Attention ≠ Alignment

低頻度語は
特別処理

- 横方向に足すと1
- 縦方向は足しても1にはならない
  → 全ての入力がカバーされていない
- Alignment Error Rate
  GIZA = 30ぐらい
  Attention = 50ぐらい
  [Liu et al., 2016a]
- GNMTにおいて単語対応が表示されない要因

NMTの課題の整理

- 扱える語彙数が少ない
  - [Luong et al., 2015b], [Jean et al., 2015]
  - [Costa-jussà and Fonollosa, 2016], [Chung et al., 2016], [Luong and Manning, 2016], [Sennrich et al., 2016b]
- 訳抜けと重複
  - [Tu et al., 2016a], [Tu et al., 2016b]
- 何を学習しているのかわからない
  - [Shi et al., 2016a], [Shi et al., 2016b]

NMTにおける語彙サイズの問題

- Softmaxの計算が重たいため、語彙サイズを制限
  - 頻度順で上位3万から5万程度、多くても10万
- 語彙範囲外の単語は特別な記号 <UNK> に置換
- 出力中の <UNK> に対応する単語を辞書等で翻訳
  - 単語アライメント結果を使って対訳文を修正 [Luong et al., 2015b]

En: The ***unk*** portico in ***unk*** …

Fr: Le $\underline{\textbf{unkpos}}_{1}$ $\underline{\textbf{unkpos}}_{-1}$ de $\underline{\textbf{unkpos}}_{0}$

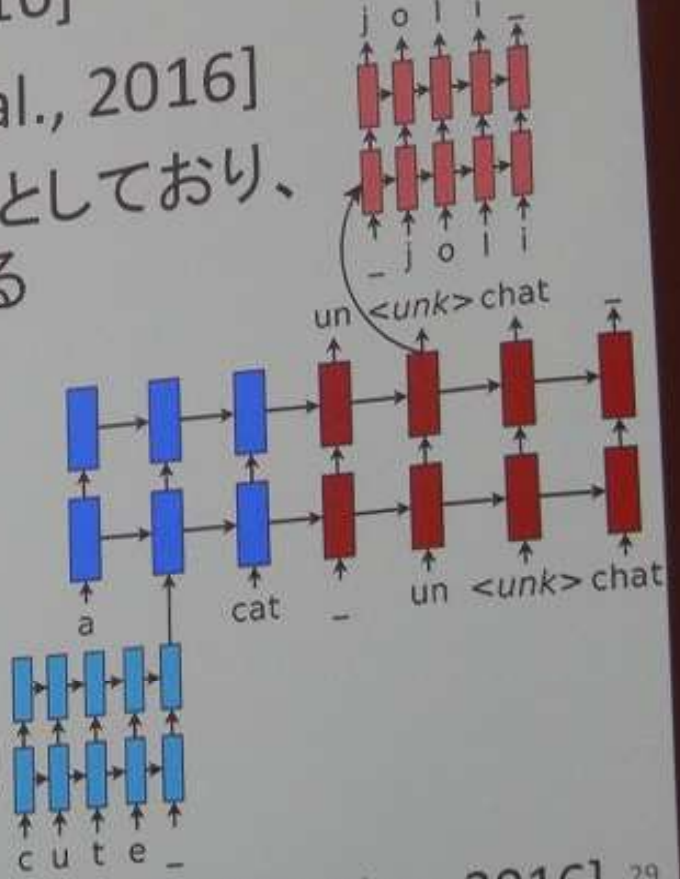
添え字は対応する入力単語の相対位置

  - <UNK>を出力する際に最も強くattentionした入力単語で置き換え [Jean et al., 2015]

これらの方法は copy model/mechanism と呼ばれる

単語ではなく文字を使う

- 入力のみ文字 [Costa-jussà and Fonollosa, 2016]
- 出力のみ文字 [Chung et al., 2016]
- 入力も出力も文字のみ [Lee et al., 2016]
  - 上記3つは単語間の空白も1文字としており、間接的に単語の情報を使っている
- 単語と文字のハイブリッド
  - 単語単位のNMTがベース
  - 入力の<UNK>は文字単位のエンコーダーが表現を作る
  - 出力の<UNK>は文字単位で翻訳



[Luong and Manning, 2016]

単語と文字の中間的な単位(sub-word)

[Sennrich et al., 2016b]

- Byte Pair Encoding (BPE) https://github.com/rsennrich/subword-nmt
  - データ圧縮方法として提案されたアルゴリズム (1994)
  - 全ての文字を語彙に登録するところからスタート
  - データの中で最も頻度の高い2文字の連続を新たな語彙として登録
  - 設定された最大語彙サイズまで登録を繰り返す
- Wordpiece Model (WPM)
  - Googleが使っているsub-word unit (BPEと同じ)
- SentencePiece https://github.com/google/sentencepiece
  - Googleの工藤さんが作ったもの
  - 事前単語分割不要で、文から直接sub-wordを学習

Byte Pair Encodingのアルゴリズム

[Sennrich et al., 2016b]

コーパス

頻度	単語	単語	単語	単語	単語
5	l o w	l o w	l o w	l o w	low
2	l o w e r	l o w e r	l o w e r	l o w e r	low e r
6	n e w e s t	n e w e s t	n e w e s t	n e w e s t	n e w e s t
3	w i d e s t	w i d e s t	w i d e s t	w i d e s t	w i d e s t

語彙 (サイズ = 15)

初期語彙 = 文字 (11個)

l, o, w, e, r, n, w, s, t, i, d

es (頻度 = 9)　lo (頻度 = 7)

est (頻度 = 9)　low (頻度 = 7)

Sub-wordの影響 (GNMTの例)

逗子市小坪5-1の小坪海岸トンネル鎌倉側で、9月24日0時頃、大きな崖崩れが発生しました。

⇨ A large cliff collapse occurred around 0 o'clock on September 24th at the Kobosa coast tunnel Kamakura side of Zushi-shi Kobosa 5-1.

小坪 ⇨ Koonsubo

小坪海岸 ⇨ Kobosu coast

逗子市小坪 ⇨ Zushi-shi Kosubo

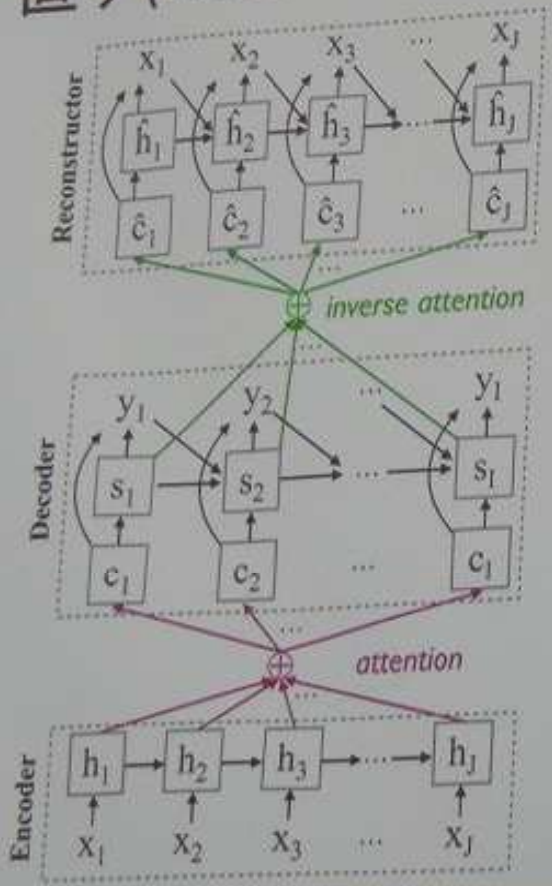
逗子市の小坪 ⇨ Zushi in Zushi City

※ 豊橋技科大 井佐原先生のスライドより

Neural Machine Translation with Reconstruction

[Tu et al., 2016a]

- 通常のNMTで翻訳し、さらにそれを原文に翻訳し直す(Reconstructor)モジュールを追加



モデル	訳抜け	重複(過剰訳)
ベースライン	18.2%	3.9%
+ reconstruction	16.2%	2.4%

Modeling Coverage for Neural Machine Translation [Tu et al., 2016b]

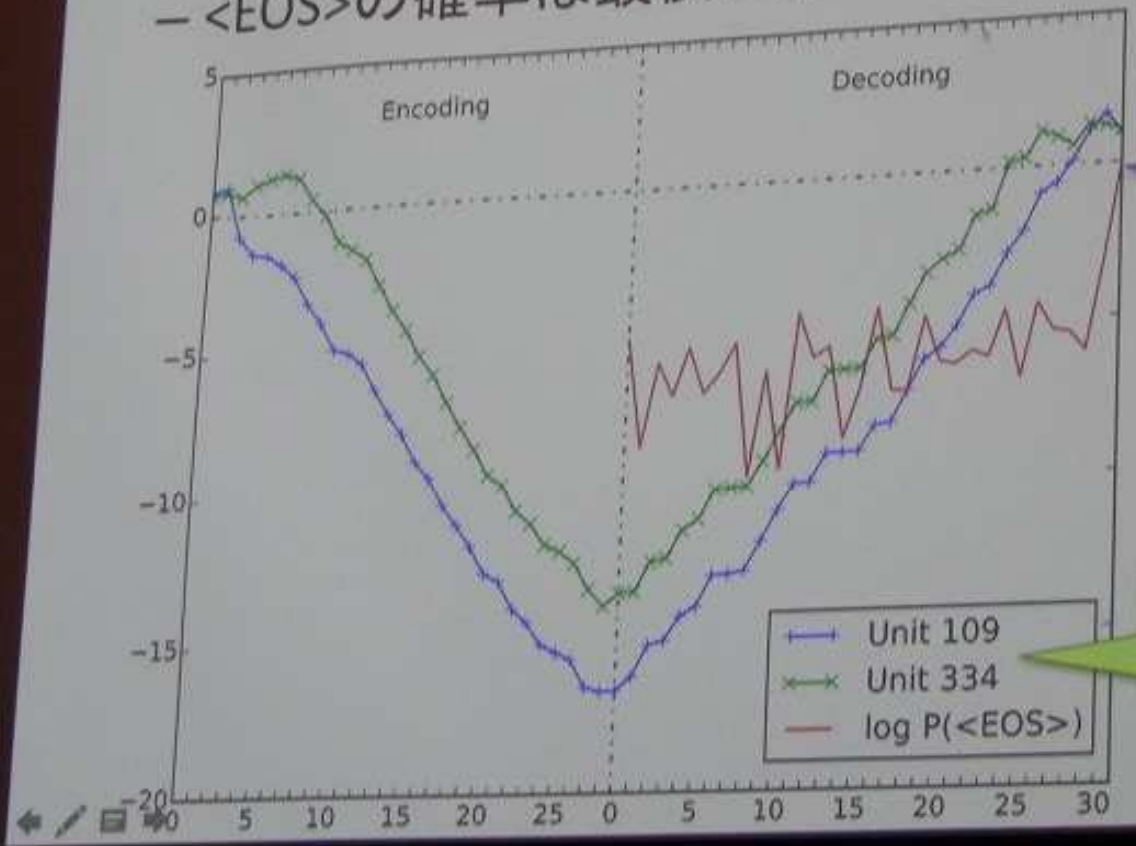
- 入力文のどの単語が翻訳されたかを追跡するカバレージベクトルを追加
- カバレージベクトルをattentionの計算に利用
- 根本解決からは程遠い

System	SAER	AER
GroundHog	67.00	54.67
+ Ling. cov. w/o fertility	66.75	53.55
+ Ling. cov. w/ fertility	64.85	52.13
+ NN cov. w/o gating ($d = 1$)	67.10	54.46
+ NN cov. w/ gating ($d = 1$)	66.30	53.51
+ NN cov. w/ gating ($d = 10$)	**64.25**	**50.50**

Why Neural Translations are the Right Length

[Shi et al., 2016a]

- attentionなしの翻訳モデルの隠れ層を分析
  - 出力の長さをコントロールしているunitが複数存在
  - <EOS>の確率は最後に急に高くなる



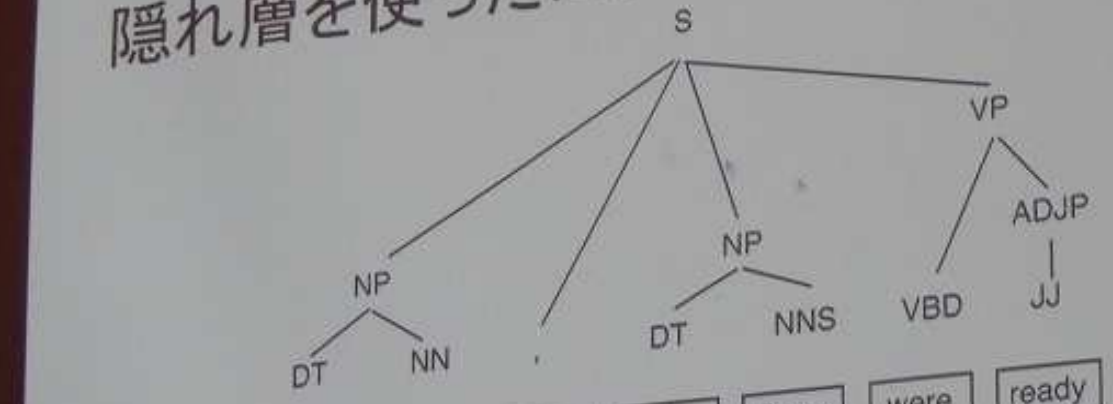
<EOS>の確率は
最後だけ高い

実験ではこの2つの
Unitが最も長さに関与

Does String-Based Neural MT Learn Source Syntax?

[Shi et al., 2016b]

- attentionなしの翻訳モデルの入出力を様々に変え、隠れ層を使ったロジスティック回帰で以下を予測

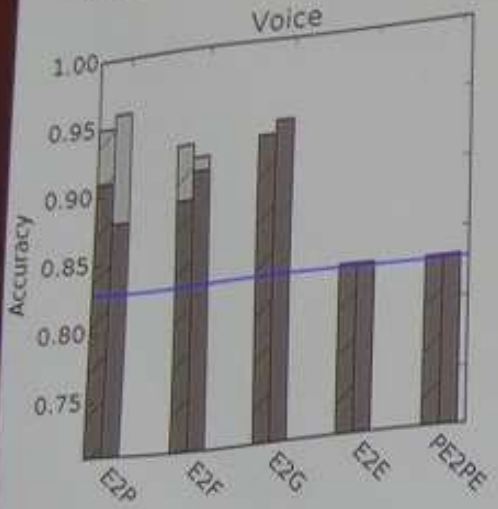


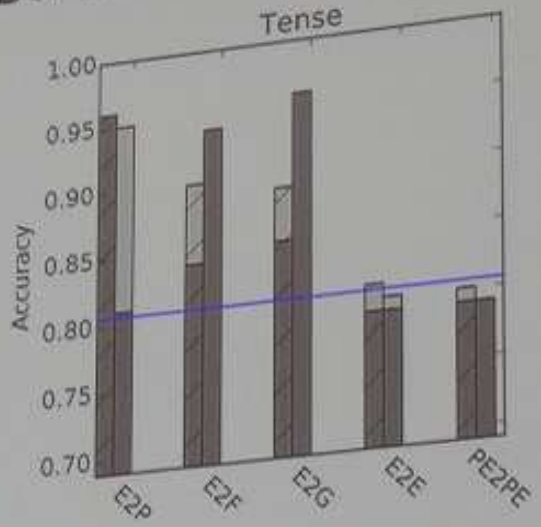
POS	DT	NN	,	DT	NNS	VBD	JJ
Smallest Phrase Constituent	NP	NP	-	NP	NP	VP	ADJP
Top-level Syntactic Sequence	(NP, NP, VP)						
Voice	Active						
Tense	Past						

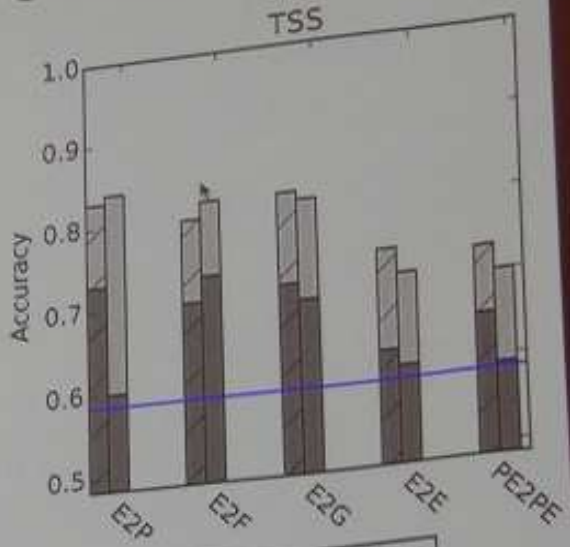
各単語を読み込んだ時点での隠れ層の状態を利用

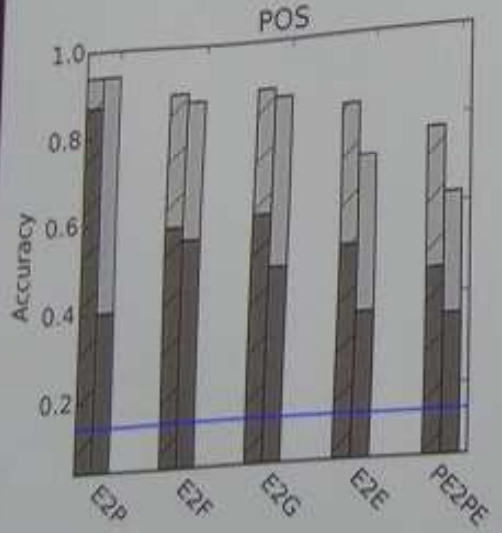
隠れ層の最後の状態を利用

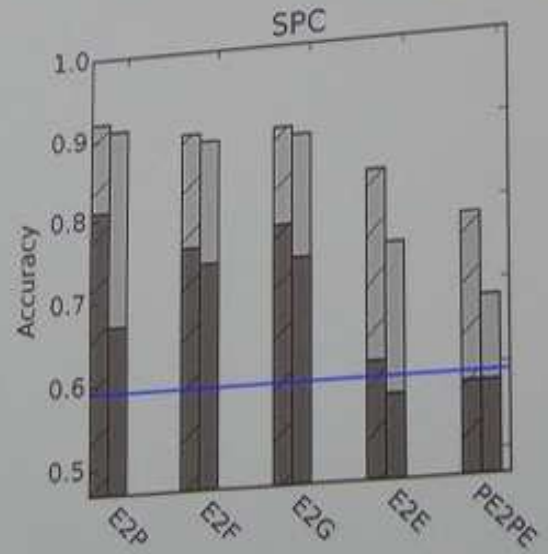
Does String-Based Neural MT Learn Source Syntax?

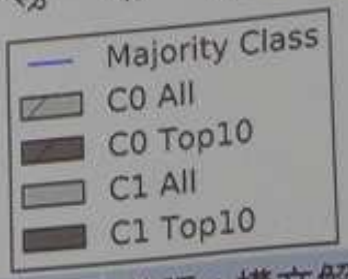








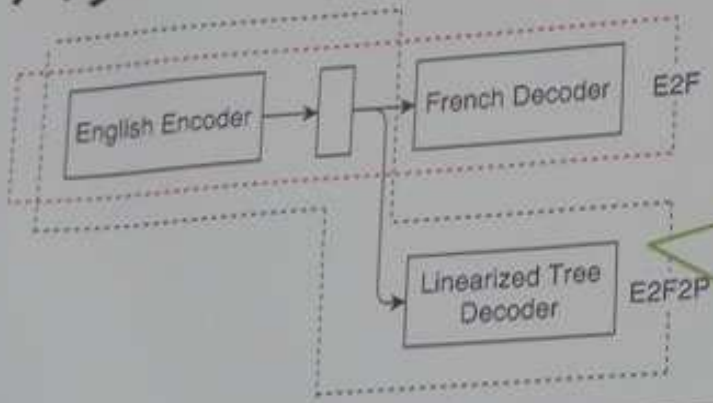


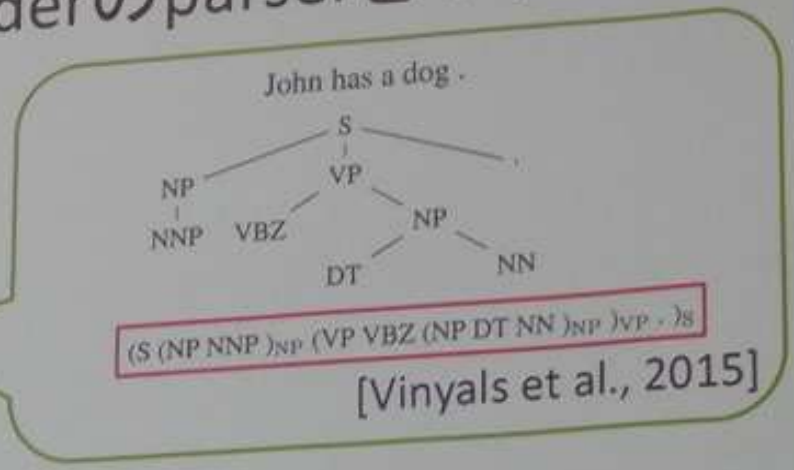


E2P	英語→構文解析
E2F	英語→仏語
E2G	英語→独語
E2E	autoencoder
PE2PE	autoencoder (語順バラバラ)

Does String-Based Neural MT Learn Source Syntax?

- まず普通にencoder-decoderを学習し、encoderのパラメータを固定してdecoderのparserを学習





[Vinyals et al., 2015]

Model	Labeled F1	POS Tagging Accuracy
PE2PE2P	58.67	54.32
E2E2P	70.91	68.03
E2G2P	85.36	85.30
E2F2P	86.62	87.09
E2P	93.76	96.00

autoencoderで学習された情報ではparseできない → 構文情報は学習されていない

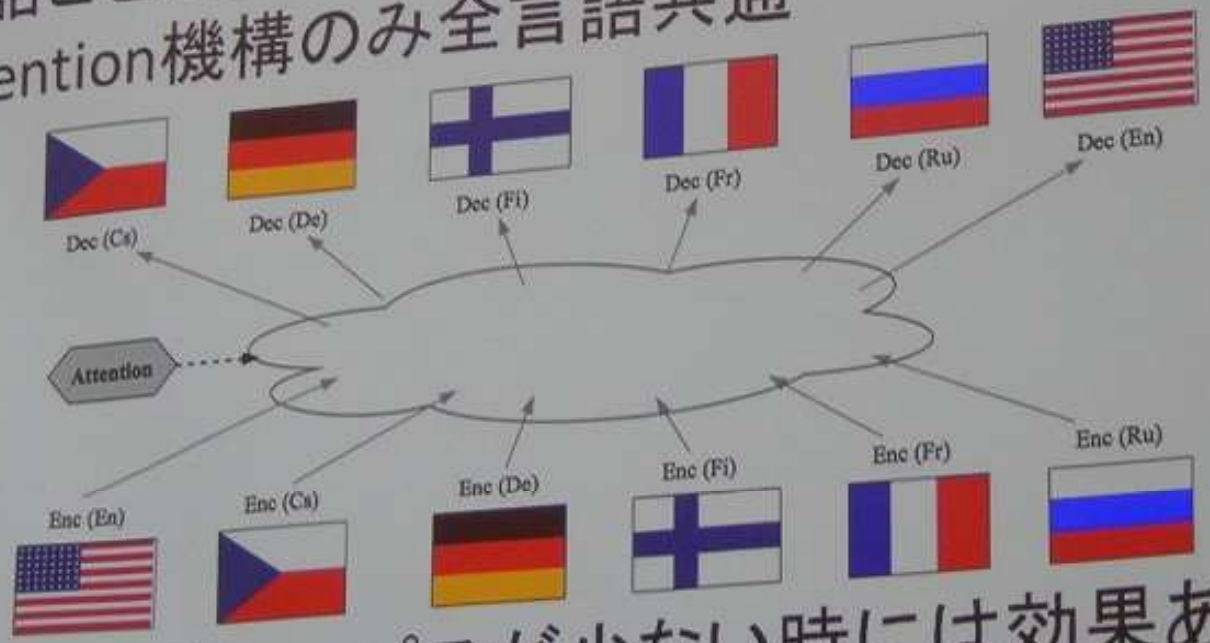
NMTで学習された情報ならある程度parseできる → なんらかの構文情報が学習されている

多言語化が容易

- SMTは基本的に記号の置き換えであるため、多言語対訳コーパスを同時に利用することは困難
- NMTは基本的に数値計算なので、言語に依らず同じ意味を表すものを同じような値に変換できれば翻訳可能
  - 昔からある中間言語のようなもの
- 直接の対訳コーパスがない言語対であっても翻訳可能（ゼロショット翻訳）
  - SMTでは英語などをピボット言語として用い、二段階に翻訳するなどする必要があった

Multi-Way, Multilingual Neural Machine Translation with a Shared Attention Mechanism [Firat et al., 2016a]

- 言語ごとにencoderとdecoderを用意
  attention機構のみ全言語共通



- 直接の対訳コーパスが少ない時には効果あり

Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation [Johnson et al., 2016]

- 入力文に <2es> (スペイン語への翻訳)のようなタグをつけ、全ての言語対の対訳コーパスを同時に使うだけでゼロショット翻訳もできるようになる
- "<2en> 私は東京大学학생입니다" みたいなことも
- ちなみに語彙サイズは全言語共通で32k (WPM)

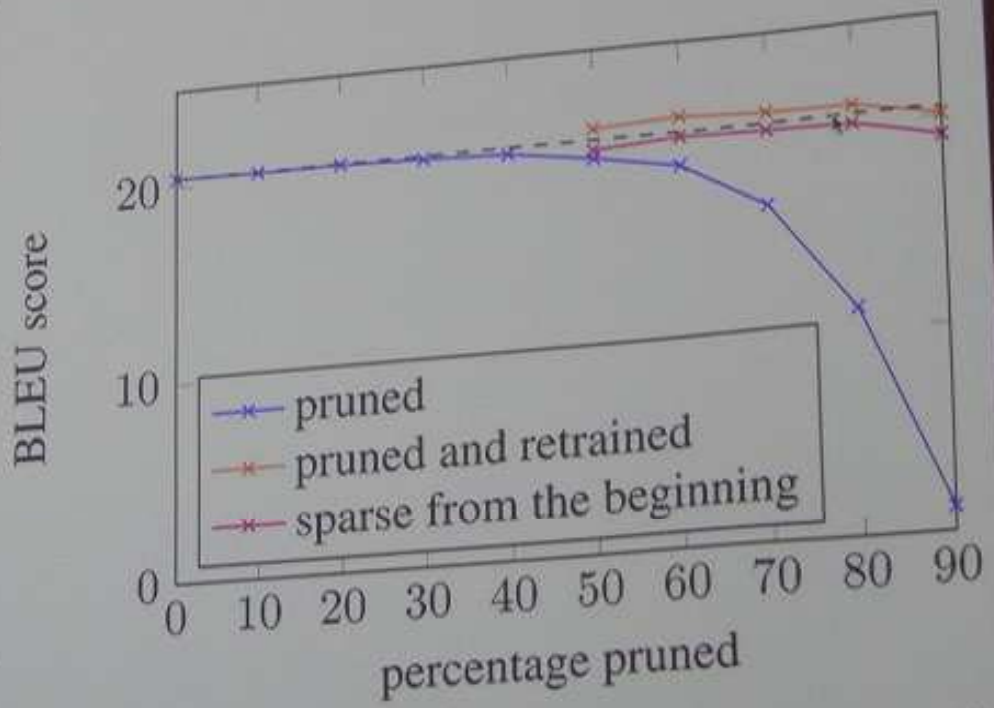
モデルの軽量化

- SMTのモデルは巨大だった・・・
  - フレーズテーブル、言語モデルなどなど
  - 数十GB、数百GB、数TB
- NMTは実数の行列を保存しておけばよい
  - ネットワークの大きさによるが、せいぜい数GB
- NMTのモデルをさらに軽量化する方法もある
  - 量子化 [Wu et al., 2016]
  - 枝刈り (pruning) [See et al., 2016]
  - 蒸留 (distillation) [Kim and Rush, 2016]

Compression of Neural Machine Translation Models via Pruning

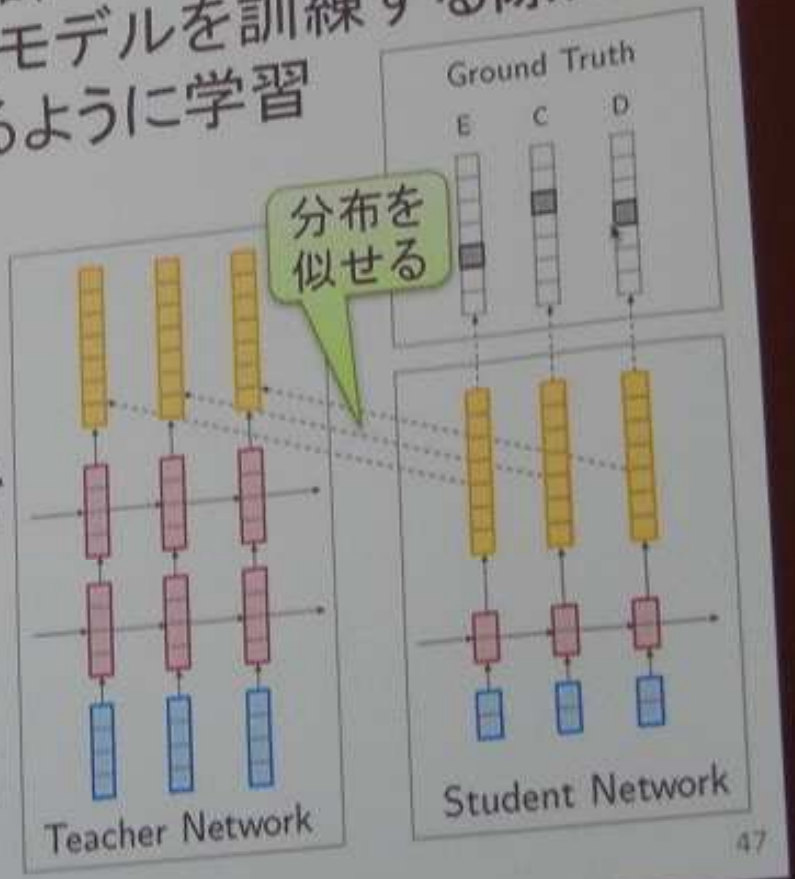
[See et al., 2016]

- パラメータの絶対値が小さいものから順に枝刈り
  - パラメータと同じshapeのmask行列を使って値を0に
- 枝刈りすると精度は下がるが、再訓練すれば元の精度まで戻せる
- 初めから枝刈りした状態で訓練しても だいたい同じ精度に



Sequence-Level Knowledge Distillation [Kim and Rush, 2016]

- 知識蒸留
  - 大きなネットワークからなる教師モデルを訓練しておき、より小さなネットワークの生徒モデルを訓練する際に教師モデルの予測分布に似るように学習
  - 1つだけの正解から学習するよりも、教師モデルの予測分布を使えるため、効率が良い
- 1単語出力するごとに分布を似せるだけでなく、出力文全体としての分布を似せる方法(sequence-level)も提案



Sequence-Level Knowledge Distillation [Kim and Rush, 2016]

- 教師モデルより1/5から1/6の生徒モデルでも
  教師モデルと遜色ない精度を達成
- なぜかビームサーチをしなくても精度が出るように
- Galaxy 6上でNMTが動く！(iPhone 8/Xなら・・・)

1秒間に翻訳できる単語数

Model Size	GPU	CPU	Android
Beam = 1 (Greedy)			
4 × 1000	425.5	15.0	–
2 × 500	1051.3	63.6	8.8
2 × 300	1267.8	104.3	15.8
Beam = 5			
4 × 1000	101.9	7.9	–
2 × 500	181.9	22.1	1.9
2 × 300	189.1	38.4	3.4

さらに枝刈りも実施

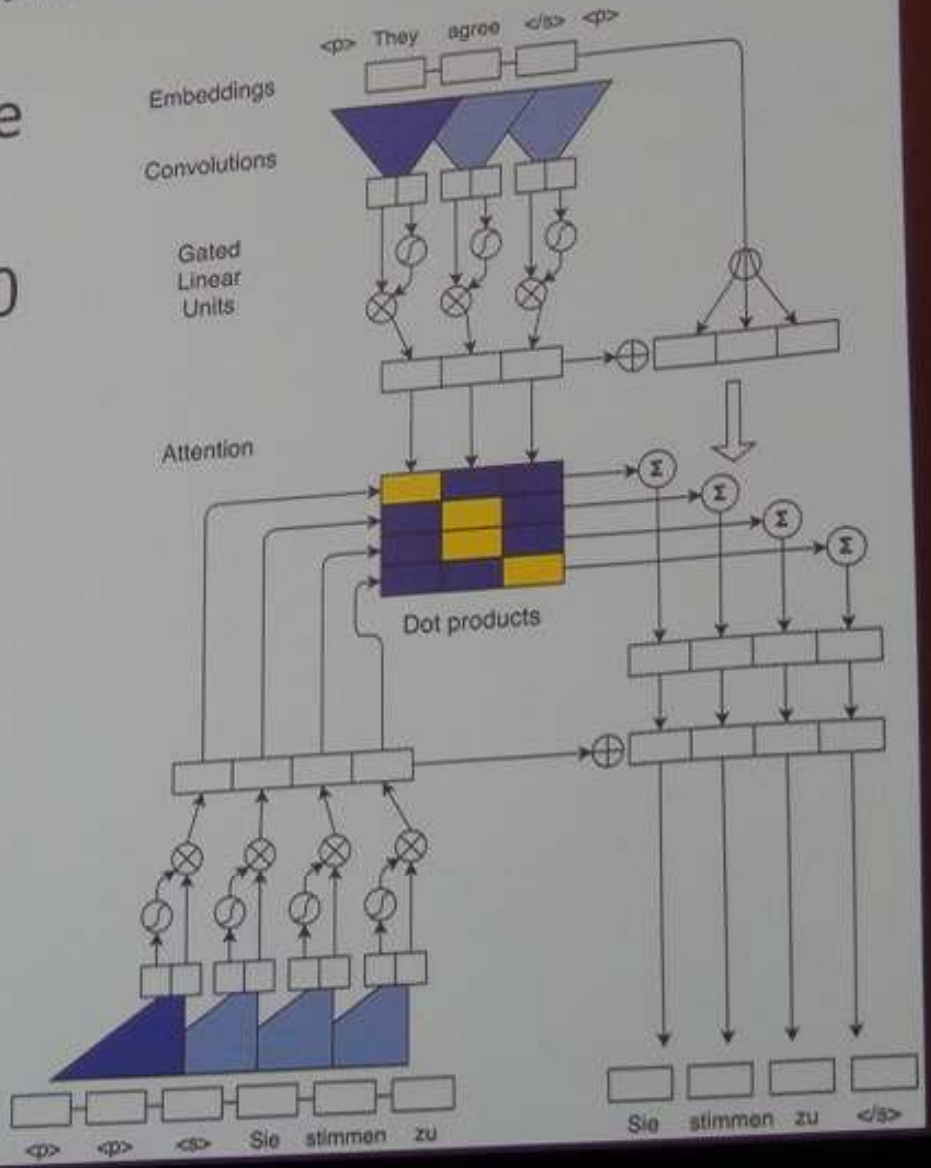
Model	Prune %	Params	BLEU	Ratio
4 × 1000	0%	221 m	19.5	1×
2 × 500	0%	84 m	19.3	3×
2 × 500	50%	42 m	19.3	5×
2 × 500	80%	17 m	19.1	13×
2 × 500	85%	13 m	18.8	18×
2 × 500	90%	8 m	18.5	26×

その他のNMTモデル

- Convolutional Sequence to Sequence Learning
  - https://arxiv.org/abs/1705.03122
  - RNNではなくCNNを使うことで高速化
- Attention Is All You Need
  - https://arxiv.org/abs/1706.03762
  - RNNもCNNもいらない！Feed-forwardのみ
  - Self-attentionにより代名詞の実体も考慮できる
- Unsupervised Neural Machine Translation
  - https://arxiv.org/abs/1710.11041
  - 対訳コーパスいらない！

その他のNMTモデル (1/3)

- Convolutional Sequence to Sequence Learning
  - https://arxiv.org/abs/1705.03122
  - RNNではなくCNNを使うことで高速化



その他のNMTモデル (2/3)

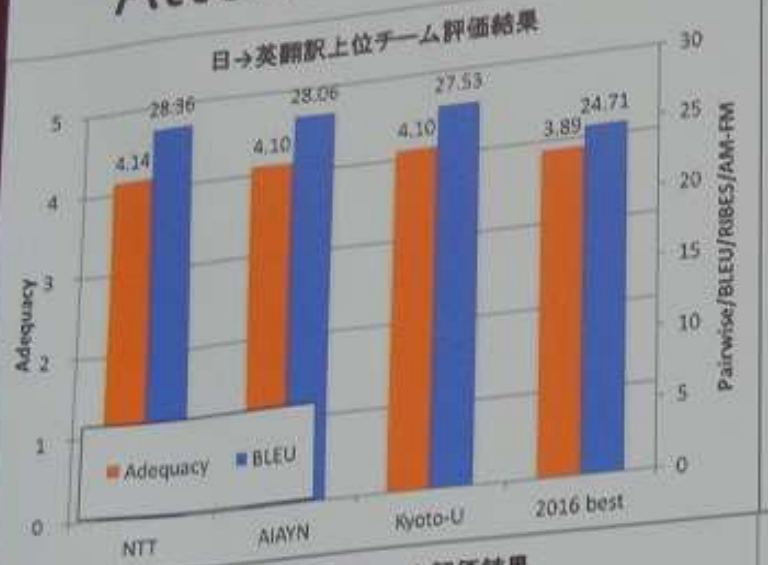
- Attention Is All You Need
  - https://arxiv.org/abs/1706.03762
  - RNNもCNNもいらない！Feed-forwardのみの Transformerを提案
  - Self-attentionにより代名詞の実体も考慮できる

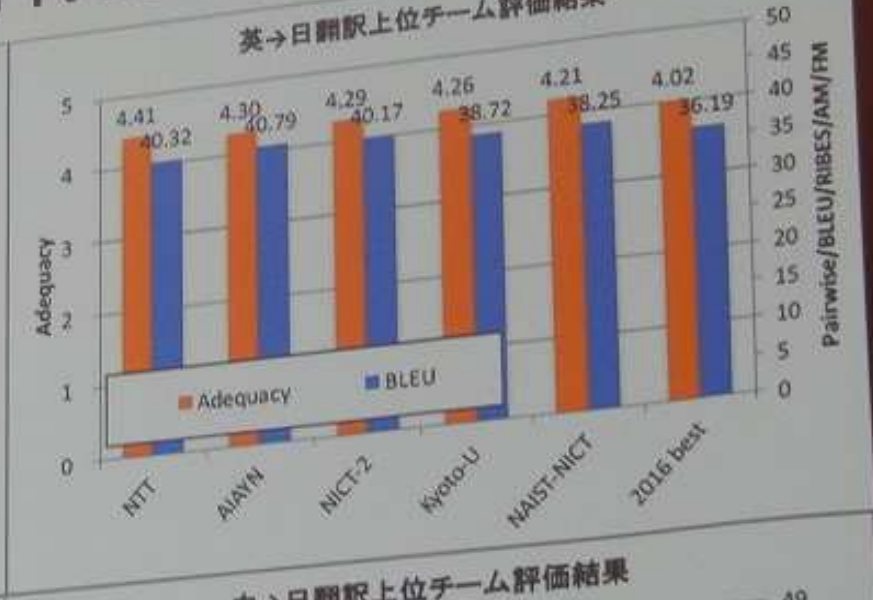
The animal didn't cross the street because it was too tired.
L'animal n'a pas traversé la rue parce qu'il était trop fatigué.

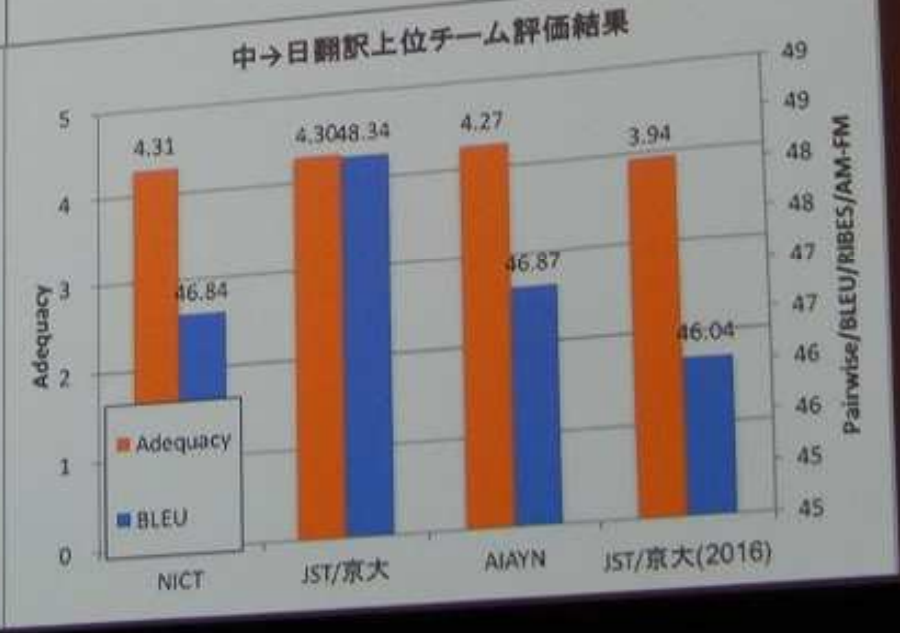
The animal didn't cross the street because it was too wide.
L'animal n'a pas traversé la rue parce qu'elle était trop large.

https://research.googleblog.com/2017/08/transformer-novel-neural-network.html

Attention Is All You Need (AIAYN) 強し







その他のNMTモデル (3/3)

- Unsupervised Neural Machine Translation
  - https://arxiv.org/abs/1710.11041
  - 対訳コーパスいらない！(単言語コーパスだけで翻訳)
  - Unsupervised cross-lingual embeddingを使い、NMTの学習中は更新しない
  - Encoderは両言語共通のものを使う(decoderは独立)
  - 語順を入れ替えた文を入力として元の語順を復元する*denoising*と、通常の入力を相手言語に翻訳し、さらに元の言語に逆翻訳する*backtranslation*をバッチごとに入れ替えて訓練

その他のNMTモデル (3/3)

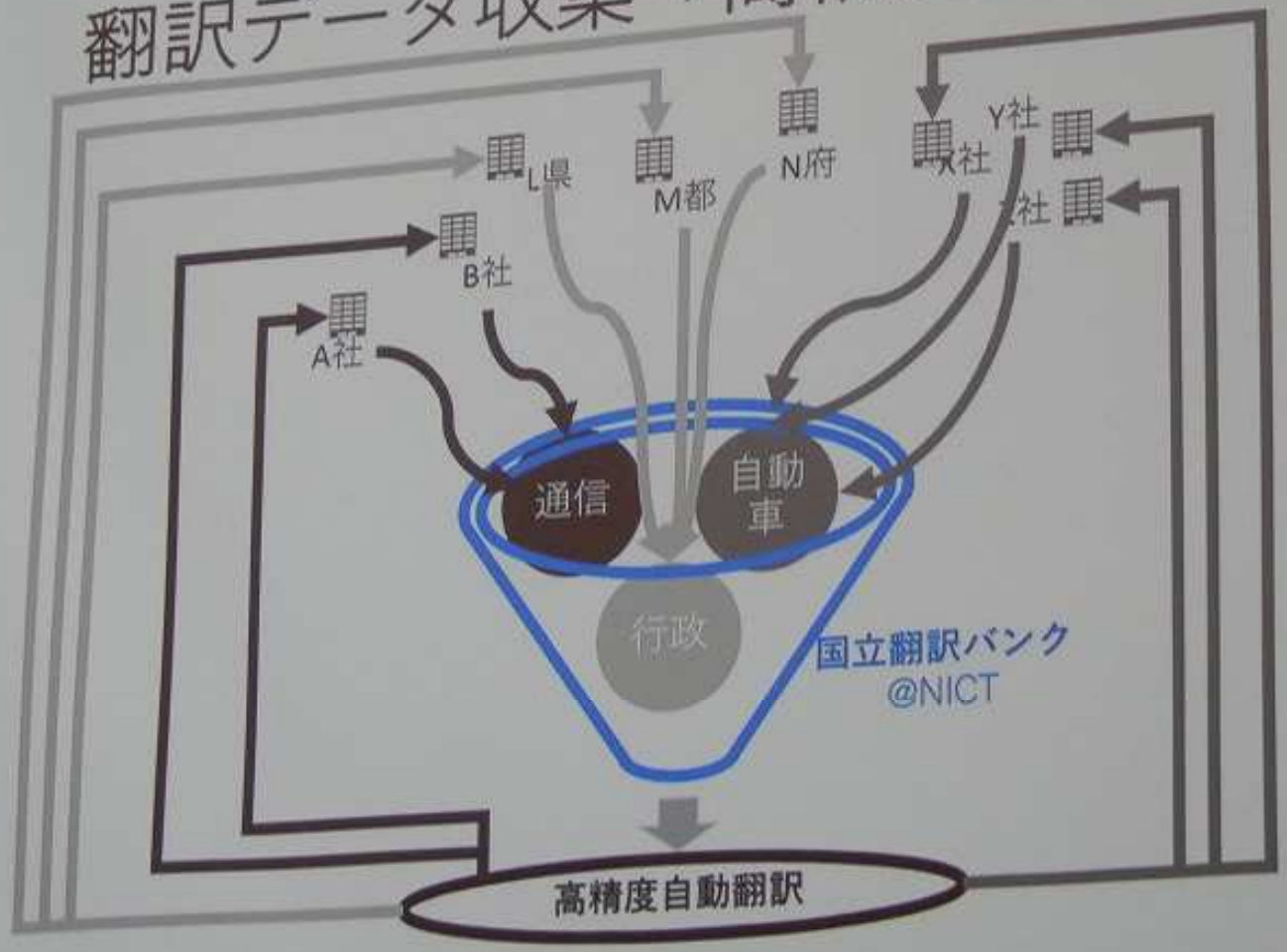
- Unsupervised Machine Translation Using Monolingual Corpora Only
  - https://arxiv.org/abs/1711.00043
  - 前のスライドの論文の翌日に投稿された！
  - denoisingとbacktranslationを使うのは同じ、embeddingとencoderは各言語独立
  - 違いはencoderの出力を言語非依存の空間にmapするために敵対性学習を行う
    - どちらの言語でも同じ内容の文は同じようにencodeされる
    - 敵対ネットワークはencodeされたものがどちらの言語のものかを当てる

翻訳エンジンには価値がない？

- すでに多くのNMTエンジンがオープンソース
  - SYSTRANも使っているエンジン: OpenNMT
  - 最先端の研究成果も利用可能
- 他の大手企業も技術を論文で公開
  - Baidu [He, 2015], Google [Wu et al., 2016]
- SMTのように開発に職人技が必要なこともない
  - 既存のモデルなら学生が1週間かければ作れる
  - 逆にいうと、MT研究への参入障壁が大幅に低下
- じゃあ（企業にとって）何が重要なのか？
  - おそらくデータ、あとエンジンを使いこなせる人

日本でのデータ共有の試み

翻訳データ収集⇒高精度自動翻訳



http://www.nict.go.jp/data/presentation/open-house2016/nict-open-house-2016-presentation-1-2.pdf

今後の展望

- NMTはここ数年で急激に発展し、SMTの精度を追い越している
- NMTの研究はまだ発展する可能性が高い
- 現状のNMTには解決すべき課題が多く残されており、実用的かと言われると疑問が残る
  - Gisting目的ならば十分
  - SMTが活躍する場もまだ残っていることは確かで、うまく組み合わせられると良い
  - 特に対訳コーパスが少量の場合NMTはSMTよりも弱い
  - NMTの発展次第ではSMTが遺産になる可能性もある

関東/関西MT勉強会

- https://sites.google.com/site/machinetranslationjp/
- MTに関する話題をざっくばらんに扱っています
- 内容は基本的にオフレコです
- 参加自由、研究者ではない方もぜひご参加下さい
- 次回は11/18（土）@NAIST