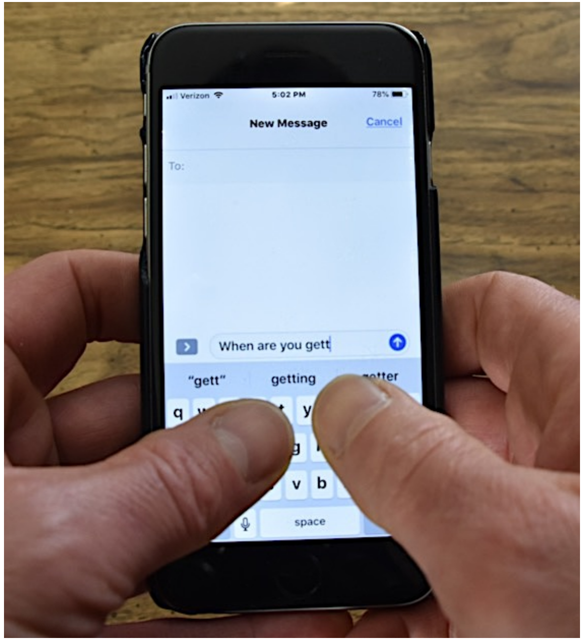


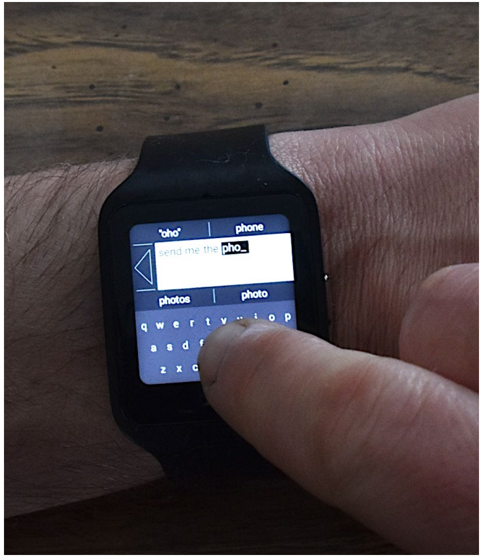


Typing on small keyboards with fat fingers...



https://www.theglobeandmail.com/report-on-business/small-business/sb-managing/rim-is-cheap-but-so-what/article616857/





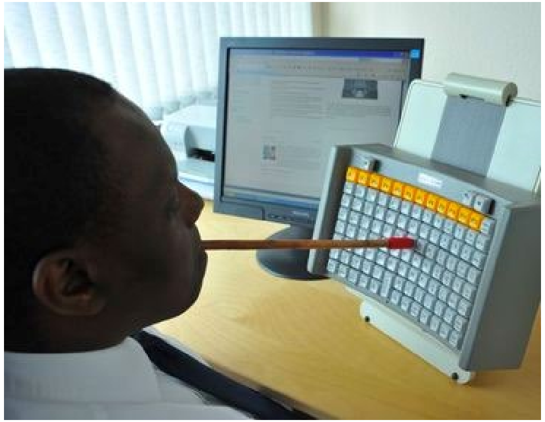
History:

1996 – RIM pager (later known as Blackberry)

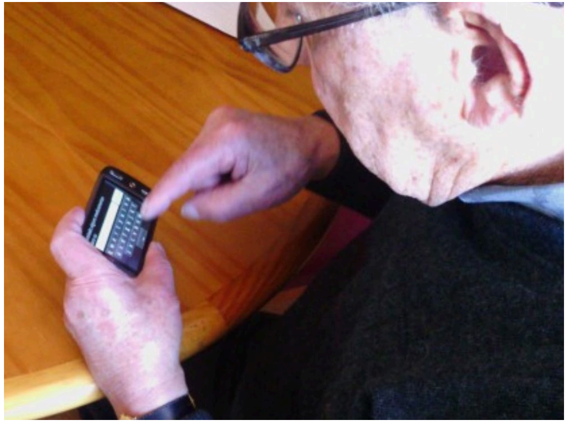
2007 – Release of first iPhone

2015 – Release of first Apple Watch

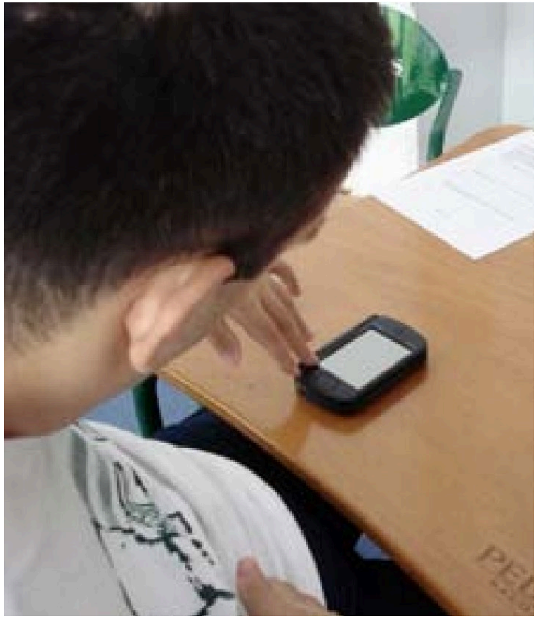
Typing if you're motor-impaired...



Person typing on a physical keyboard with a mouth stick[1].



Elderly participant typing on a touchscreen keyboard[2].



Motor-impaired user interacting with a touchscreen interface[3].

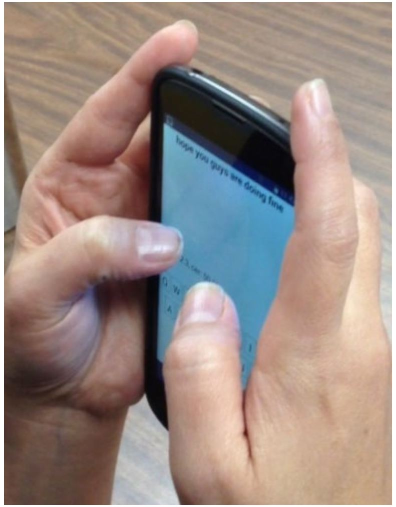
[1] https://www.maltron.com/head-mouth-stick-keyboard-info.html

[2] Nicolau, H. and Jorge, J. Elderly Text-Entry Performance on Touchscreen. In *Proc. ASSETS'12*.

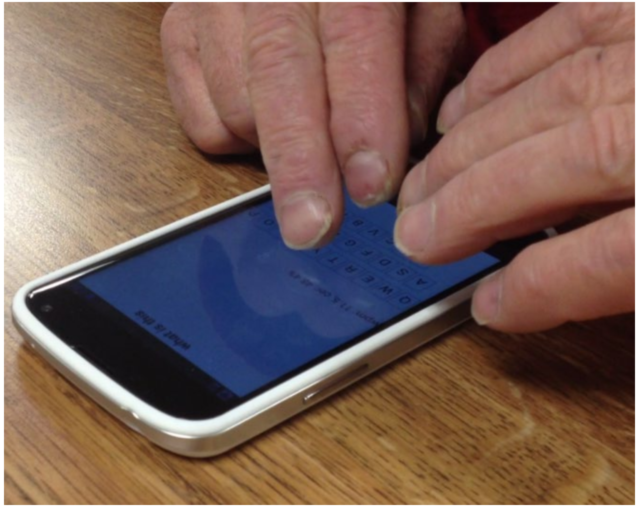
[3] Guerreiro, T., et al. Towards Accessible Touch Interfaces. In *Proc. ASSETS'10*.

Typing if you're visually-impaired...

Undergrad research assistant typing on a virtual keyboard while blindfolded.



Legally blind participant typing on a virtual keyboard.



Completely blind participant typing on a virtual keyboard.

Vertanen, K., et al. The Feasibility of Eyes-Free Touchscreen Keyboard Typing. In *Proc. ASSETS 2013*.

Can you guess what I'm writing?

Key nearest each tap: have a fod dat

More likely answer: **have a good day**

Other possibilities: gave a good dat

have a good dat

have a good date

have a god day

. . .

Scoring possible hypotheses

Based on our observed taps,
we had a set of possible sentences:


```
have  a  good  day
have  a  fod   dat
gave  a  good  dat
have  a  good  dat
have  a  good  date
have  a  god   day
...
```

We would like the compute to **assign a score** to each sentence.
This will allow us to **select the best one**, the best three, ...

Probability of an event

P(•) denotes the ***probability*** of some event

It is a value between `0.0` and `1.0`

Larger value = more likely event


```
P(have a good day)  = 0.671
P(have a fod  dat)  = 0.101
P(gave a good dat)  = 0.010
P(have a good dat)  = 0.010
P(have a good date) = 0.004
P(have a god  day)  = 0.003
  ...
```

After assigning a probability to each sentence, we might assume the one with highest probability is what they actually wanted to write.

Conditional probability

P(A | B) denotes:
probability of event A given B

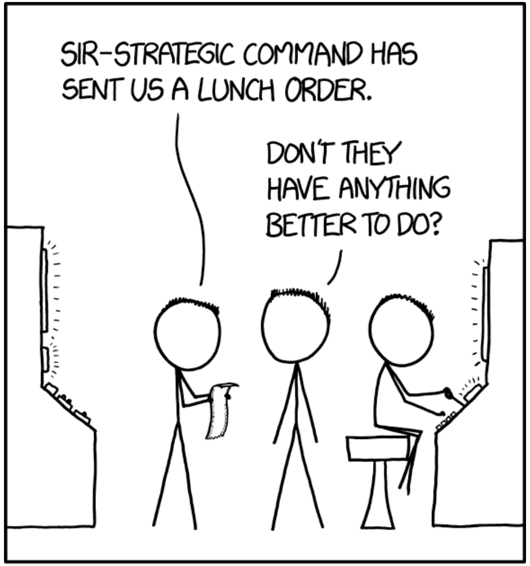
$P(\text{have a good day} \mid \text{tap}_1, \text{tap}_2, \ldots \text{tap}_{14}) = 0.671$

Given I saw the user perform this sequence of 14 taps,
what is the probability they wanted to write "have a good day"?

Auto-correction algorithm

Algorithm:
Search for the most probable text given the taps

$$P(Letters|Taps) \propto \underbrace{P(Taps|Letters)}_{\substack{\text{input}\\ \text{model}}} \underbrace{P(Letters)}_{\substack{\text{language}\\ \text{model}\\ \text{(LM)}}}$$



EVERYONE COMPLAINS ABOUT AUTOCORRECT, BUT WE FORGET ABOUT THE TIME IT PREVENTED A NUCLEAR WAR.

https://xkcd.com/1834/

Some possible input models

```
P(user hit key "p" | output character "p")
```

Deterministic: "Whatever key I hit, that is the right one"

```
P(user hit key "p" | output character "p") = 1.0
P(user hit key "p" | output character "o") = 0.0
P(user hit key "p" | output character "l") = 0.0
...
P(user hit key "p" | output character "z") = 0.0
```

Examples: Physical keyboards, some dwell keyboards or row-column scanners?

space	E	A	R	D	U
T	O	I	L	G	B
N	S	F	Y	V	J
H	C	P	K	Q	.
M	W	X	Z	,	?

Some possible input models

```
P(user hit key "p" | output character "p")
```

Adjacent: "Whatever key I hit, that is probably right but it could be an adjacent key"

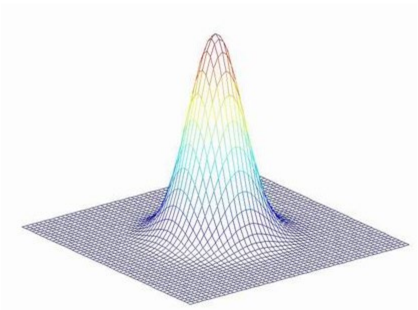
```
P(user hit key "p" | output character "p") = 0.8
P(user hit key "p" | output character "o") = 0.1
P(user hit key "p" | output character "l") = 0.1
P(user hit key "p" | output character "i") = 0.0
P(user hit key "p" | output character "k") = 0.0
...
P(user hit key "p" | output character "z") = 0.0
```

A simple input model for auto-correct

Algorithm:

Search for the most probable text given the taps

$$P(Letters|Taps) \propto \boxed{\underbrace{P(Taps|Letters)}_{\text{input model}}} \underbrace{P(Letters)}_{\text{language model}}$$



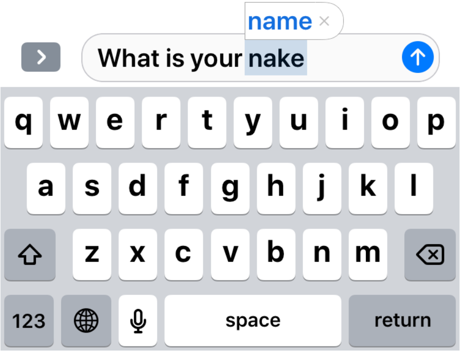
2D Gaussian centered at each key

P(12|or)=0.123
P(12|ot)=0.123
...
P(12|ir)=0.097
P(12|it)=0.097
...
p(12|zy)=0.00000003
P(12|zz)=0.00000001

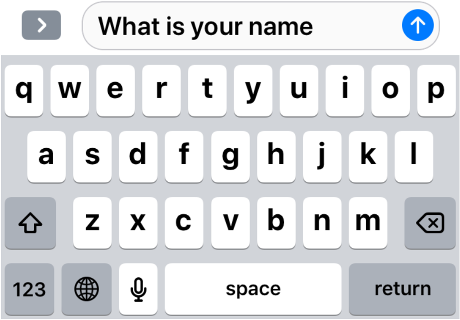
26^2=676 possibilities

Terminology: Auto-correct vs. Word predictions

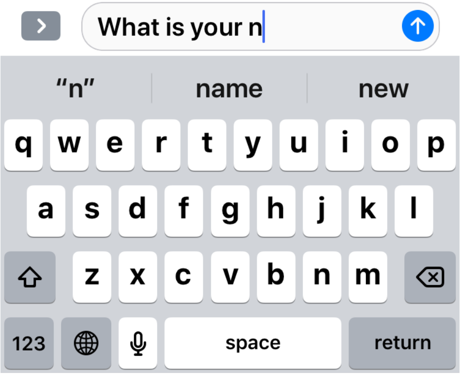
Auto-correct: Correcting errors made in last bit of typing



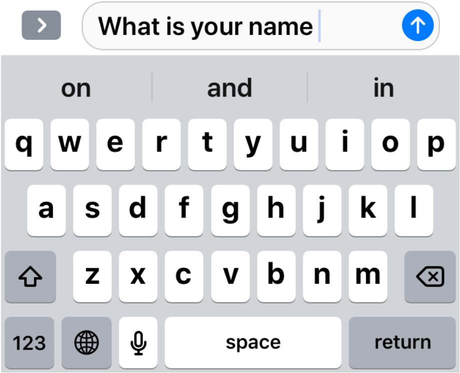
Pressing spacebar



Word predictions: Guessing next word based on typing thus far



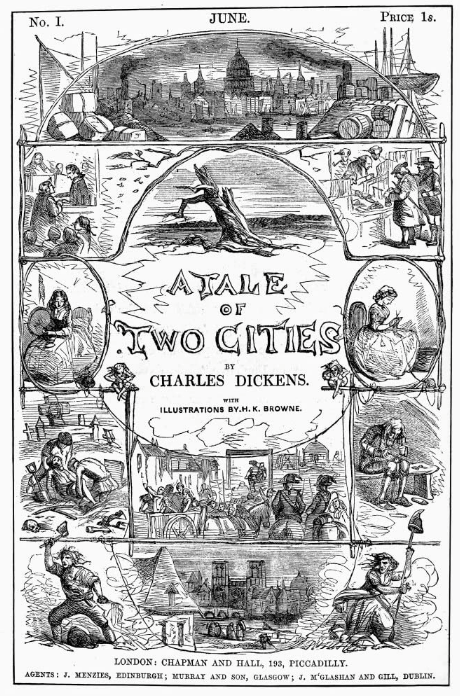
Pressing button above keyboard



Training a language model

Step 1: Get a whole bunch of text

It was the best of times, it was the worst of times,
it was the age of wisdom, it was the age of foolishness,
it was the epoch of belief, it was the epoch of incredulity,
it was the season of Light, it was the season of Darkness,
it was the spring of hope, it was the winter of despair,
we had everything before us, we had nothing before us,
we were all going direct to Heaven, we were all going direct
the other way--in short, the period was so far like the present
period, that some of its noisiest authorities insisted on its
being received, for good or for evil, in the superlative degree
of comparison only.



Training a language model

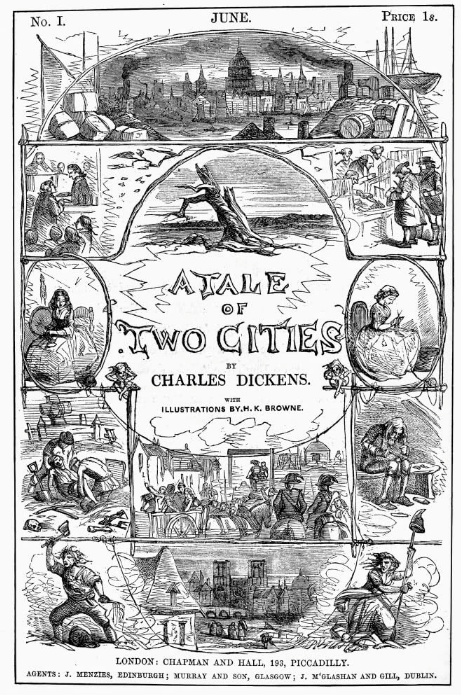
Step 2: Normalize text (some possible choices)

Convert to lowercase

Remove symbols

Normalize whitespace

```
it was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```



A simple word language model

Step 3: Count things up

Train a ***unigram*** (1-gram) word language model

Count occurrences of each unique word

```
it was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```

A simple word language model

Step 3: Count things up

"it" never seen before, add to table

unigram	count
it	1

```
it was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```

A simple word language model

Step 3: Count things up

"was" never seen before, add to table

unigram	count
it	1
was	1

```
it was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```

A simple word language model

Step 3: Count things up

`"times"` (fast forward)

unigram	count
it	1
was	1
the	1
best	1
of	1
times	1

```
it was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```

A simple word language model

Step 3: Count things up

"it" was already added, increment its count

unigram	count
it	2
was	1
the	1
best	1
of	1
time	1

```
it was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```

A simple word language model

Step 4: Calculate probabilities

Finished going through text, how probable is each word?

```
total = 119 words
P(word) = count / total
```

```
it was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```

unigram	count	prob
the	14	0.118
of	12	0.101
was	11	0.092
it	10	0.084
we	4	0.034
age	2	0.017
us	2	0.017
had	2	0.017
going	2	0.017
were	2	0.017
its	2	0.017
for	2	0.017
times	2	0.017
season	2	0.017
...		
or	1	0.008
sum	119	1.000

A simple word language model

Step 5: Use the language model

Helps decide between competing text hypotheses

$$P(Letters|Taps) \propto \underbrace{P(Taps|Letters)}_{\text{input model}} \underbrace{P(Letters)}_{\text{language model}}$$

P(12|or)=0.123
P(12|ot)=0.123
...
P(12|ir)=0.097
P(12|it)=0.097
...
P(12|zz)=1.2e-6

unigram	count	prob
the	14	0.118
of	12	0.101
was	11	0.092
it	**10**	**0.084**
we	4	0.034
age	2	0.017
us	2	0.017
had	2	0.017
going	2	0.017
were	2	0.017
its	2	0.017
for	2	0.017
times	2	0.017
season	2	0.017
...		
or	**1**	**0.008**
sum	119	1.000

Language modeling for the win???

Assume 2 *new* taps:

unigram	count	prob
the	14	0.118
of	12	0.101
was	11	0.092
it	10	0.084
we	4	0.034
age	2	0.017
us	2	0.017
had	2	0.017
going	2	0.017
were	2	0.017
its	2	0.017
for	2	0.017
times	2	0.017
season	2	0.017
...		
or	1	0.008
sum	119	1.000

<u>What word comes after</u>:

`i would like a big helping` ← "of" most probable

`can you please help` ← "us" makes sense, but unigram votes "of"

Problem 1: Fixed context length

`i will go either to the morning` ← Need even longer context!

`the ego superego and` ← "id" makes sense, but never observed

Problem 2: Sparsity

Bigram word language model

bigram	count
it was	1

Step 3 (revisited): Count things up

Train a ***bigram*** (2-gram) word language model

An n-gram language model, here n=2

Count all contiguous pairs of words

```
it was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```

Bigram word language model

Step 3 (revisited): Count things up

Train a ***bigram*** (2-gram) word language model

An n-gram language model, here n=2

Count all contiguous pairs of words

```
it was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```

bigram	count
it was	1
was the	1

Using bigram word language model

What comes after:

`can you send me the` $\boxed{?}$

P(next-word | the) = ???

11 choices, but most are bad

More training data?

Definitely a good idea!

Longer context length?

3-gram, 4-gram, 5-gram, ...

send me the present

send me the email

...

Need 4-gram or longer language model to see "send"

bigram	prob
the age	0.017
the epoch	0.017
the season	0.017
the best	0.008
the other	0.008
the period	0.008
the present	0.008
the spring	0.008
the superlative	0.008
the winter	0.008
the worst	0.008

Bigrams starting with "the"

bigram	count	prob
was the	10	0.085
it was	10	0.085
age of	2	0.017
going direct	2	0.017
were all	2	0.017
all going	2	0.017
times it	2	0.017
season of	2	0.017
epoch of	2	0.017
the age	2	0.017
we had	2	0.017
we were	2	0.017
of times	2	0.017
the season	2	0.017
...		
insisted on	1	0.008
sum	118	1.000

Full table of bigrams

Scaling up: Bigger word language models

- **Training details**
  - Project Gutenberg: **1.2B words of text**
  - 100K word vocabulary

n-gram size	Compressed size (MB)	Total n-grams (M)	Train time (mins)	Train memory (GB)
1-gram	0.5	0.1	4	0.03
2-gram	96	19	9	0.7
3-gram	918	158	23	6
4-gram	3,767	531	49	25
5-gram	8,827	1,123	99	63
6-gram	15,620	1,837	203	115
7-gram	23,609	2,586	410	176

Using large 3-gram word language model

What comes after:

can you send me the ?

P(next-word | me the) = ???

```
3-gram
me the most
me the truth
me the other
me the whole
me the same
me the honour
me the way
me the story
me the first
me the more
me the greatest
me the best
me the money
me the next
me the following
me the honor
...
```

most frequent

less frequent

Top 3-grams starting "me the" (5408 total)

Using large 4-gram word language model

What comes after:

`can you send me the ?`

`P(next-word | send me the) = ???`

```
4-gram
send me the money
send me the letter
send me the name
send me the first
send me the necessary
send me the bill
send me the manuscript
send me the means
send me the account
send me the balance
send me the book
send me the latest
send me the length
send me the most
send me the names
send me the news
...
```

most frequent

less frequent

Top 4-grams starting "send me the" (181 total)

Using large 5-gram word language model

What comes after:

`can you send me the ?`

`P(next-word | you send me the) = ???`

5-gram	
you send me the length	most frequent
you send me the baptismal	
you send me the green	
you send me the address	
you send me the answers	
you send me the article	
you send me the customary	
you send me the fifty	
you send me the hull	
you send me the manuscript	
you send me the money	
you send me the name	
you send me the proper	
you send me the register	
you send me the right	
you send me the six	less frequent

5-grams starting "you send me the" (16 total)

Same idea, but with characters...

Step 3 (revisited): Count things up

Example: bigram (2-gram) ***character*** language model

Sweep a 2 character window through the text

```
it was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```

Bigram character language model

bigram	count
it	1

Step 3 (revisited): Count things up

"it" never seen before

Add to table

```
it was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```

Bigram character language model

Step 3 (revisited): Count things up

"t_" never seen before

Add to table (using _ to denote space)

bigram	count
it	1
t_	1

```
it_was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```

Bigram character language model

Step 4: Calculate probabilities

Final counts on this tiny text (sorted by count)

Calculate probabilities as with word model

```
it was the best of times it was the worst of times it was the
age of wisdom it was the age of foolishness it was the epoch of
belief it was the epoch of incredulity it was the season of
light it was the season of darkness it was the spring of hope
it was the winter of despair we had everything before us we had
nothing before us we were all going direct to heaven we were
all going direct the other way in short the period was so far
like the present period that some of its noisiest authorities
insisted on its being received for good or for evil in the
superlative degree of comparison only
```

bigram	count
e_	29
_w	21
s_	20
t_	19
th	19
_t	18
_o	16
he	16
_i	15
it	14
as	13
f_	13
of	12
wa	12
in	11
re	10
...	
sum	592

Using bigram character language model

What comes after:

he needs to be educate$\boxed{?}$

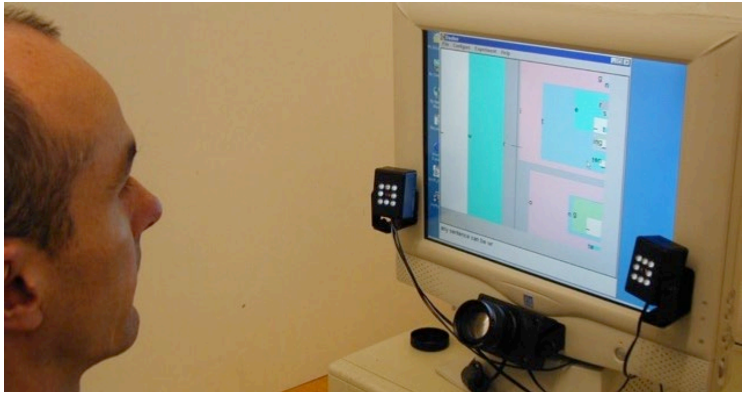
P(next-char | e) = ???

bigram	count	prob
...		
e_	29	0.049
es	9	0.015
er	8	0.014
ea	3	0.005
ec	3	0.005
ed	3	0.005
ef	3	0.005
ei	2	0.003
en	2	0.003
ep	2	0.003
ev	2	0.003
ee	1	0.002
eg	1	0.002
el	1	0.002
...		

Bigrams starting with letter e

Example of character language modeling: Dasher

- **Dasher**
  - Write by navigating through nested letter boxes
    - Via mouse, eye/head tracker, buttons, touchscreen, ...
    - Box size based on letter's probability under a character LM
  - Anything can be written (but unlikely text takes longer)

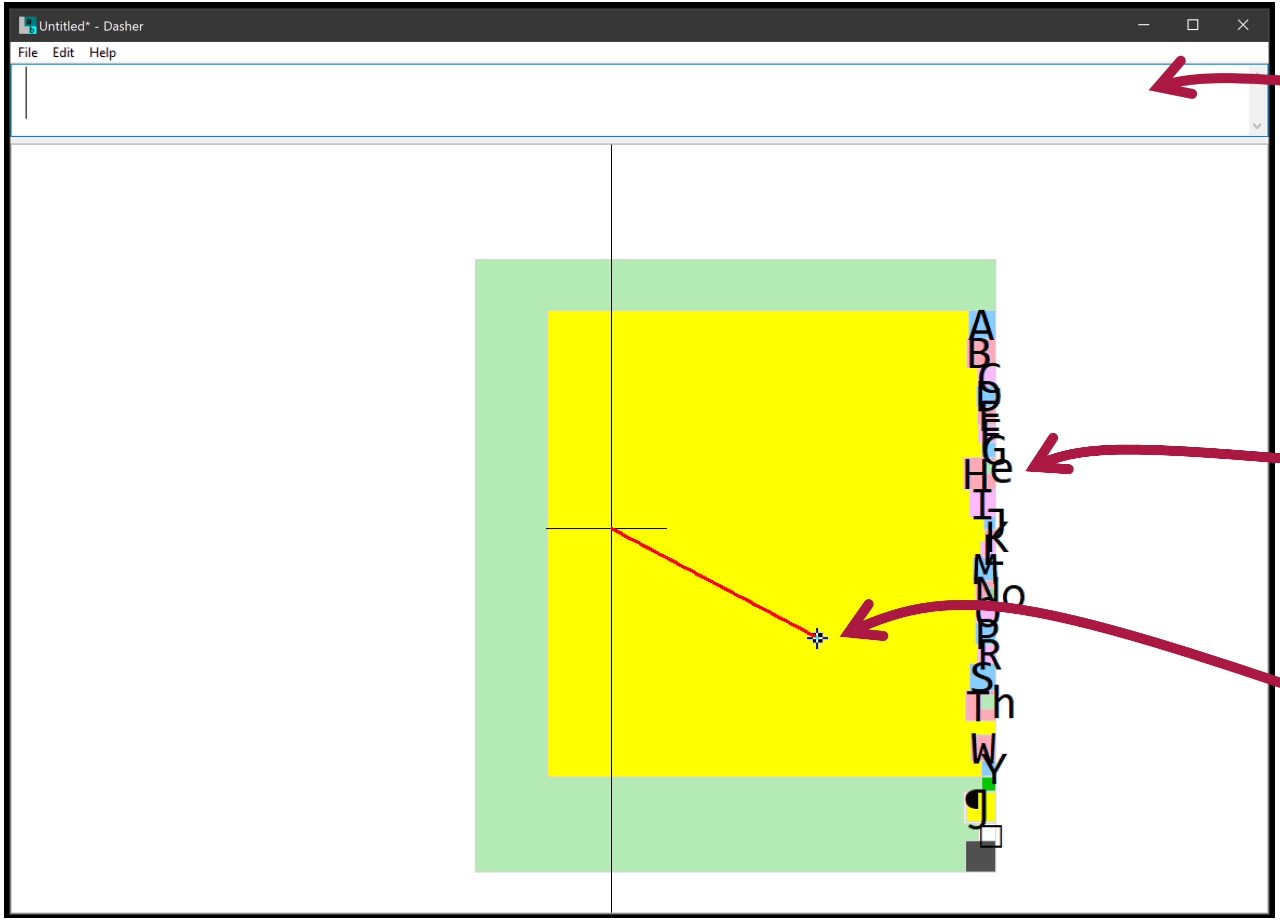


David MacKay (inventor of Dasher) writing using an eye-tracker.

Steve Saling showing off his Dasher front-facing screen.

Ward, D. et al. Dasher - a Data Entry Interface Using Continuous Gestures and Language Models. In *Proc. UIST 2000*.

Writing "The" with Dasher, step 1

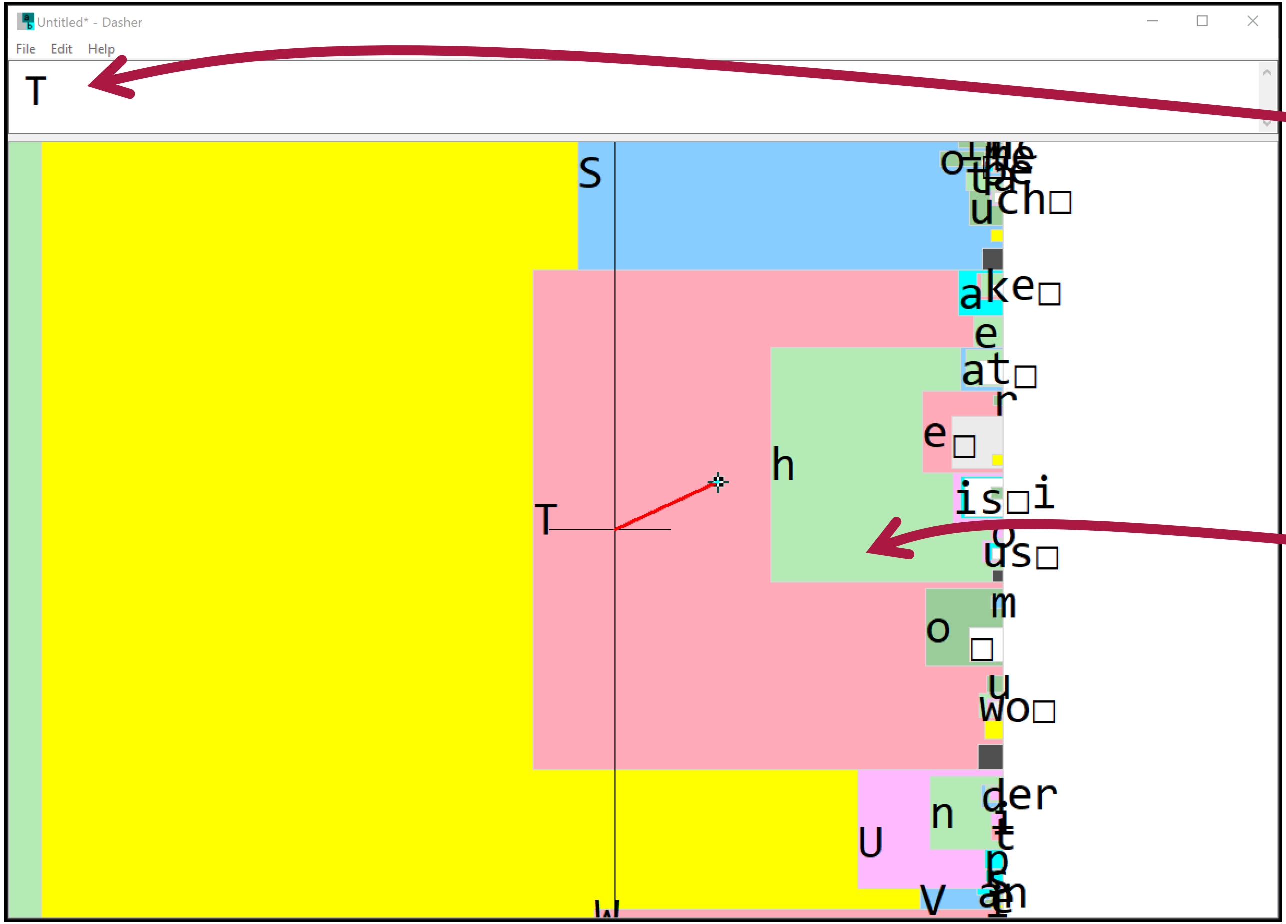


Text I write appears here.

Letters in alphabetical order: A (top) to Z (bottom). Some letters aren't currently visible, but they're still there!

Crosshairs is where I'm pointing.

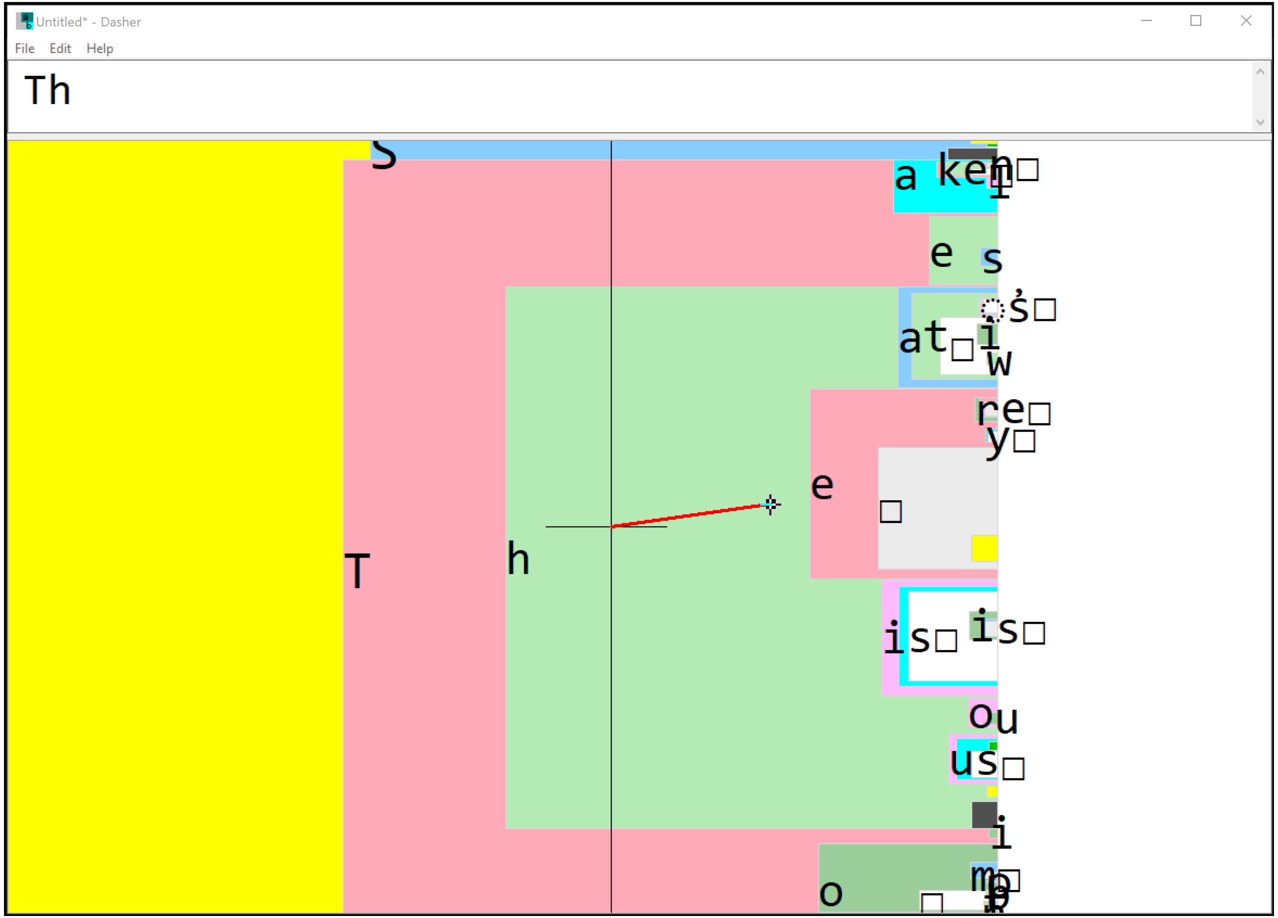
Writing "The" with Dasher, step 2



Once I navigate into the "T" box, letter is written and added to the top area.

Note how boxes of letters that often occur after "T" are bigger like "h".

Writing "The" with Dasher, step 3

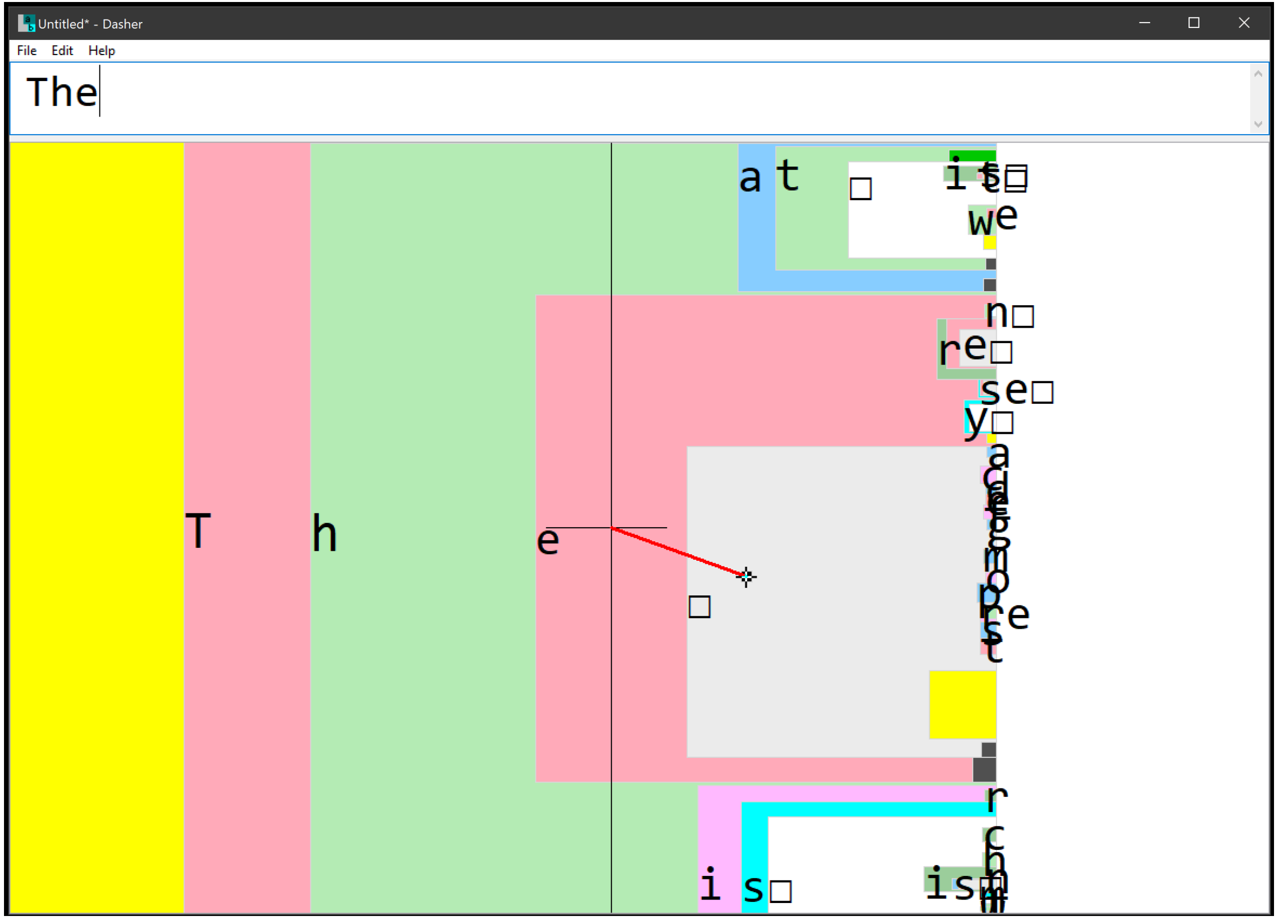


Entering "h" box writes that letter.

Dasher now strongly predicts letters that often occur after "Th".

Entire words like "There" and phrases like "This is" are visible.

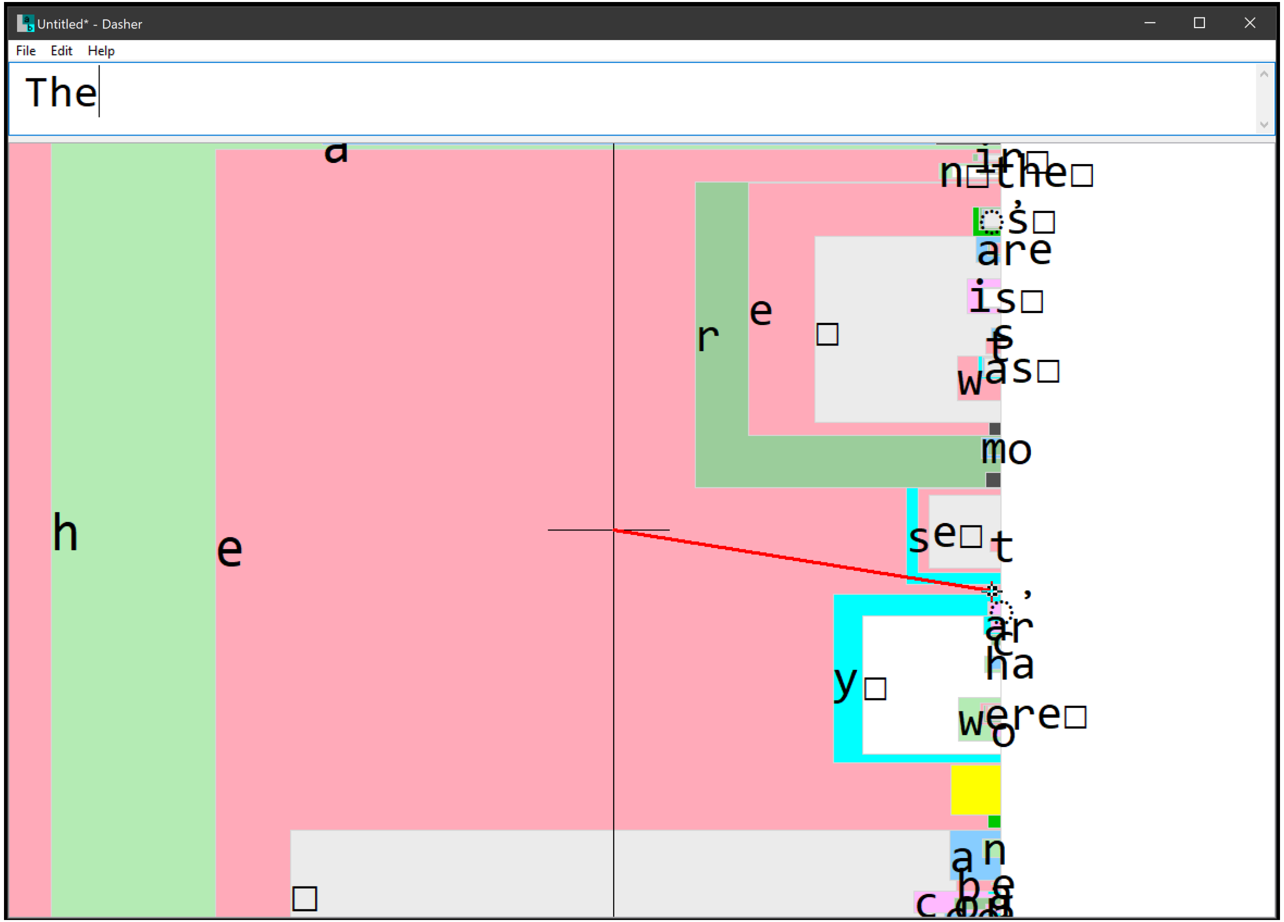
Writing "The" with Dasher, step 4



I have written "The".

Navigating to □ symbol writes a space.

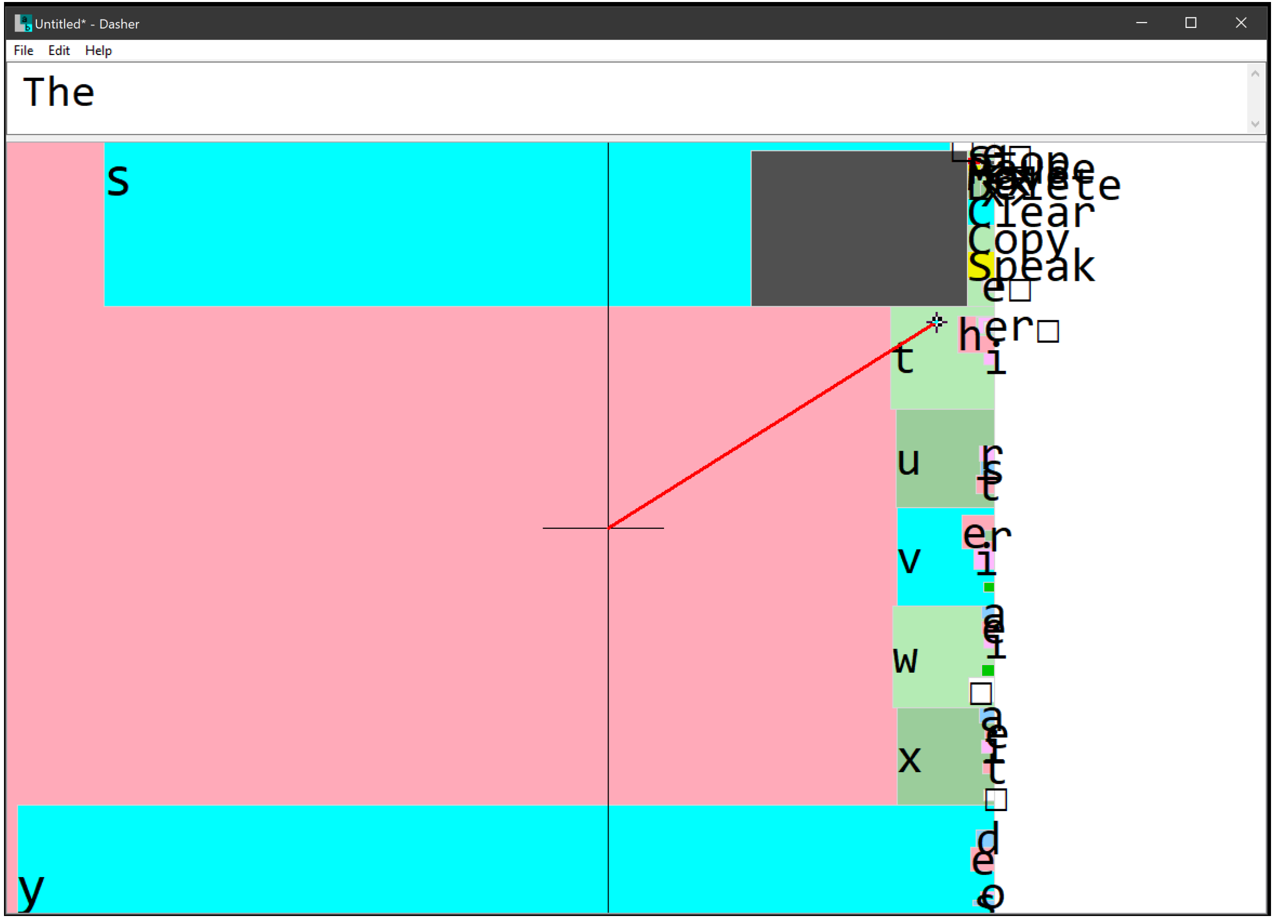
You can still write "Theta" if you really want...



Letter "t" is still there.

Just navigate towards where it appears in the alphabet.

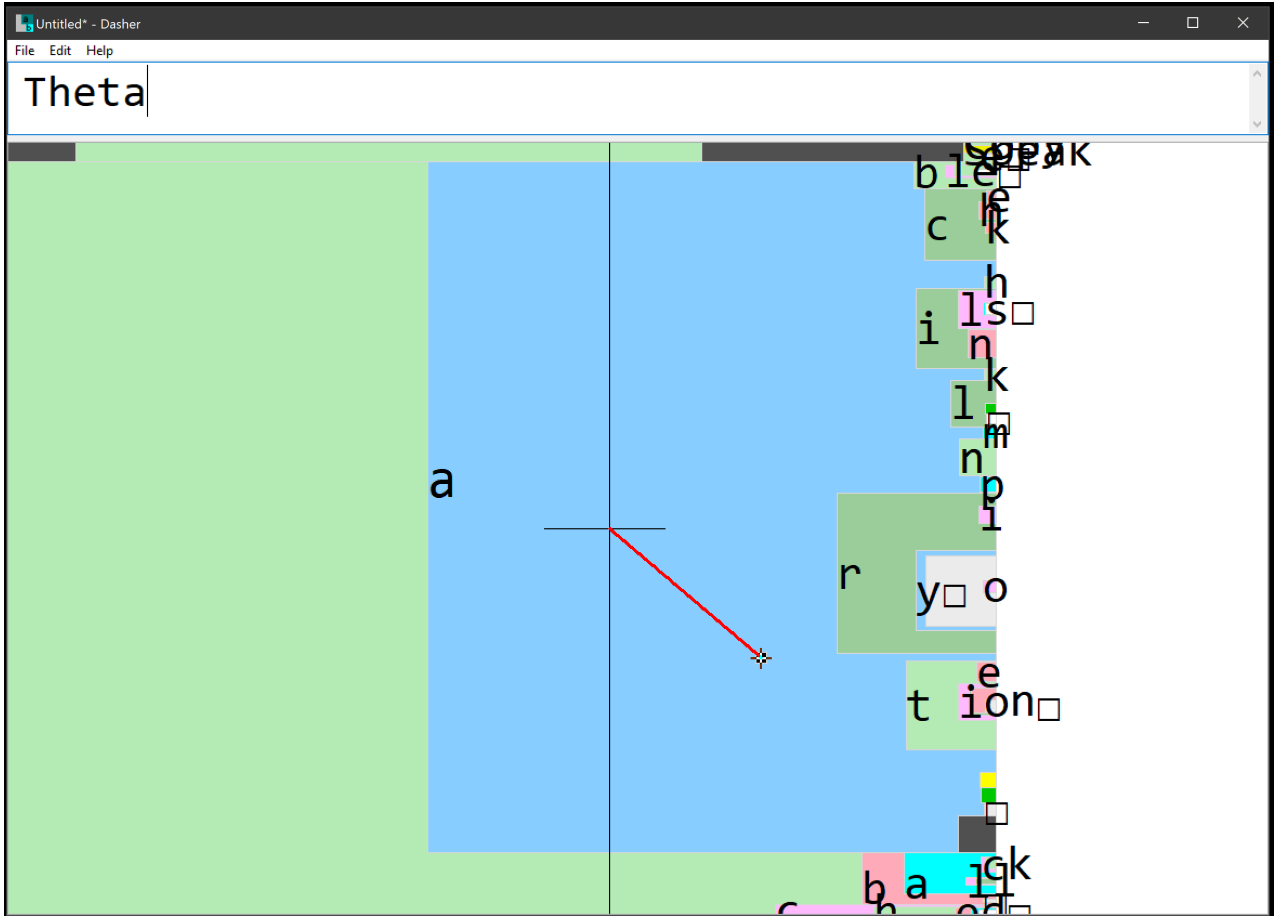
You can still write "Theta" if you really want...



Letters "t" through "x" about the same size.

Language model's training data lacks examples of words like "Theta" or "Theurgist".

You can still write "Theta" if you really want...



We made it!

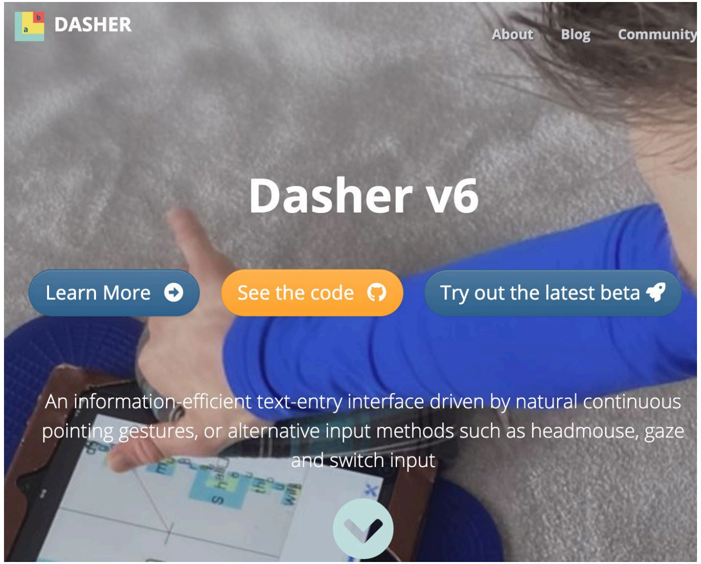
<u>Lesson</u>:
How long it takes to write something in Dasher depends on how well it is predicted by the language model.

Dasher v5 demo

https://github.com/dasher-project/dasher/releases

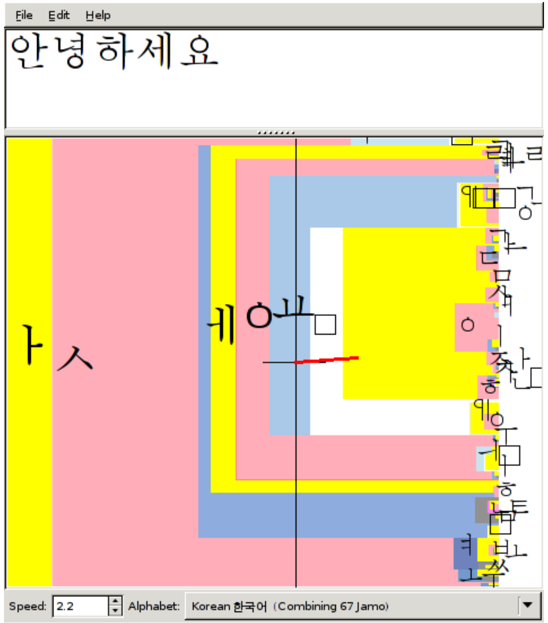
Dasher: Some details

- **Free and open source**
  - Main website: http://www.inference.org.uk/dasher/
  - Latest version (v5): https://github.com/dasher-project/dasher/releases
  - Google group: dasher-users
- **Re-implentation project (Dasher v6):**
  - https://dasher-site.netlify.app/
  - Complete rewrite using modern technologies
  - Improved user interface, documentation, ...
  - Separate and flexible language modeling layer
    - https://github.com/google-research/mozolm



Dasher: Some details

- **Adaptive language model**
  - Prediction by Partial Match (PPM)
  - Default: conditions on previous 5 characters
  - Initial training data: 300K characters of text from various sources
  - Adapts as you write
- **Supports many languages**
  - ~50 European, Asian, African, and Semitic languages
- **Supports many access methods**
  - Mouse, eye tracker, header tracker, touchscreen, 1 or more buttons, ...

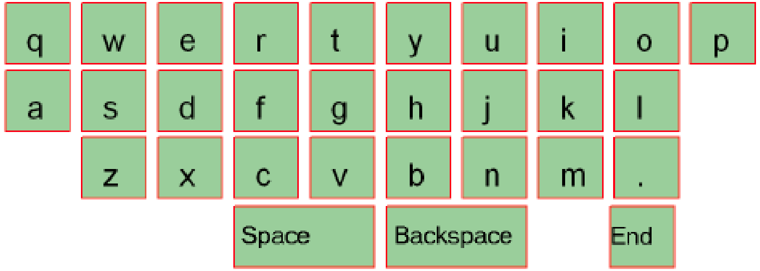


Writing in Korean in Dasher.

Dasher: User study

- **Dasher vs. Eye-typing**
  - 12 users, without motor impairment
  - Users adjustable dwell time
    - Started at 1000ms
    - By end of study, participants averaged 800ms
  - Nine 15-minute sessions with each interface
  - Wrote short memorable sentences

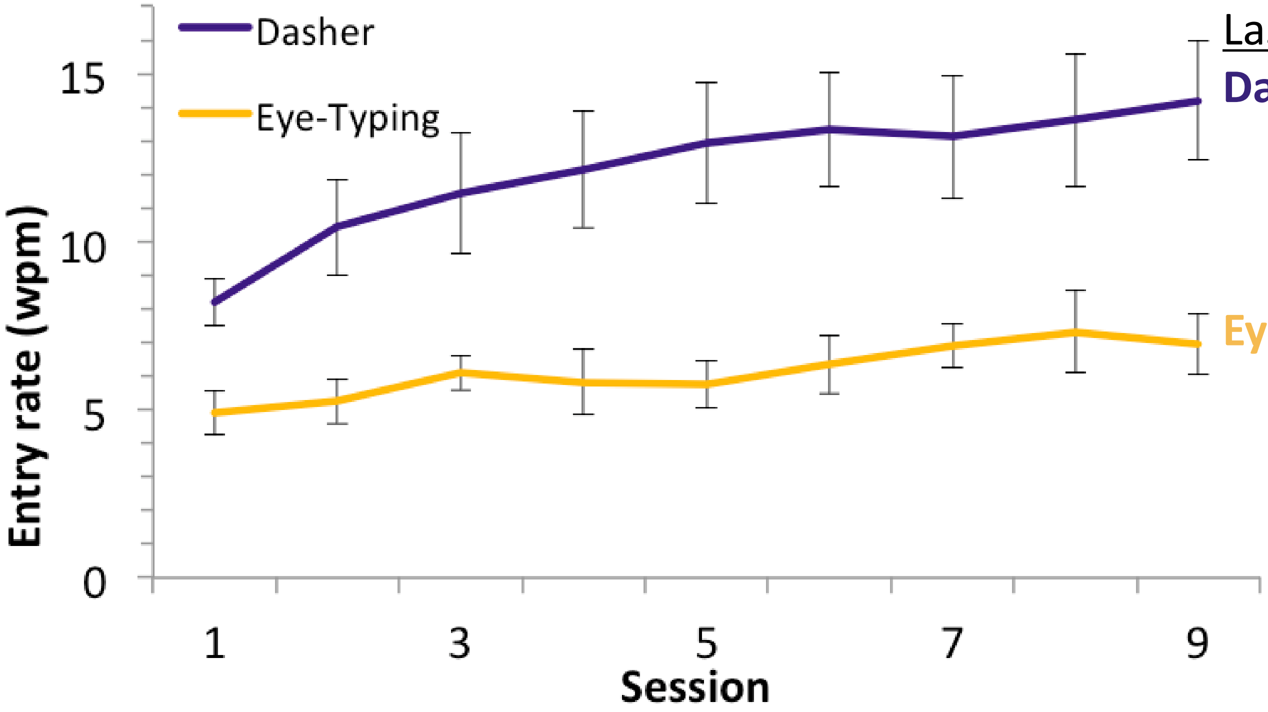
Participant using the Dasher interface controlled by a Tobii P10 eye-tracker.



QWERTY keyboard used in the dwell eye-typing interface.

Rough, et al. An Evaluation of Dasher with a High-Performance Language Model as a Gaze Communication Method. In *Proc. AVI 2014*.

Dasher: User study results



Last session:

Dasher, 14 words-per-minute

Eye-typing, 7 words-per-minute

Participants' entry rate using Dasher and Eye-typing over 9 sessions.
Error bars are 95% confidence intervals.

Dasher: Language model experiment

- **How important is an adaptive language model?**
  - Study interface: used a **static 10-gram language**
    - Trained on 3.1B characters from Twitter
    - Model pruned to a compressed size of 39MB
  - Compare against **Dasher's PPM model** (order 5, like a 6-gram model)
    - 300K training character
    - **With and without adaptation**
- **Metric: *perplexity***
  - How many likely choices for the next character / word?
    - E.g. random sequence of numbers has a per-character perplexity of 10
  - Lower perplexity is better
  - Measured on 1,347 sentences written by Enron employees on Blackberries

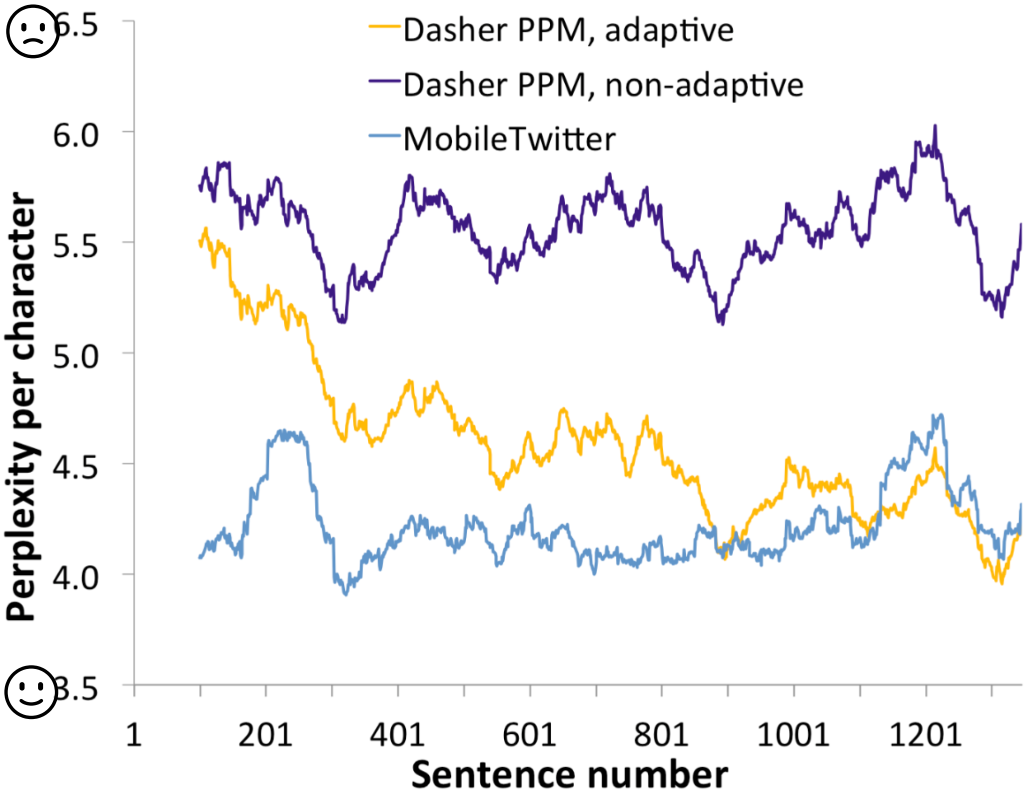
Dasher: Language model experiment results

<u>**Average perplexity**</u>

PPM, non-adaptive 5.6

PPM, adaptive 4.6

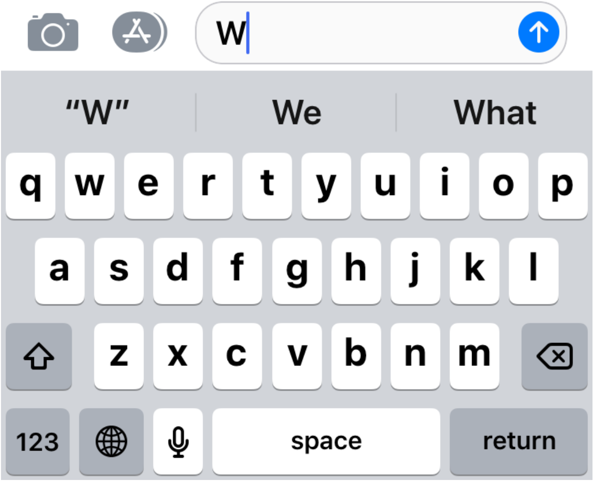
Static Twitter LM 4.2



(Lines represent 100 sentence moving average)

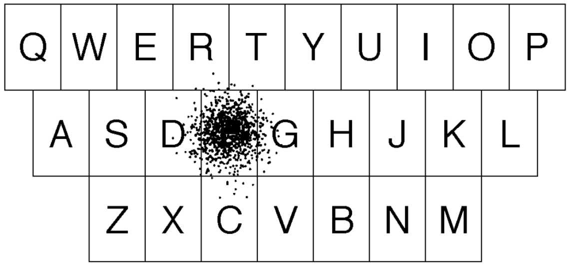
Learning from a user's writing

- **Language model adaptation / personalization**
  - Could learn words they commonly use (perhaps ones unique to the user)
  - Could learn a user's common n-grams
  - Hopefully makes auto-correct or word-predictions better
- **Metric: *keystroke savings***
  - Potential keystrokes saved by a **hypothetical user**
  - Utilizing some number of **word predictions slots**
  - Higher keystroke savings is better
  - Example:
    - Typing "What" plus space = 5 keystrokes
    - Typing "W" and selection prediction = 2 keystrokes
    - Keystroke savings = 60%



Impact of language model adaption

- **Computational experiment with unigram cache model**
  - Used sent email of 45 Enron employees, e.g.:
    - "I have to tell you, you were a hit at the party the other night. People are talking. How are you today?"
  - Primed on a user's first 30 days of emails, tested on subsequent days
  - Simulated noisy typing on a touchscreen keyboard



Model	Word error rate	Keystroke savings
No language model	38.4%	-
Background LM only	5.7%	42.1%
Uniform cache adaptation	4.6%	45.7%
Decaying cache adaptation	4.6%	45.8%

Fowler, A. et al. Effects of Language Modeling and its Personalization on Touchscreen Typing Performance. In *Proc. CHI 2015*.

Language model adaption: Open questions

- **What is actual impact on users?**
  - Perplexity and keystroke savings are ***intrinsic*** evaluations
    - Allows comparison of many possible models or algorithms
    - But do gains actually result in improved user performance?
  - We need ***extrinsic*** evaluations
    - In an actual interface, does it improve entry rate, error rate, effort, satisfaction?
    - But this is a hard study to run: longitudinal evaluation in a real-world setting
- **Performance on actual AAC user text**
  - May exhibit different characteristics than Enron employees
- **Impact of more advanced algorithms**
  - Unigram cache is a pretty simple model
  - Context-awareness: emailing is different than talking

Problems with n-gram language models

- **Fixed context length:** Only sees previous n-1 words
  - Longer context increases training cost and model size
  - Last n-1 words can't really model syntax and semantics
  - Possible fixes: Distributed training, model pruning, more complex models

```
I agree with everything in this except the dog park part.
You shouldn't take your puppy to a park until it is fully vaccinated.
Puppies especially are susceptible to Parvo.
```

Three sentence passage extracted from Reddit.

- **Sparsity:** Despite lots of data, we haven't seen everything!
  - Words treated independently: `democracy ≠ dog ≠ puppy ≠ puppies`
  - Rare or new words in a language?
  - Possible fixes: Smoothing methods, unknown word, subword vocabulary

Enter the neural networks...

METRO

NEWS...

Elon Musk-founded OpenAI builds artificial intelligence so powerful it must be kept locked up for the good of humanity

Comment

Jasper Hamill
Friday 15 Feb 2019 10:06 am

Scientists at an organisation founded and sponsored by Elon Musk have announced the creation of a terrifying artificial intelligence that's so smart they refused to release it to the public.

OpenAI's GPT-2 is designed to write just like a human and is an impressive leap forward capable of penning chillingly convincing text.

It was 'trained' by analysing eight million web pages and is capable of writing large tracts based upon a 'prompt' written by a real person.

But the machine mind will not be released in its fully-fledged form because of the risk of it being used for 'malicious purposes' such as generating fake news, impersonating people online, automating the production of spam or churning out 'abusive or faked content to post on social media'.

OpenAI wrote: 'Due to our concerns about malicious applications of the technology, we are not releasing the trained model.

Elon is one of the world's most famous doom-mongers and fears the rise of the machines will end very badly for humans Pictures: REX/Joe Rogan Experience

https://metro.co.uk/2019/02/15/elon-musks-openai-builds-artificial-intelligence-powerful-must-kept-locked-good-humanity-8634379/

AI Business

Google Brain unveils trillion-parameter AI language model, the largest yet

by Louis Stone 1/15/2021

Currently more of a research project than a commercial product

Google Brain has developed an artificial intelligence language model with some 1.6 trillion parameters.

That puts it at nine times the size of the OpenAI's 175 billion parameter GPT-3, previously considered to be the world's largest language model.

While it gives some indication of the project's scale, the models are too different in architecture for a meaningful 'apples-to-apples' comparison.

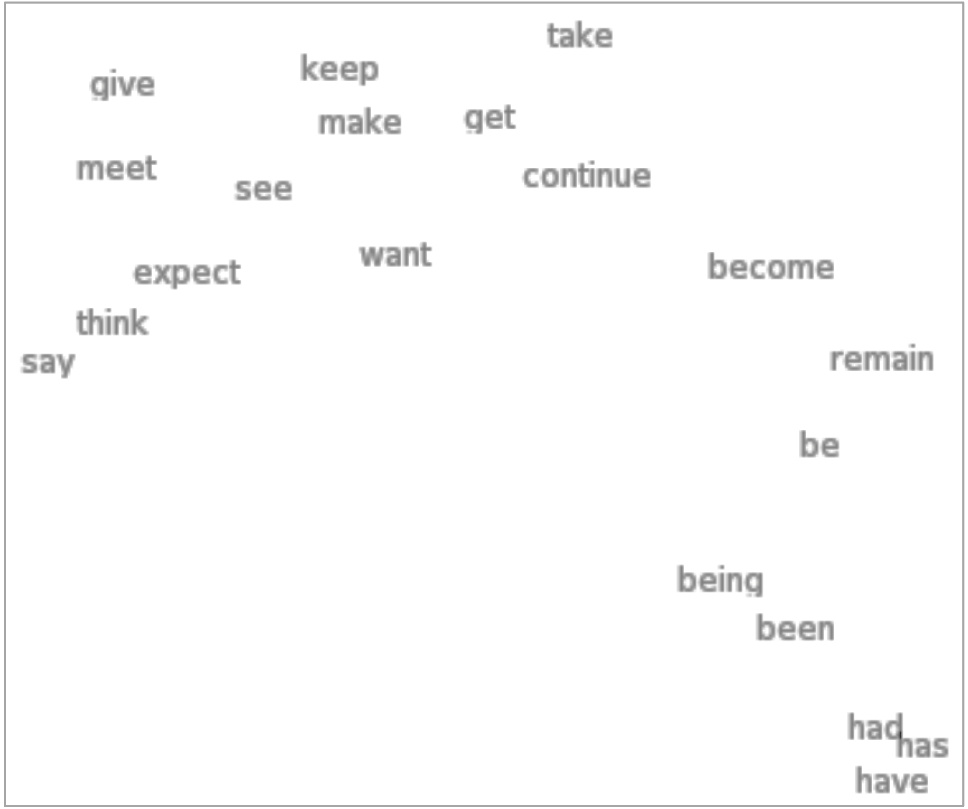
Model	Params (B)
OpenAI GPT	0.1
OpenAI GPT-2	1.5
OpenAI GPT-3	175
Google switch transformer	1,600

I like big parameters and I cannot lie

Neural network language models: Advantages

- **No fixed context length**
  - Better at long range dependencies
- **Sparsity less an issue**
  - Words are a continuous vector

$$\textit{expect} = \begin{pmatrix} 0.286 \\ 0.792 \\ -0.177 \\ -0.107 \\ 0.109 \\ -0.542 \\ 0.349 \\ 0.271 \\ 0.487 \end{pmatrix}$$



http://web.stanford.edu/class/cs224n/

Neural network language models: Advantages

- **More compact than n-gram models**
  - Size doesn't increase every time it sees a novel bit of text
- **Great performance!**
  - Typically better than n-gram model on the same data
  - Using together with n-gram model = better than either alone
  - Using an ***ensemble*** of neural models improves performance

Problems with any language model

- **Sensitivity to training data:** Predicts text similar to genre trained on
  - Lots of ***corpora*** (collection of text) of written English

Reddit	I wish it could have let us know in advance. No one can regulate your emotions for you.
Apache email	There's a backlog of thousands of defects, just keep moving forward. Could you describe how I can reproduce the problem?
Blog posts	In The Rocker Dwight Schrute trades his shirt and tie for a pair of drumsticks. What is the impact of that going to be on your business?

  - Conversational corpora much harder to find

Switchboard	do you cook for yourself or do you cook for others mostly I just cook for myself
Movie subtitles	I don't understand why he's so upset, sensei. Got anything cold to drink?

  - Possible fixes: Data select from corpora, adapt model on user's text

Modeling conversations

- **Speaking to someone ≠ writing to someone**
  - Augmentative and Alternative Communication (AAC)
  - AAC users may use their device to speak for them
  - How can we optimize for conversational communication?
- **Intelligent training data selection**
  - Billions of words of text available (for English anyway)
    - We can filter out most of it and still have plenty!
  - Train ***in-domain*** language model on a small amount of conversational text
  - Train ***out-of-domain*** language model on the large training set
  - Score sentences with both, keep those scoring above a threshold[1]

[1] Moore, R. and Lewis, W. Intelligent Selection of Language Model Training Data. In *Proc. ACL 2010*.

Selecting AAC-like training data

- **Problem 1: Where to get AAC-like conversational text?**
  - No large scale ***in-domain*** training sets
  - Actual AAC user text hard to get due to privacy and ethical reasons
  - A few small AAC-like phrase sets:
    - **COMM** - Phrases written in response to hypothetical communication situations[1]
    - **SPECIALISTS** - Phrases suggested by AAC specialists at University of Nebraska-Lincoln

COMM	SPECIALISTS
`Yes, I am studying hard.`	`Guess what I want?`
`Is the staff treating you well?`	`It is different alright`
`This is a really good sale.`	`When you find out let me know`
`Can I have a large orange drink with no ice?`	`A little better`

[1] Venkatagiri, H. Efficient Keyboard Layouts for Sequential Access in Augmentative and Alternative Communication. In *Augmentative and Alternative Communication (1999)*.

Crowdsource AAC-like data?!?

- **Microtask on Amazon Mechanical Turk**
  - Workers invented messages
  - Judged other messages
  - After quality control:
    - 5,890 messages
    - From 289 unique workers
  - http://www.aactext.org/imagine

Due to a medical condition or accident, **imagine you can't talk or type** on a normal keyboard. Instead, you use a special **communication device that speaks for you**. You operate this device by pushing a button whenever your desired letter is highlighted. By repeatedly pushing the button, you can spell out words, phrases or entire sentences.

Invent a fictitious (but plausible) communication you might make using your device. Think of the things you might want to say to your family, friends, care-givers, and people you meet in the community. Please proceed **quickly and accurately**. Do NOT include any private information (such as real email addresses, phone numbers, or names). **Invent a new communication** for each task of this type.

Write as if you were actually using your communication device to speak for you. Do NOT write about your actions or state of mind.

Vertanen, K. and Kristensson, P.O. The Imagination of Crowds: Conversational AAC Language Modeling using Crowdsourcing and Large Data Sources. In *Proc. EMNLP 2011*.

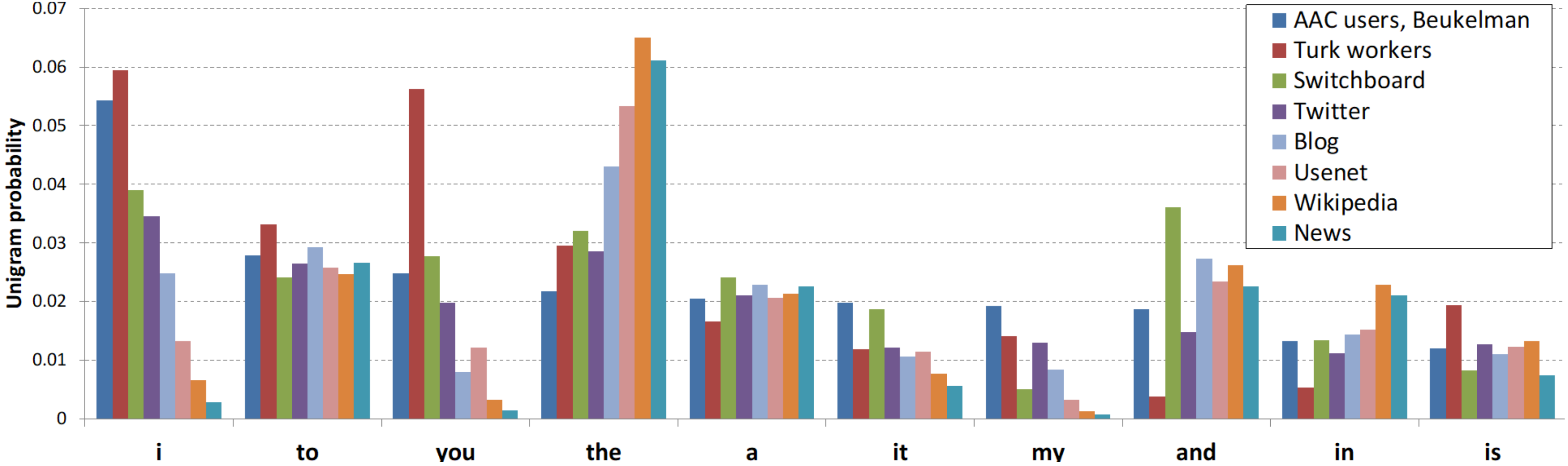
Random sample of workers' messages

```
Good to see you again.
I am very well thank you.
I don't know.
Glad to see you Fred.
How is Betty?
Well you always told me to watch what I was saying now I can haha!
What did the lawyer say?
Am very thankful.
How are you now?
Can you get me a drink?
I'll try, maybe next week.
I wish you wouldn't be mean to me!
It's hot outside.
Please change the channel.
I have his number.
Remember to lock the door when you leave.
```

Selecting AAC-like training data

- **Problem 2: What is the best choice for a large training set?**
  - Many large ***out-of-domain*** training sets to choose from:
    - NEWS – Newspaper articles from several newswire corpora
    - WIKIPEDIA - Articles and discussion threads from Wikipedia
    - USENET – Messages from a Usenet corpus
    - SWITCHBOARD – Transcripts of 2,217 telephone conversations
    - BLOG – Blog posts from the ICWSM corpus
    - TWITTER – Tweets sampled between 12/2010 – 3/2011

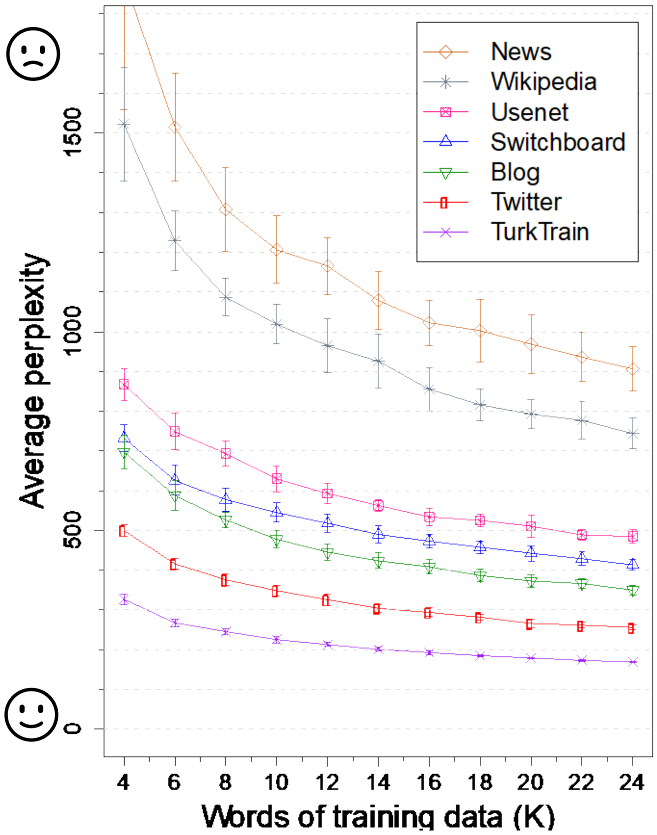
A (rough) comparison against actual AAC data



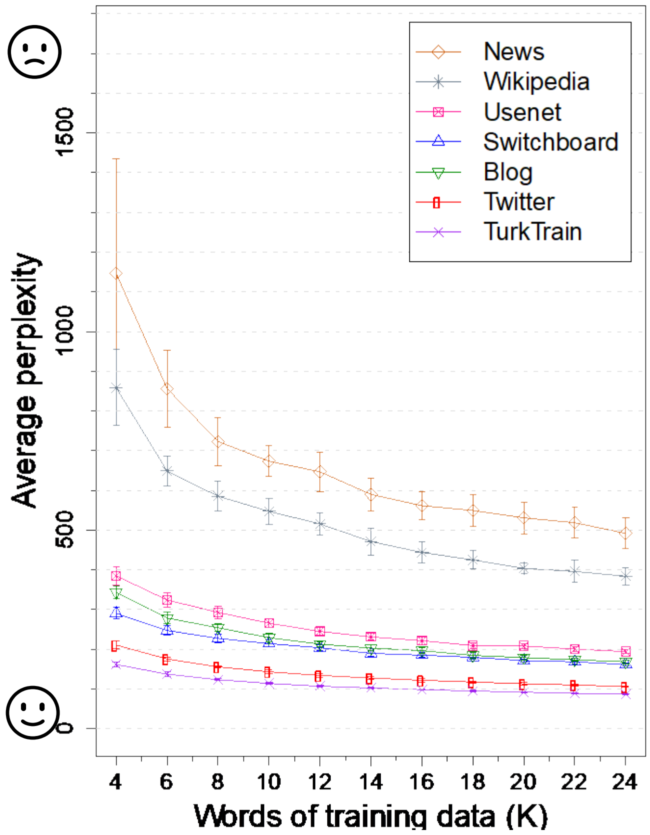
The unigram probabilities of the top 10 words of five nonspeaking adults over 14 days[1] *compared with the probability in our Amazon Turk data and other large training sources.*

[1] Beukelman, D. et al. Frequency of word occurrence in communication samples produced by adult communication aid users. *Journal of Speech and Hearing Disorders, 1984.*

Perplexity training on different sources



COMM test set



SPECIALISTS test set

Result:
Turk data was the best, but Twitter was a close second.

However, crowdsourcing millions of words of text could get expensive!

Twitter data selection example

CE difference	Sentence
-2.0	all's well now almost too late any news you want to share about your company
-1.0	aah lunch time about to leave the party actions speak louder then words my friend
0.0	aaron carter better be much alive aawh he is so cute ability is what you are capable of doing
1.0	a bad bounce leads to another kingston goal a bad gurl is lyk smoke in d eye or vineger on d teeth abba the concert tonight
2.0	a b c d e f g h i j k l m n o p q r s t v w x y z oops abidan actually passed the cinnamon challenge xd a bit of a touch up in ps

more AAC like

less AAC like

Results of using data selection

- **Training a AAC conversational language model:**
  - In-domain-text: Amazon Turk worker invented messages
  - Out-of-domain text: TWITTER, BLOG, and USENET
  - Trained 3 languages models, then mixed together

Language model	Perplexity COMM	Perplexity SPECIALISTS	Keystroke savings COMM	Keystroke savings SPECIALISTS
Switchboard (no data selection)	166	64	54.4%	57.7%
Twitter (no data selection)	56	27	60.9%	61.9%
Mixture (with data selection)	48	26	62.5%	63.1%

  - Latest trained models: https://imagineville.org/software/
  - Open question: How well does this work on text from actual AAC users?

Recruiting: AAC text donation effort

You can help improve future AAC interfaces by donating sentences from your history

We are collecting sentences written by users of Augmentative and Alternative Communication (AAC) interfaces. These sentences will be used to help improve AAC text entry interfaces. Contributed sentences will become part of a **public data set**. This data set will be available for download by researchers working to improve AAC text entry interfaces.

Participant requirements:

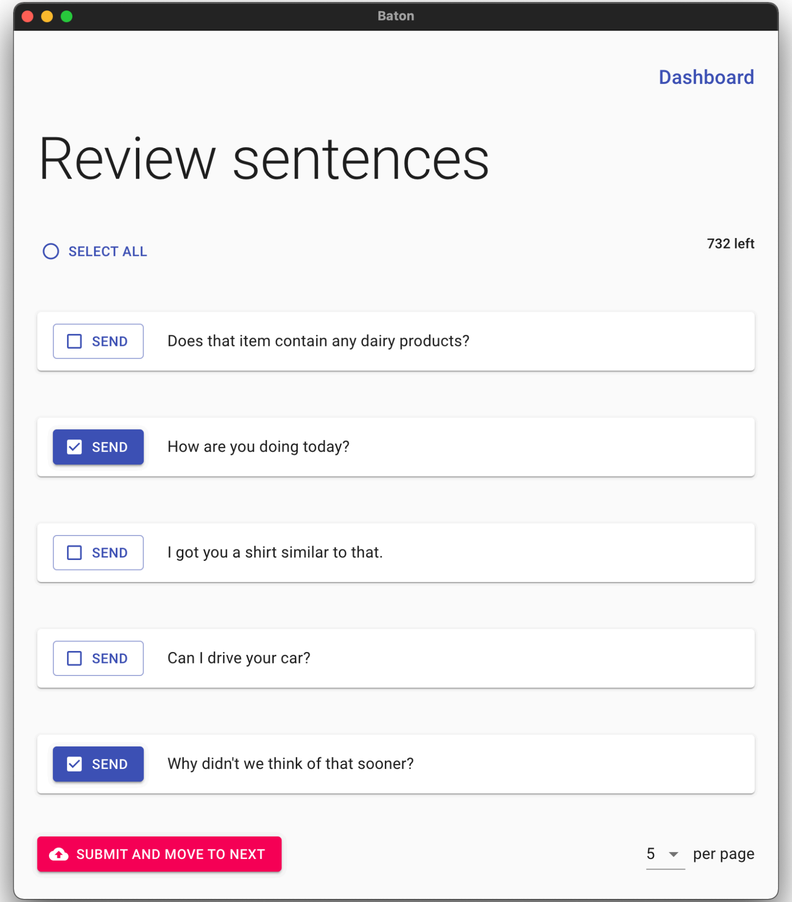
- You must be 18 years of age or older
- You **must use an AAC interface** for at least some of your communication needs
- You must use an AAC interface that logs your writing to a simple text file
- You must not have a cognitive impairment

Details:

You will need to install our custom application on your computer. The application runs on Windows, Mac, and Linux operating systems. You will point the application at a text file of things you have written using your AAC interface. You can then **select whatever sentences you would like to share** with the project.

Contact Keith Vertanen, vertanen@mtu.edu if interested. More details: https://aactext.org/donate

Baton donation app



User must explicitly select sentences to donate
Reviewed prior to adding to public data set

Privacy options:

- Sentences put in global pool
- Associated with anonymous ID
- Associated with anonymous ID + user details

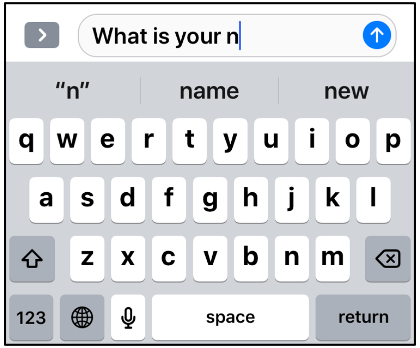
Imports from:

- Dasher
- Grid 3
- Plain text file
- Comma separated file

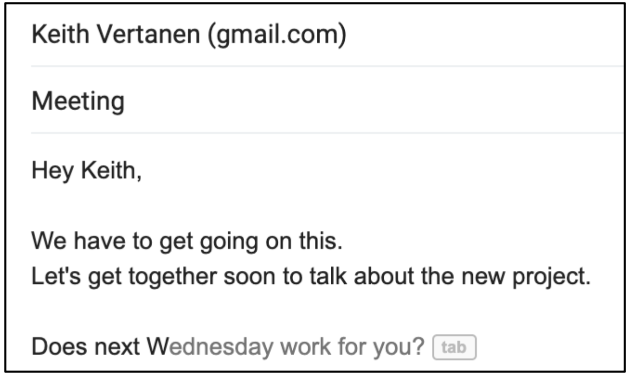
Demo of different language models

Language models and text entry

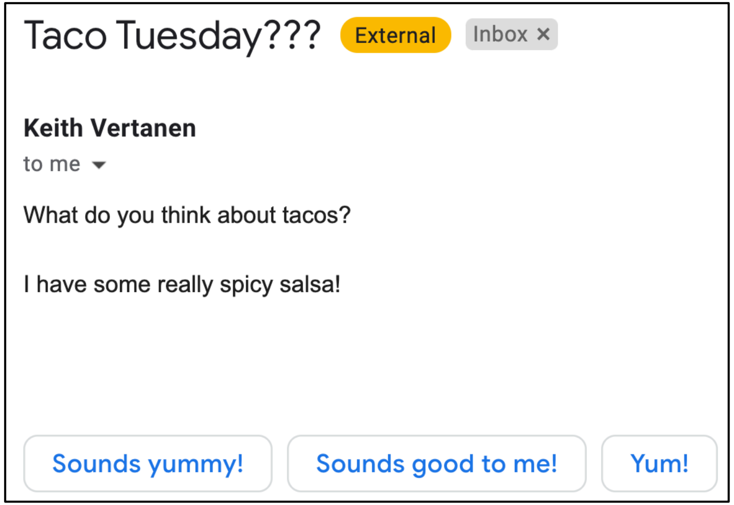
- **Uses of a language model (thus far):**
  - **Auto-correct** to handle mistargeted keys, missing keys, additional keys
  - Dasher: fundamental to the interaction technique
- **Some other uses:**



Word predictions



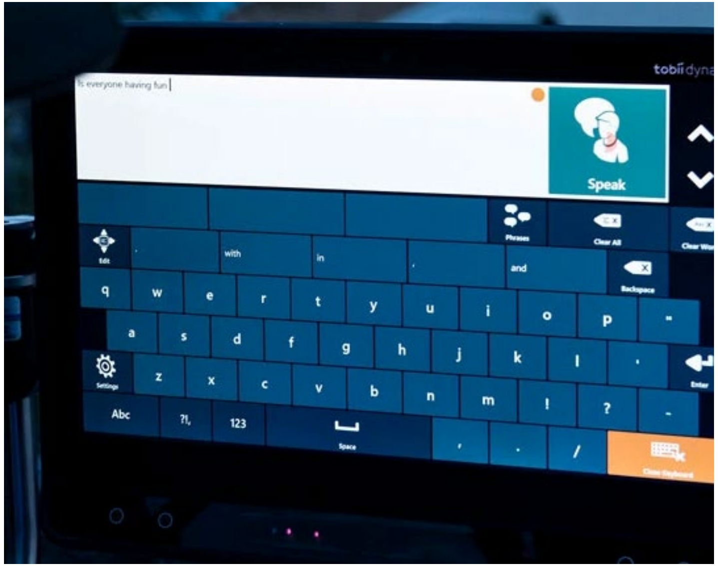
Phrase predictions



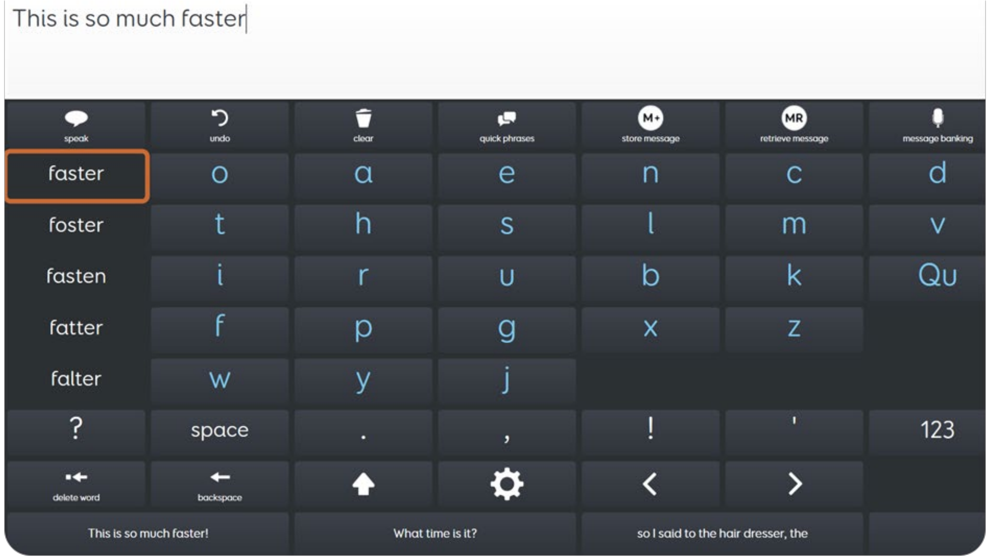
Entire reply predictions

Word predictions

- **AAC interfaces often offer word predictions**
  - Can accelerate a user with a slow input rate
  - Can help to avoid recognition errors



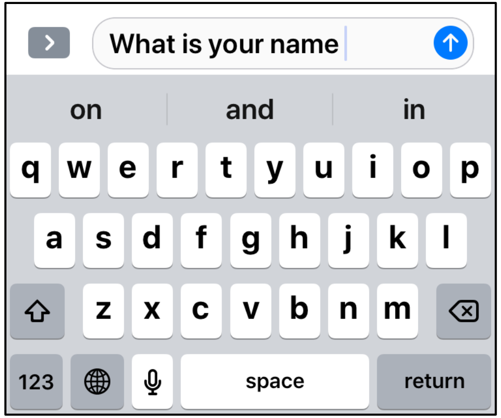
Tobii Communicator eye keyboard. Five word predictions appear above the keyboard



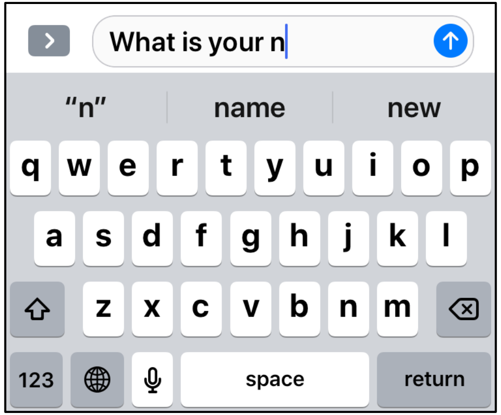
Text Talker by Smartbox. Five word predictions appear in the leftmost column.

How do word predictions work? The basics

- **If you haven't started the current word:**
  – Ask language model for X most probable given any previous text



- **If you typed part of word:**
  – Lookup all words in some ***vocabulary*** matching letters typed thus far
  – Ask language model for X most probable given any previous text



Word predictions: Design decision 1

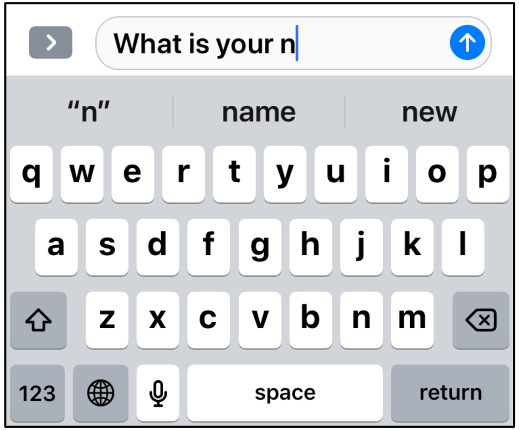
- **How many predictions X to offer?**
  - More is better from a keystroke savings perspective, but...
  - Increases cognitive overhead to check them
  - Takes up more screen space
    - Smaller targets on a virtual keyboard
  - May increase time to select targets
    - More things in a row-column scanner

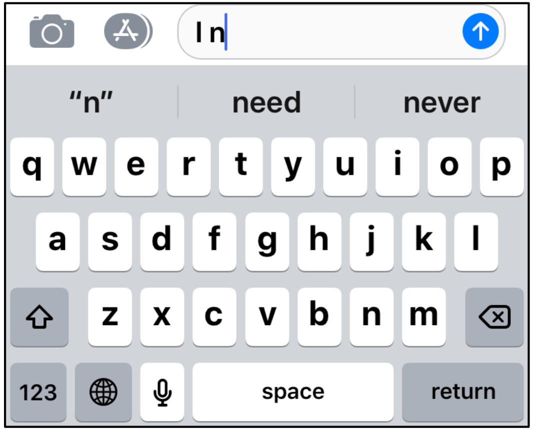
Word predictions	Keystroke savings
1	46.2
2	53.1
3	56.6
4	59.1
5	60.5
6	61.8
7	62.7
8	63.7
9	64.3
10	64.5

Keystroke savings (%) with increasing number of word predictions. Results on 4,567 AAC-like sentences using a Twitter 4-gram optimized for AAC-like text.

Word predictions: Design decision 2

- **Condition on previous words?**
  - Word **unigram** language model, ignores previous text
  - Unigram predictions are **worse**, but...
  - Can be **more predictable** for a user
    - After typing "n", "name" always appears in slot 2





Different word predictions for words starting with "n".

Model	Keystroke savings
1-gram	45.8
2-gram	56.1
3-gram	59.6
4-gram	60.5

Keystroke savings (%) with increasing language model context siz. Results on 4,567 AAC-like sentences using a Twitter 4-gram optimized for AAC-like text.

Word predictions: Design decision 3

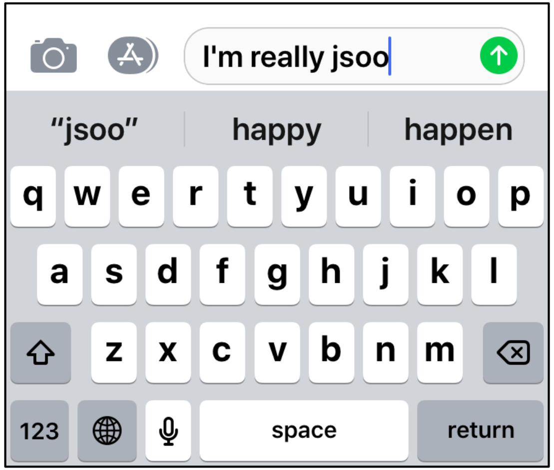
- **What vocabulary to use?**
  - Are words appropriate given a user's literacy?
  - Do you want to predict naughty words?
  - Words ideally in the language model's training data
    - Otherwise no basis for ranking the word
    - What if the user adds words to the vocabulary?

Vocab size	Keystroke savings
100k	60.5
20k	60.0
10k	58.8
5k	56.8
2k	51.7
1k	47.1

Keystroke savings (%) with decreasing vocabulary size. Results on 4,567 AAC-like sentences using a Twitter 4-gram optimized for AAC-like text and five prediction slots.

Word predictions: Design decision 4

- **Consider a user's input accurate?**
  – User may make errors typing characters



Treat as certain

– Return only predictions starting with exact prefix

– But any mistake when typing the prefix must be corrected

Treat as uncertain

– Use input model to obtain set of possible prefixes

– Find words in vocabulary that match possible prefixes

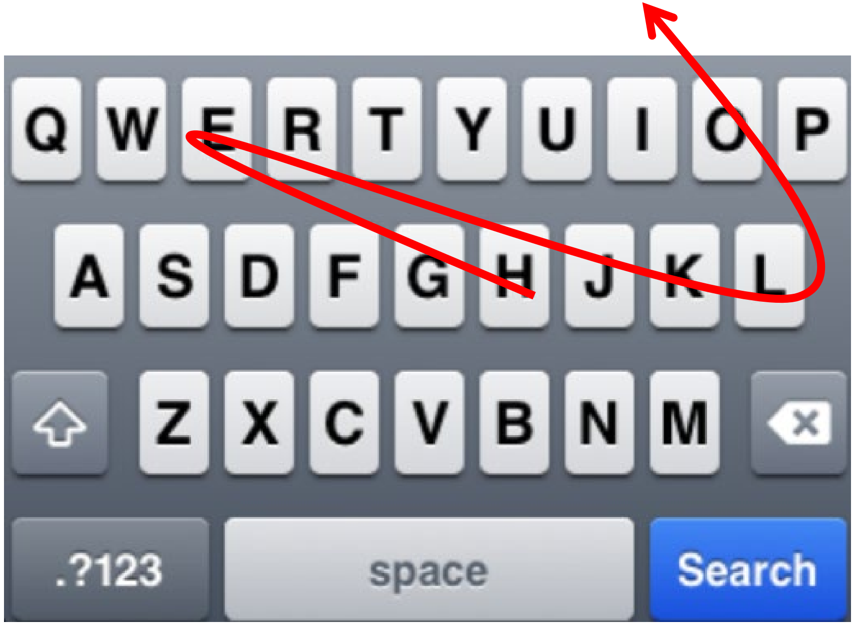
– Combine probabilities from input model and language model

Even more uses of language models

Abbreviated input[1]

Touchscreen device demo:

https://keyboard.imagineville.org



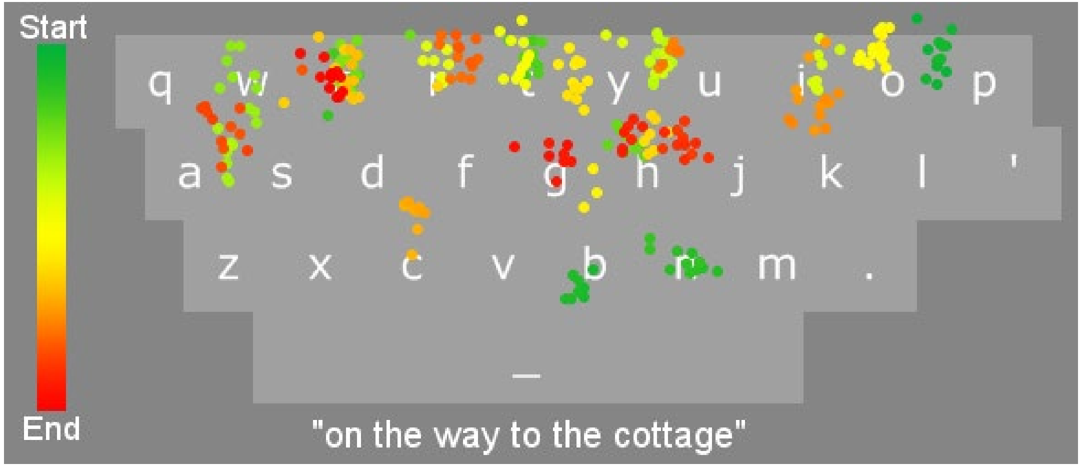
Word gesture input (aka swipe, glide)[2]

[1] Adhikary, J. and Vertanen, K. Accelerating Text Communication via Abbreviated Sentence Input. In *Proc. of ACL 2021*.

[2] Kristennson, P.O. and Zhai, S. SHARK[2]: a large vocabulary shorthand writing system for pen-based computers. In *Proc. of UIST 2004*.

Dwell-free eye typing

- **What if you didn't have to dwell to eye type?**



User gazes at each desired letter in sequence, skipping spaces.

System recognizes entire sentence.

  - Participants reached **46 words-per-minute**[1]
  - But we cheated: recognizer knew the answer!
    - Users only needed to **look within 1.5 key radii** of target letters
  - Implemented for real in Tobii Communicator 5 https://www.youtube.com/watch?v=FbPmSrsunCQ

[1] Kristennson, P.O. and Vertanen, K. The Potential of Dwell-Free Eye-Typing for Fast Assistive Gaze Communication. In *Proc. of ETRA 2012*.

Single switch communication

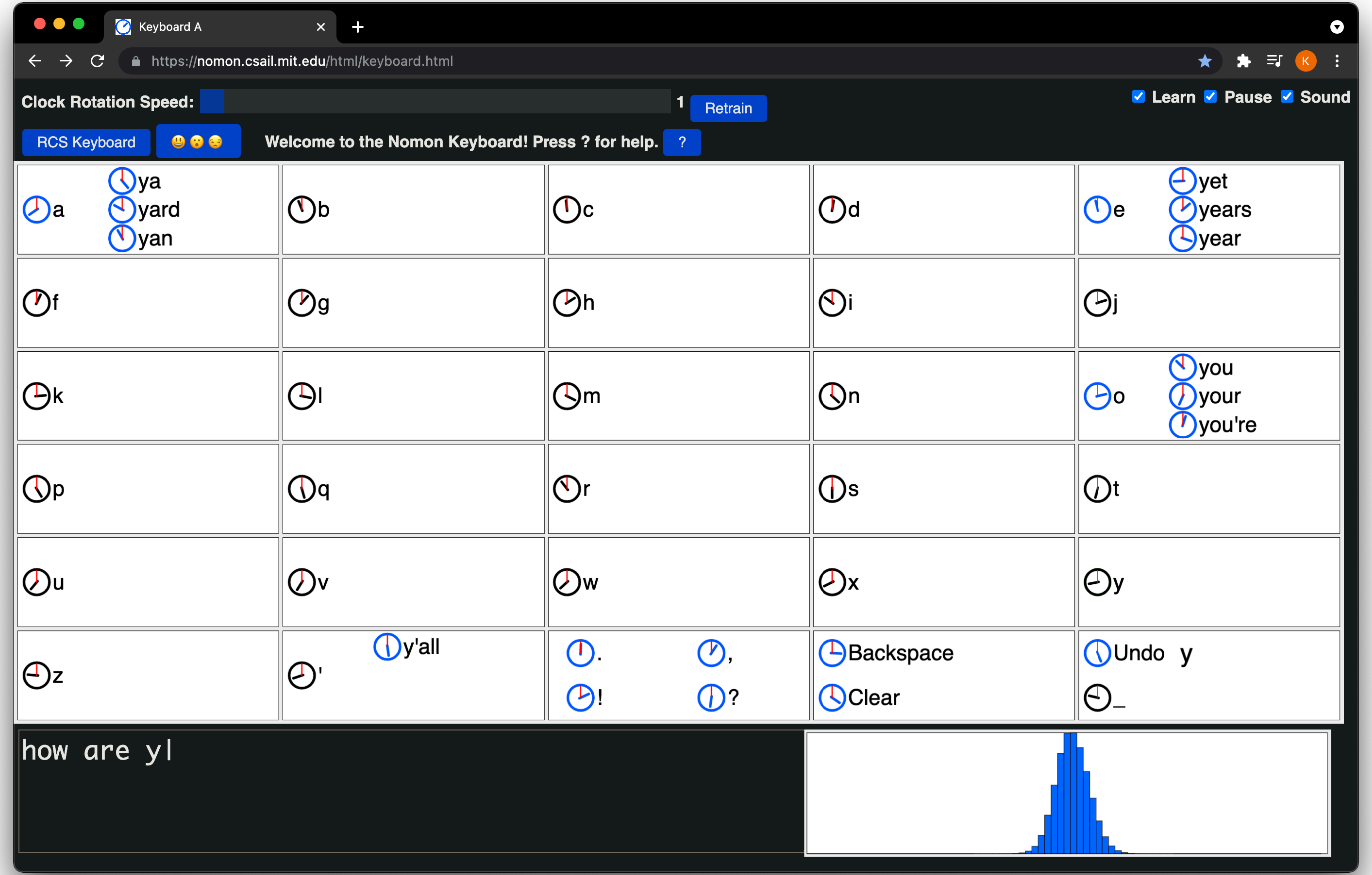
- **Row-column scanning (RCS)**

space	E	A	R	D	U
T	O	I	L	G	B
N	S	F	Y	V	J
H	C	P	K	Q	.
M	W	X	Z	,	?

https://www.wikiwand.com/en/Switch_access_scanning

- **Limitations:**
  - Options must be arranged in a grid
  - No built in error tolerance
  - Selection time does not scale well with number of options

Nomon: flexible writing with a single switch



Dr. Tamara Broderick (MIT)

Nicholas Bonaker (MIT)

Dr. Emli-Mari Nel

Dr. Keith Vertanen (MTU)

Broderick T. and MacKay D.J.C. Fast and Flexible Selection with a Single Switch. *PLOS ONE 2009.*

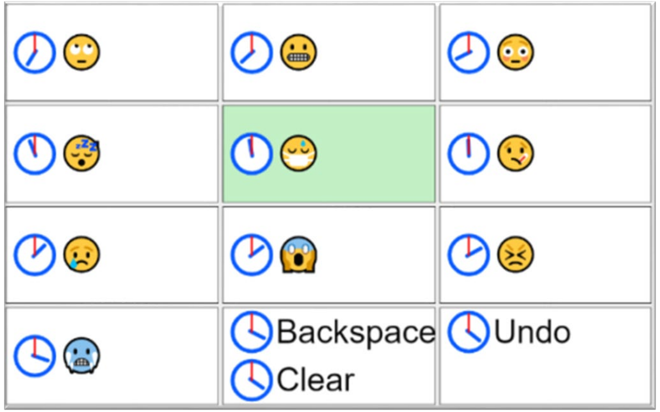
Nomon demo

https://nomon.csail.mit.edu

Use spacebar as your switch

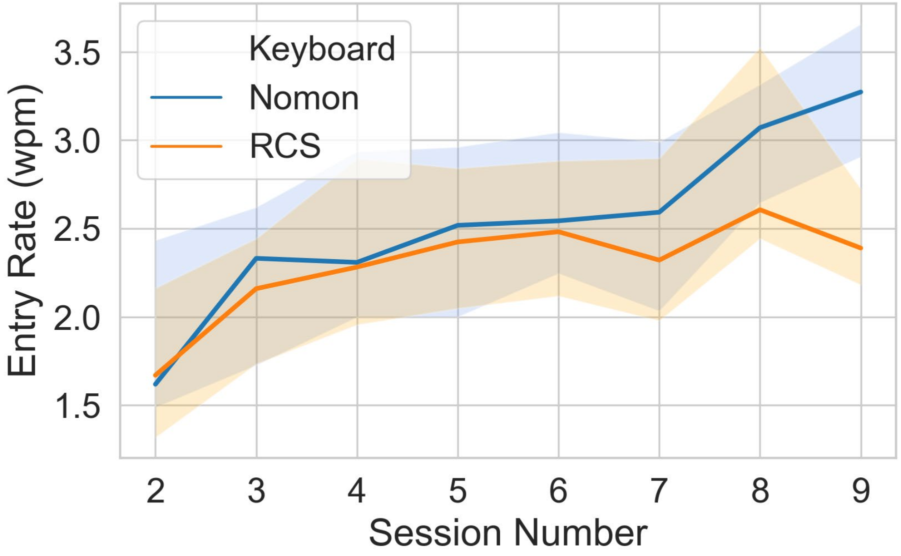
Nomon user study

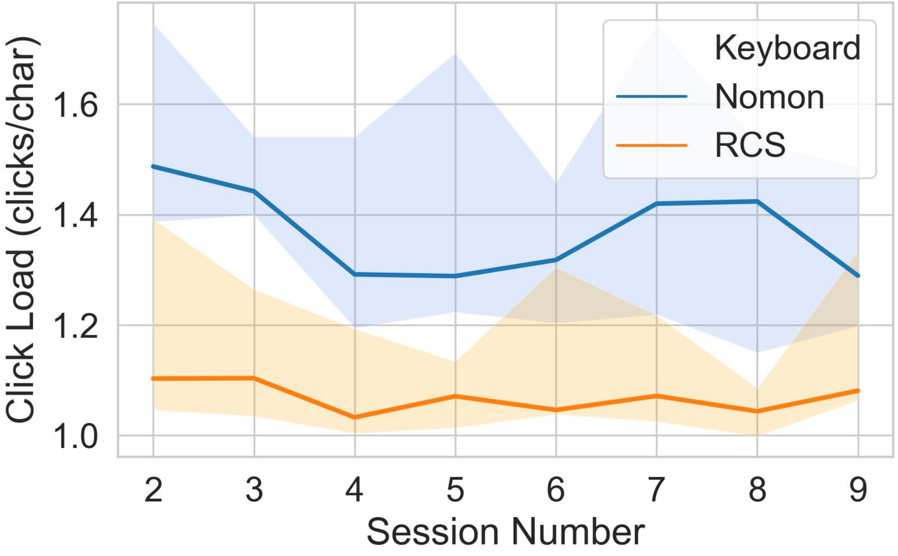
- **Nomon vs. row-column scanning (RCS)**
  - 13 participants without motor-impairment
  - 10 sessions
  - Participants used a **webcam switch**
    - Webcam performance similar to actual switch users
  - **Text entry task**
    - Entered short phrases
  - **Picture Selection**
    - Selected from a large set of emojis
  - More details:
    - https://www.youtube.com/watch?v=eIm5G1sqHAM



A portion of the picture selection grid.

Nomon study: Text entry results

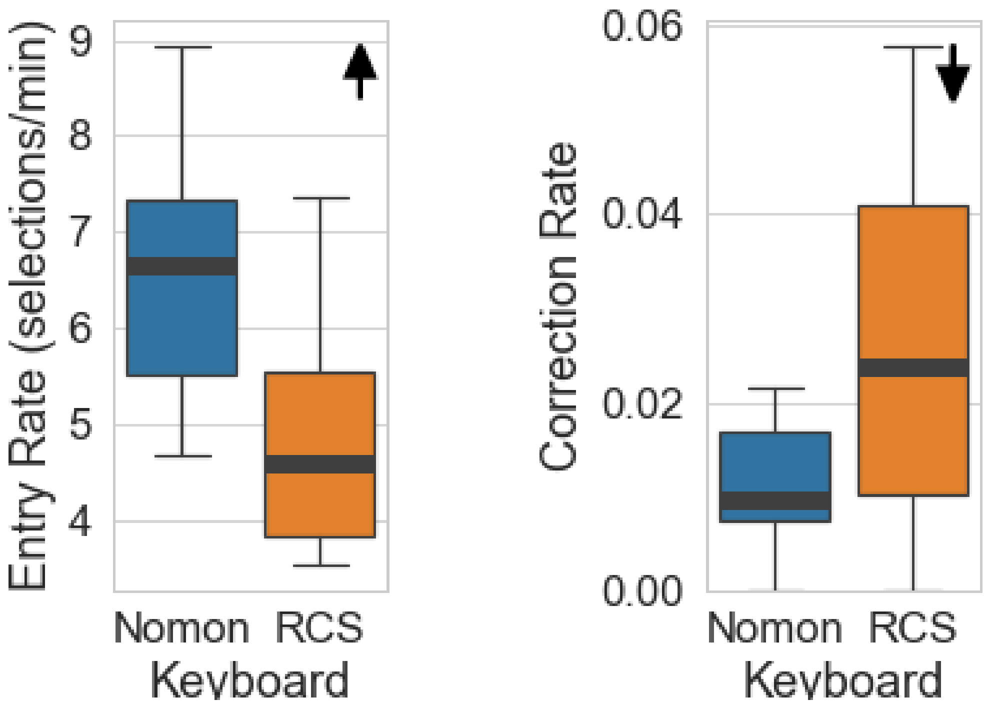




Participants plateaued after 5th session with row-column scanning (RCS)

Participants **continued to improve with Nomon**

Nomon study: Picture selection



Participants selected options **35% faster** with Nomon

Participants performed **half as many corrections** using Nomon

Paid Research Participants Needed

- Seeking individuals with motor-impairments and experience using **single-switch interfaces**
- You will be asked to use two assistive-technology keyboard interfaces
- Study will take place online and at your own pace
- Must be in the **US/UK** and **18+**.
- Compensation is **$15 USD** per session
- **Interested?** Email Nick Bonaker at **nbonaker@mit.edu**
- Study is run by Nicholas Bonaker (MIT), Prof. Tamara Broderick (MIT), Dr. Emli-Mari Nel, and Prof. Keith Vertanen(MTU)

Faster writing by abbreviating?

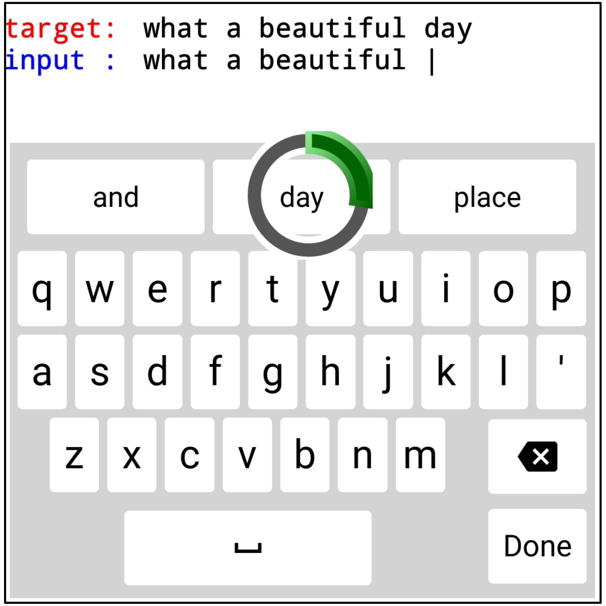
- **Abbreviated sentence input**
  - Fast input difficult for people who are motor-impaired
  - Word completions can accelerate input
    - But comes with a cognitive cost
    - Takes physical effort to select
  - Input entire sentence
    - Remove spaces and mid-word vowels
    - Recognizer infers most likely sentence

`plseclltmrrw` —Recognizer→ `please call tomorrow`

Adhikary, J. and Vertanen, K. Accelerating Text Communication via Abbreviated Sentence Input. In *Proc. of ACL 2021*.

Faster writing by abbreviating?

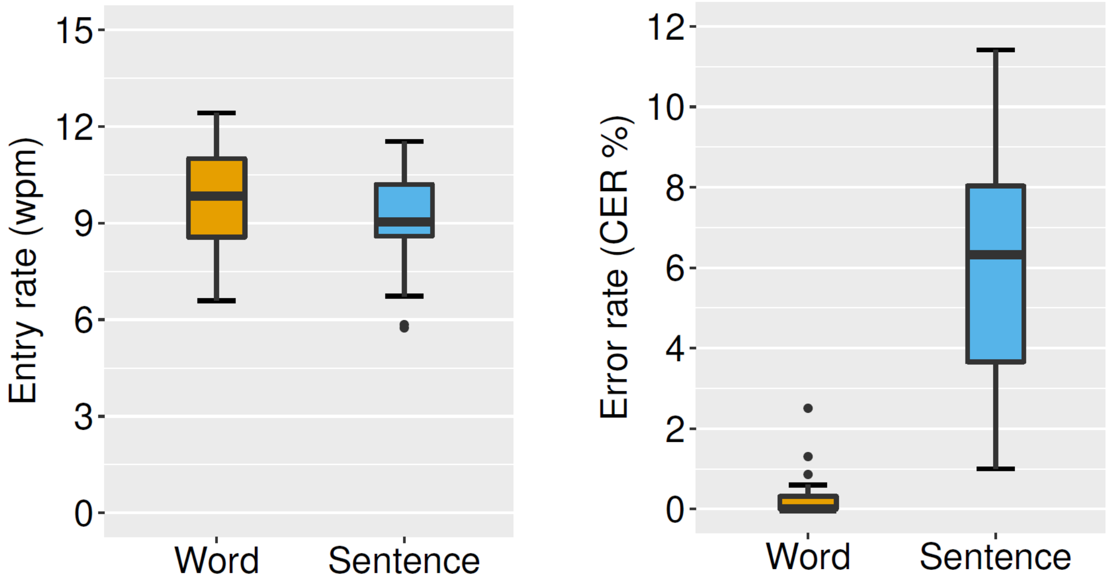
- **Faster than a keyboard with word completions?**
  - Probably only if you are slow at typing!
  - Simulate motor-impairment: users had to dwell click keys



Condition: **Word**

Condition: **Sentence**

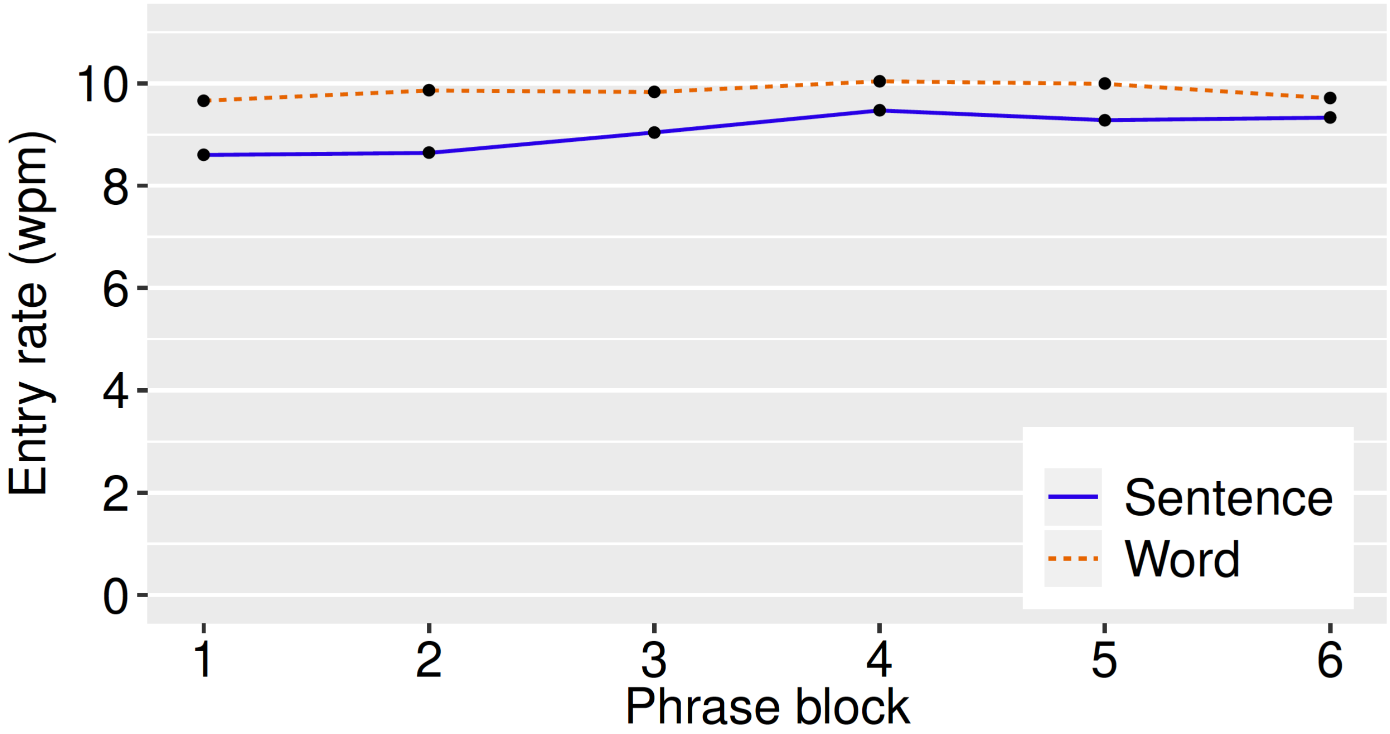
Results from study with 28 participants



Word-at-a-time input and sentence abbreviated input had **similar entry rates**

Much higher error rate of sentence abbreviated input (though some participants did abbreviate according to guidelines)

Entry rate over the session



Participants improved with practice, **similar entry rate in final phrase blocks**

Input of common sentences

- **Can we enable quick input of really common sentences?**
  - Sentence that probably have been **said many times**
  - Previous abbreviation system:
    - Error rate 7.2%, uncomfortably high!
    - May need to abbreviate even more aggressively to be faster
    - Used an AAC-optimized language model, but was still fairly general
    - No correction interface if system got it wrong
  - New effort: Develop a **corpus of frequent conversational phrases**
    - Mined from large data source and ranked using machine learning
    - Predict from the ranked set
    - Improved input and correction interface

Recruiting: Evaluating and Using Sentences in AAC Interfaces

We are looking for volunteers to complete a survey exploring the use of sentences in Augmentative and Alternative (AAC) interfaces. The survey will ask you to **rate how useful a series of sentences might be in an AAC interface**. You will also be asked your opinion about **how sentences might best be incorporated into future AAC interfaces**.

You can complete the study in your web browser. You can complete it over multiple sessions if you like. The survey is completely anonymous.

To participate, you must:

- **Be an AAC user, AAC practitioner, AAC software developer, AAC researcher, or communicate with someone who uses AAC**
- Be 18 years of age or older
- Be a fluent speaker of English
- Have no cognitive impairment

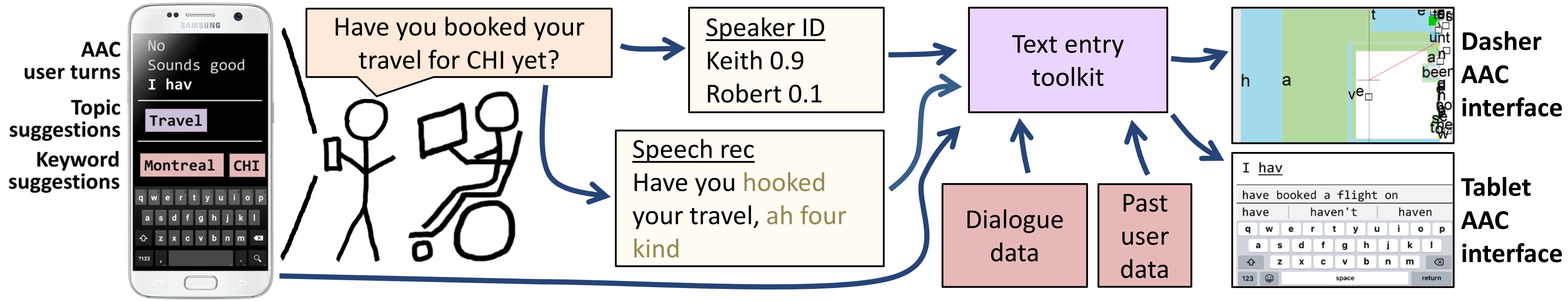
Email Keith Vertanen at vertanen@mtu.edu to be informed when survey opens.

Future directions

- **Bigger and better language modeling**
  - Ensemble of neural and n-gram language models
  - Personalized to a user's past communications
  - Learning and improving as a user writes
  - Open problems:
    - Using large and personalized models while online or offline
    - On low resource devices
    - Maintaining privacy if models are hosted and/or trained online
    - Efficacy on actual AAC user text and in an actual interface

Future directions

- **Context sensitive predictions**
  - Who you are talking to, what they said
  - Where you are, what you are looking at, ...
  - Explicit help (co-construction) from communication partner
  - Open problems:
    - Proving robust gains are possible
    - Privacy concerns of AAC users and their communication partners



Future directions

- **Supporting diverse language needs**
  - Predictions for symbolic communicators
  - Predictions that are literacy level appropriate
    - And change as the user does
  - Supporting non-English speakers, multilingual speakers
  - Enhanced predictions in targeted domains
    - E.g., talk about the weather mode
- **More novel interaction methods**
  - Dasher, Nomon, dwell-free eye-typing
  - What is next???

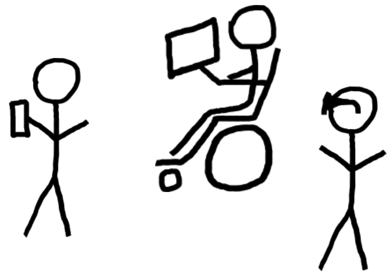
How can you help?

- **Nomon single-switch study**
  - Who: US/UK AAC users with single-switch experience
  - Multiple sessions writing using Nomon

$$$

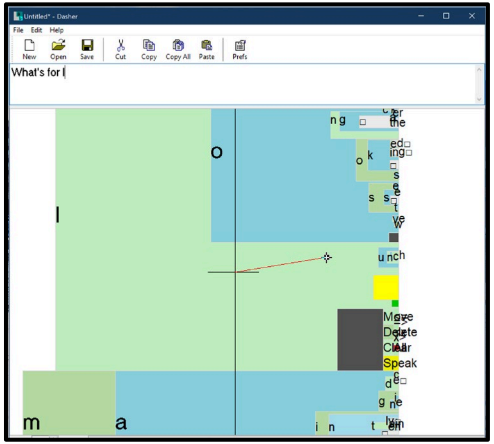
- **Baton AAC history collection**
  - Who: AAC users (via Dasher, Grid3, plain text)
  - Select sentences you think would help AAC researchers

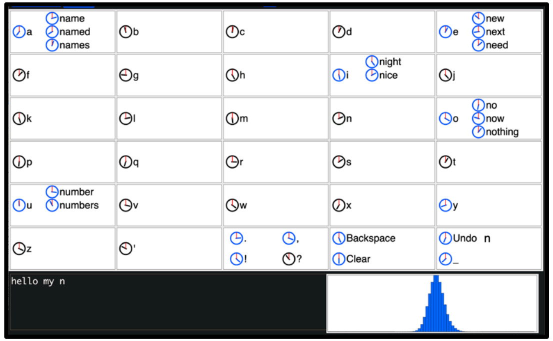
- **Rating conversational utterances**
  - Who: AAC users, AAC practitioners, AAC researchers
  - Rate between 20 and 100 sentences for usefulness
  - Share how you envision AAC interfaces using utterance predictions



Summary

- **Language modeling for predictive AAC**
  - Count things up in a large text file
    - The text matters, as does learning from a user's text
  - Word predictions: many design decisions
- **Two open-source AAC interfaces**
  - Dasher, write by pointing
    - https://github.com/dasher-project/dasher/releases
  - Nomon, flexible single-switch selection
    - https://nomon.csail.mit.edu/html/keyboard.html
  - Try them and send us your feedback!





Keith Vertanen
vertanen@mtu.edu https://keithv.com