Simultaneous Machine Translation

~ Nalin Kumar
Int Msc (Batch 16)
NISER, HBNI
India

SiMT (Simultaneous Machine Translation)

- Translation of continuous input text/speech stream into another language with the lowest latency and highest quality possible, or, generating partial translations before observing the entire source sentence

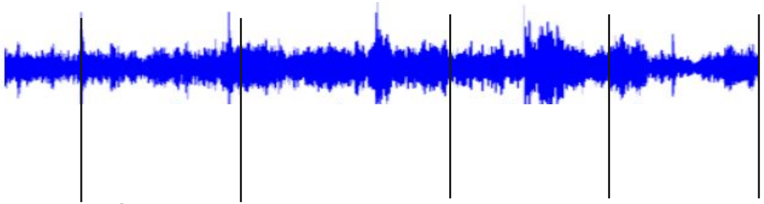
- In summary:
  - translation has to start with an incomplete source text
  - input text/speech is read and translated progressively
  - translated output is based either on some algorithm for anticipation/or wait for a meaningful phrase

Text to Text

Source (Hindi)	Iss	subah	mai	office	jaunga	
Target (English)		This	morning	I	-	will go to office

Speech to Text

Source (Hindi)



Target (English)

This Morning I will go to office

read

write

	This	morning	I	will	go	to	office
iss							
subah							
mai							
office							
jaunga							

Can neural machine translation do simultaneous translation?

~ Kyunghyun Cho, Masha Esipova (New York University) in 2016

- One of the first works involving the attention-based encoder-decoder NMT models.
- Earlier methods usually included two-fold process:
  - segmentation: model first divides a source sentence into phrases.
  - translation of the phrases segmented
- This paper proposes *simultaneous greedy decoding*, that is capable of performing simultaneous translation using an NMT model.
- Unlike the conventional methods, this approach jointly does both the tasks.

Simultaneous greedy decoding

```
Input X; Y = ϕ; Input s_0; Input δ    (s_0 = # of initial inputs;   δ = step size)
s <- s_0 ;  t=1; y_0 = <sos>
while true do:
 if s <= |X|:
     y_t <- NMT(X_s, y_<t)
     if Λ(X_{s+δ}, X_s):                              (Λ: Waiting criteria)
        s <- s + δ (Wait, i.e., prediction made in this step is not  included in the output text)
     else
        Y <- Y ∪ y_t
        t = t+1
 else:
     y_t <- NMT(X_s, y_<t)
     Y <- Y ∪ y_t
     t = t+1
 if  Y[-1] = <eos>
     break
```

Waiting Criteria

1. Wait-If-Worse: Waits if the confidence of current prediction "y_t" is better than the confidence of "y_t" when added $\boldsymbol{\delta}$ more source symbols, i.e., wait if "y_t" has lesser likelihood of coming when there is more source symbols.

   $\log p(y_t|y_{<t},\{NMT, X_s, y_{<t}\}) > \log p(y_t|y_{<t},\{NMT, X_{s+\boldsymbol{\delta}}, y_{<t}\})$

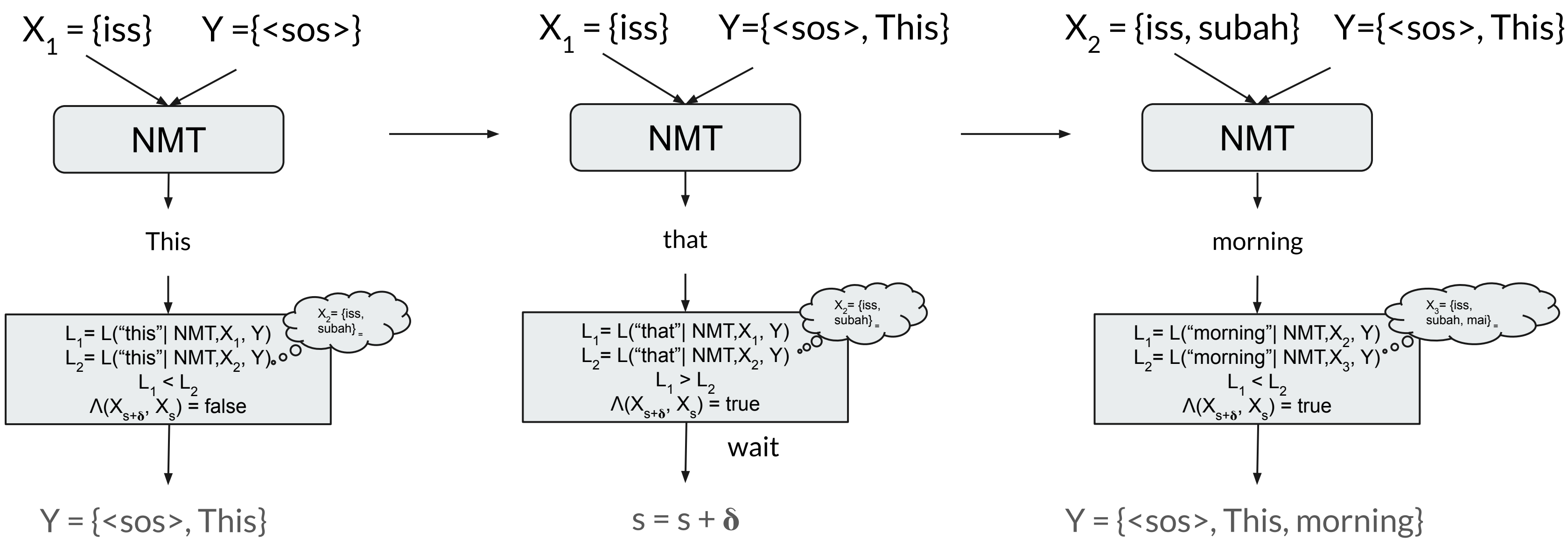
2. Wait-If-Diff: Waits if current prediction does not match with prediction made with adding $\boldsymbol{\delta}$ extra source symbols.

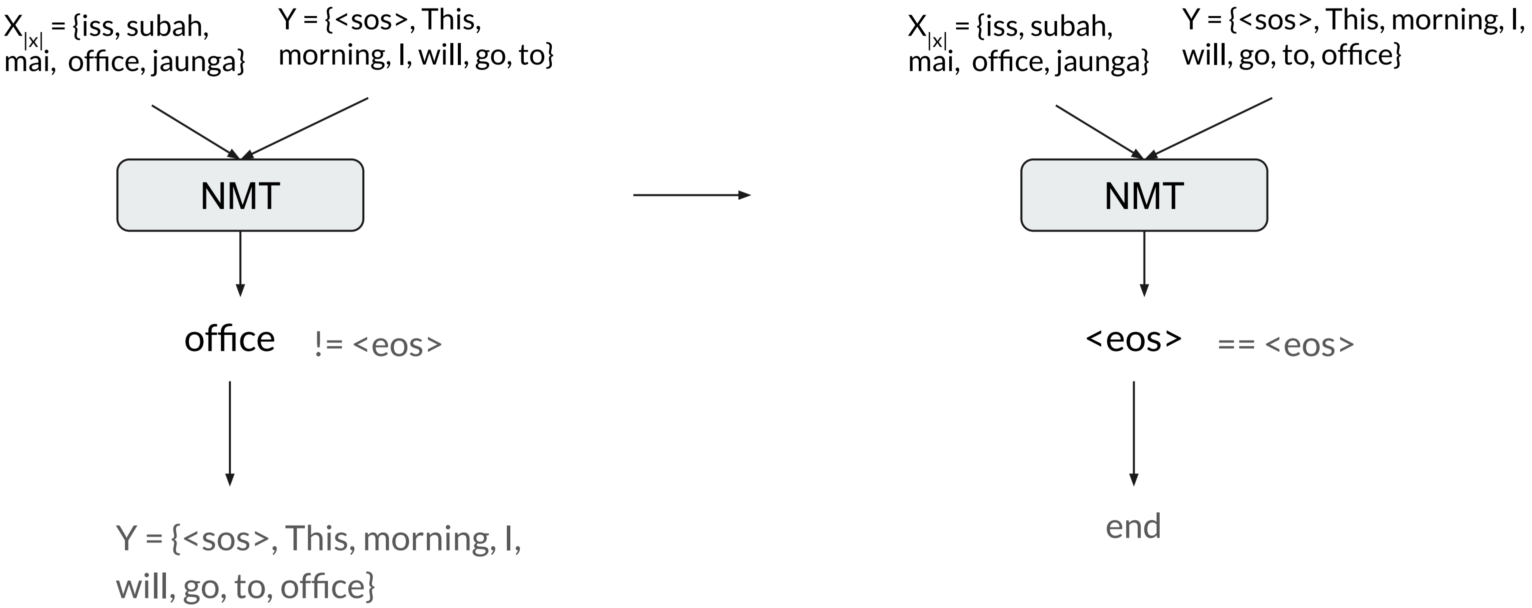
   Different from 1, as there might be a case where $NMT(X_{s+\boldsymbol{\delta}}, Y_{<t}) = NMT(X_s, Y_{<t})$ but $L(NMT(X_s, Y_{<t})) > L(NMT(X_{s+\boldsymbol{\delta}}, Y_{<t}))$

   $\Lambda(X_{s+\boldsymbol{\delta}}, X_s) : NMT(X_{s+\boldsymbol{\delta}}, Y_{<t}) \mathrel{!=} NMT(X_s, Y_{<t}))$

An example for Wait-if-Worse

Let $s_0 = 1$, $\boldsymbol{\delta} = 1$, X = {iss, subah, mai, office, jaunga}, Y= {<sos>}




$X_{|x|}$ = {iss, subah, mai, office, jaunga}
Y = {<sos>, This, morning, I, will, go, to}
NMT
office
!= <eos>
Y = {<sos>, This, morning, I, will, go, to, office}
$X_{|x|}$ = {iss, subah, mai, office, jaunga}
Y = {<sos>, This, morning, I, will, go, to, office}
NMT
<eos>
== <eos>
end

Evaluating AST Systems

- Trade-off between translation quality and time delay
- Translation quality: BLEU
- Time delay (Latency)
  - Average Proportion (AP) [1]: measures the proportion of the area above a policy path
  - Consecutive Wait (CW) [2]: is the number of source words waited between two target words
  - Average Lagging [3]: the goal of AL is to quantify the degree the user is out of sync with the speaker, in terms of the number of source words

1. Cho, Kyunghyun, and Masha Esipova. "Can neural machine translation do simultaneous translation?." *arXiv preprint arXiv:1606.02012* (2016).
2. Gu, Jiatao, et al. "Learning to Translate in Real-time with Neural Machine Translation." *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. 2017.
3. Ma, Mingbo, et al. "STACL: Simultaneous translation with implicit anticipation and controllable latency using prefix-to-prefix framework." *arXiv preprint arXiv:1810.08398* (2018).

Average Proportion (AP)

- For each target symbol $y_{t'}$, how many source symbols were reqd. (s(t))
- Using this, we define the total amount of time spent translating a given source sentence, *delay in translation:*

$$0 < \tau(X, \hat{Y}) = \frac{1}{|X||\hat{Y}|} \sum_{t=1}^{|\hat{Y}|} s(t) \leq 1$$

read

write

	This	morning	I	will	go	to	office
iss							
subah							
mai							
office							
jaunga							

Results (Comparison b/w both directions)

- trade-off between the delay and quality is maintained in both translation directions (En→Cz and Cz→En),
- Wait-if-diff tends to improve the translation quality when translating to En while maintaining the delay

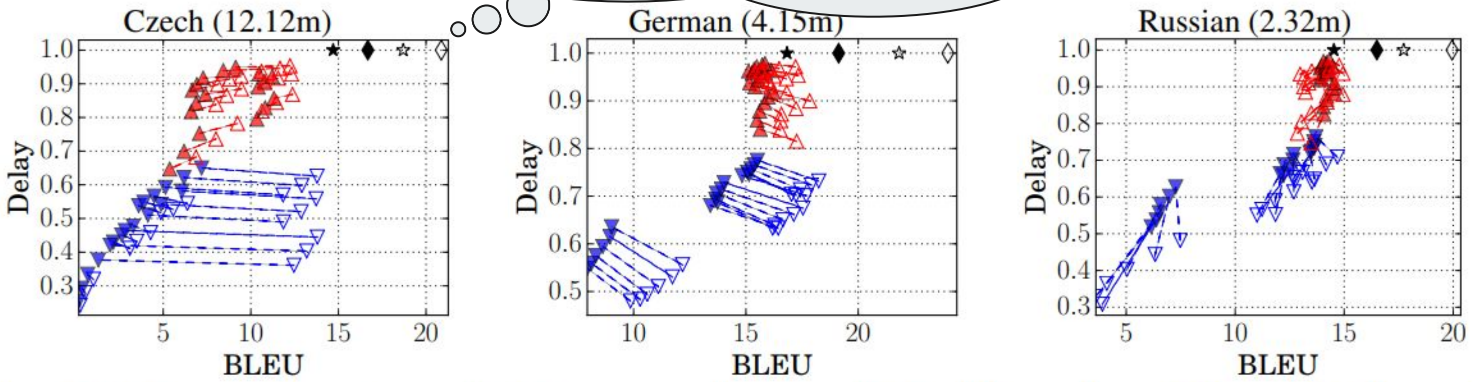


Figure 1: Quality vs. Delay τ plots for all the language pair–directions. ▲: Wait-If-Worse (En→). ▼: Wait-If-Diff (En→). △: Wait-If-Worse (→En). ▽: Wait-If-Diff (→En). ★: consecutive greedy decoding (En→). ♦: consecutive beam search (En→). ☆: consecutive greedy decoding (→En). ◊: consecutive beam search (→En). Each dashed line connects the points with the same decoding parameters (δ and s_0) between translating to and from English. Delay τ: Lower the better. BLEU: Higher the better.

For $s_0 = \{2, 3, 4, 5, 6, 7\}$ and $\delta = \{1, 2, 3\}$

Results (Comparison b/w both directions)

- Delay-quality pattern is maintained in both translation directions (En→De and De→En),
- Delay slightly decreases while increasing the translation quality when translating to English

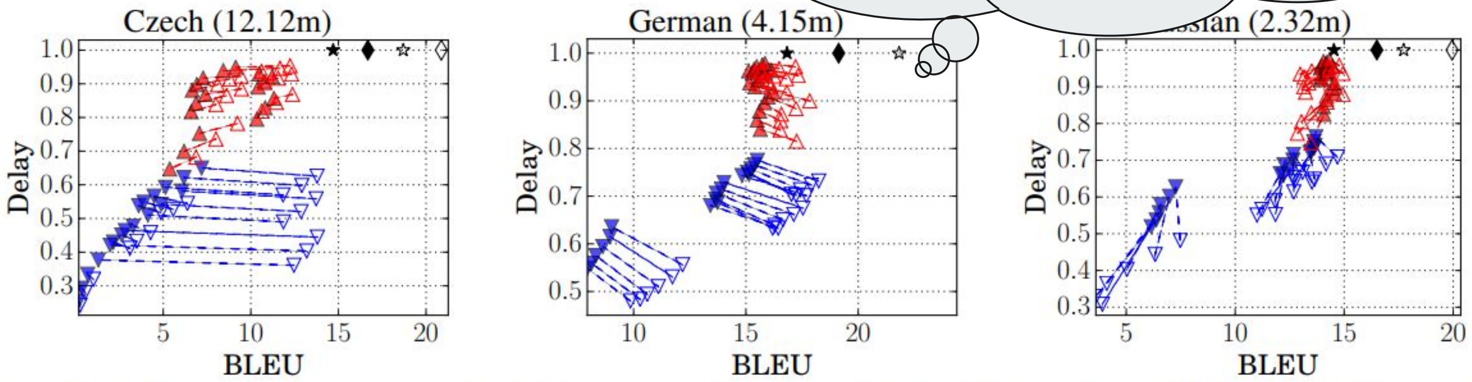


Figure 1: Quality vs. Delay τ plots for all the language pair–directions. ▲: Wait-If-Worse (En→). ▼: Wait-If-Diff (En→). △: Wait-If-Worse (→En). ▽: Wait-If-Diff (→En). ★: consecutive greedy decoding (En→). ♦: consecutive beam search (En→). ☆: consecutive greedy decoding (→En). ◊: consecutive beam search (→En). Each dashed line connects the points with the same decoding parameters (δ and s_0) between translating to and from English. Delay τ: Lower the better. BLEU: Higher the better.

For $s_0 = \{2, 3, 4, 5, 6, 7\}$ and $\delta = \{1, 2, 3\}$

Results (Comparison b/w both directions)

- Delay-quality pattern is somewhat different in both directions
- Translation quality does not improve while translating to En. However, delay is lower when translating to En

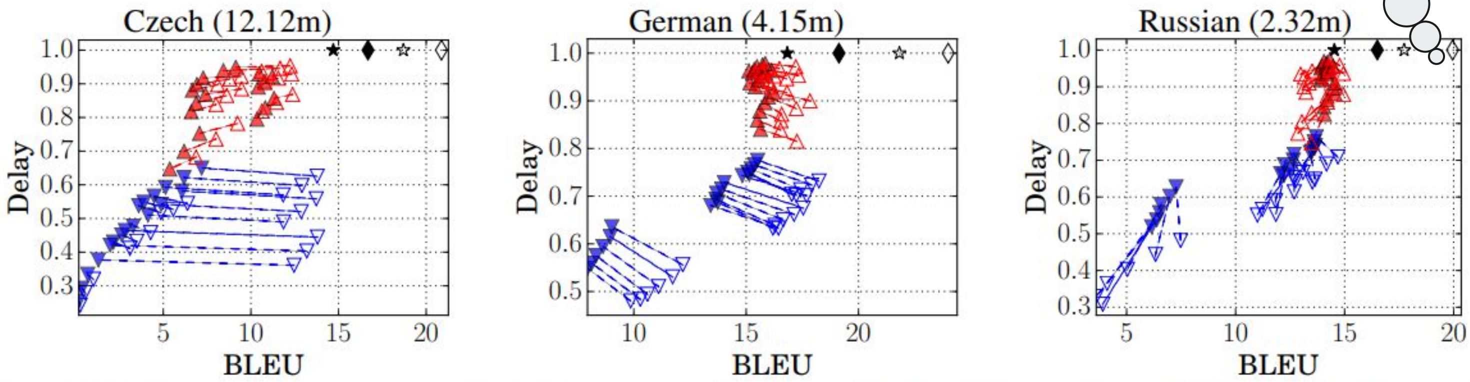


Figure 1: Quality vs. Delay τ plots for all the language pair–directions. ▲: Wait-If-Worse (En→). ▼: Wait-If-Diff (En→). △: Wait-If-Worse (→En). ▽: Wait-If-Diff (→En). ★: consecutive greedy decoding (En→). ♦: consecutive beam search (En→). ☆: consecutive greedy decoding (→En). ◊: consecutive beam search (→En). Each dashed line connects the points with the same decoding parameters (δ and s_0) between translating to and from English. Delay τ: Lower the better. BLEU: Higher the better.

For s_0 = {2, 3, 4, 5, 6, 7} and δ = {1, 2, 3} morphology : how they are formed, and their relationship to other words in the same language.

Results (Comparison b/w both directions)

- Delay decreases when the model translates to English, compared to translating from English, with the exception of Czech and the Wait-If-Worse criterion.
- Reason: richness of morphology (Cz, De, Ru > En)
- i.e., each word in these languages has more information than a usual English word, it becomes easier for the simultaneous greedy decoder to generate more target symbols per source symbol
- The same explanation applies well to the increase in delay when translating from English, as the languages with richer morphology often require complex patterns of agreement across many words.

Results (Comparison b/w both criteria)

- Wait-If-Diff criterion tends to cover wider spectra of the delay and the translation quality
- Wait-If-Worse has more delayed translation and less variance in quality

Coming up next

- Recent Advances (Part 2 and 3)
- Exploring multilinguality in SiMT (Part 4)

Thank You

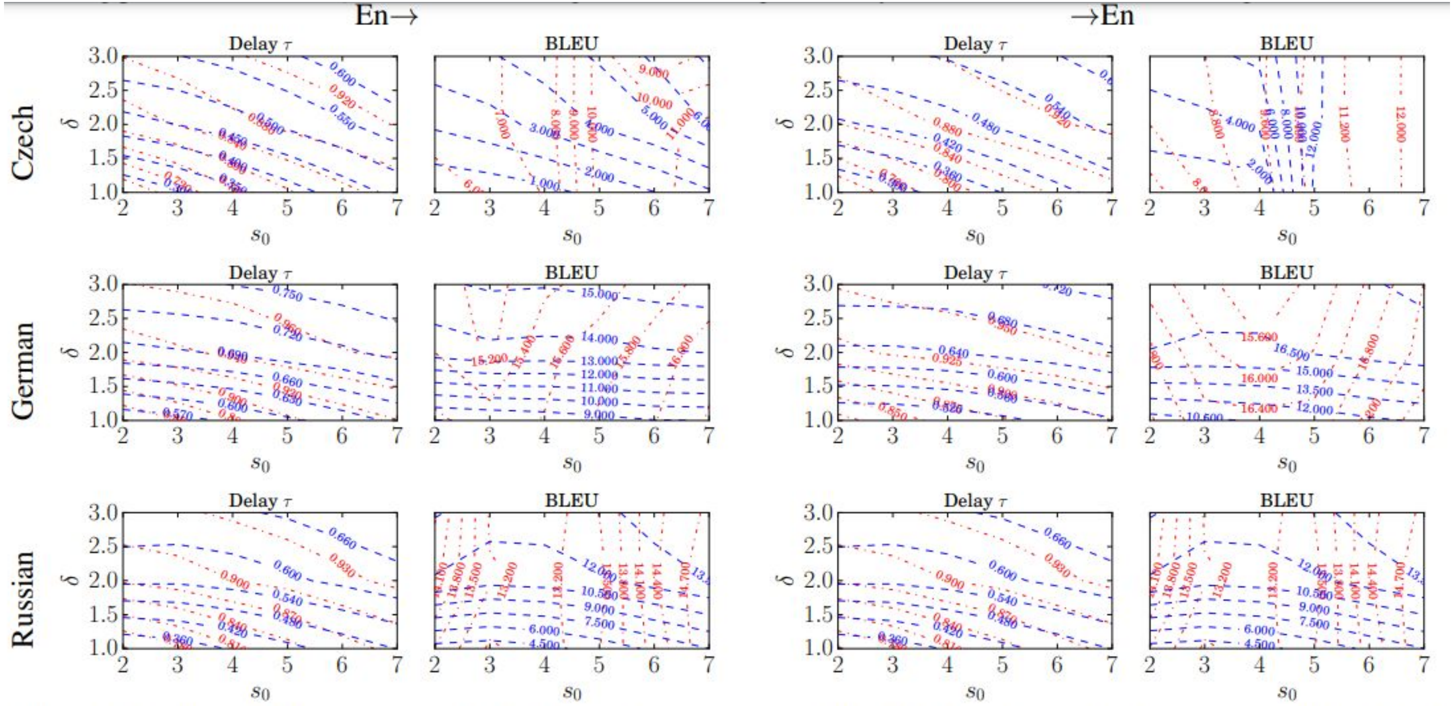


Figure 2: Quality and delay τ per s_0 and δ. Red dash-dot curves (– · –): Wait-If-Worse. Blue dashed curves (– –): Wait-If-Diff.

In Fig. 2, we see a stark difference between these two criteria. This difference reveals itself when we inspect the translation quality w.r.t. the decoding parameters (right panel of each sub-figure). The Wait-If-Worse criterion is clearly more sensitive to s_0, while the Wait-If-Diff one is more sensitive to δ. The only exception is the case of translating Czech to English, in which case the Wait-If-Diff behaves similar to the Wait-If-Worse when $s_0 \geq 5$. On the other hand, the delay patterns w.r.t. the decoding parameters are invariant to the choice of criterion.