Confusion Forest に基づく機械翻訳システムコンビネーション

渡辺 太郎　　隅田 英一郎
情報通信研究機構　知識創成コミュニケーション研究センター
{taro.watanabe,eiichiro.sumita}@nict.go.jp

1　はじめに

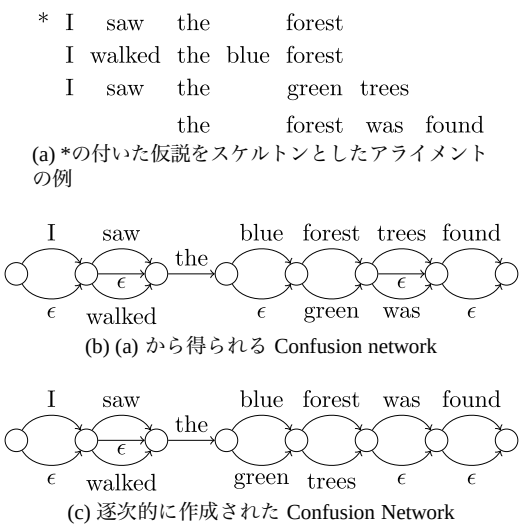
音声認識のローバー法 [6] などシステムコンビネーションでは、複数のシステム間の合意を求め、より良い解を得る。現在主流の機械翻訳システムコンビネーションでは、複数の出力を Confusion Network(CN) などのグラフ構造で表現する [1]。CN は単語単位の近さを基準に構築され、ある基準となる出力に対し、他の出力とのアライメントを計算、各エッジに単語のラベルを持つネットワークが形成される。新しい翻訳仮説は CN から最適経路を求めることにより得られる。

本稿では、単語の近さではなく、統語的な近さを用いた、新しいシステムコンビネーションの手法を提案する。複数の機械翻訳システムから CN を作成するのではなく、構文森 [2] を用いた、Confusion Forest(CF) を生成する。構文森は複数の構文解析木のノードを共有することにより圧縮したデータ構造であり、以下のように、文法に基づく手法を用いて生成する。まず、システムの出力を構文解析する。構文解析木を構成するルールを抽出する。このルールの集合は、原言語の入力文に対する目的言語の文法であり、この文法からアーリー法 [5] にて構文森を生成する。新しい翻訳は構文解析森から最適な導出木を求めることにより得られる。

統計的機械翻訳ワークショップ (WMT10) における、チェコ語およびドイツ語、スペイン語、フランス語の各言語から英語への機械翻訳システムコンビネーションタスク [3] において実験を行った。この結果、従来法に比べほぼ同様な翻訳を得ることができ、仮説を大幅にコンパクトな空間に表現できることを確認した。また、二つの言語対にて有意な差が見られた。

2　Confusion Network

CN に基づく機械翻訳システムコンビネーションは、まず、各出力間のアライメントを計算する [16]。次に、



(a) *の付いた仮説をスケルトンとしたアライメントの例

(b) (a) から得られる Confusion network

(c) 逐次的に作成された Confusion Network

図 1: Confusion Network の例

複数の出力の中から、例えば Minimum Bayes Risk などの基準を用いて、単語の並び替えの基準となるスケルトンを選択する。他の仮説は、スケルトンに対するアライメントに基づいて組み合わされ、ネットワークが形成される。図 1(a) はスケルトンに対し、複数のシステムの出力をアライメントした例を示している。図 1(a) から、各エッジに ϵ を含む、単語のラベルが付けられた図 1(b) のような CN が生成さる。このようなペア単位のアライメントでは、例えば、「green」が「forest」と対になるエラーのため、単語の繰り返しが多く見られる。逐次的な手法 [15] では、仮説単位のアライメントを計算するのではなく、逐次的に仮説が組み合わされる CN と、各出力とのアライメントを計算している。これにより、例えば図 1(c) のように、「green trees」と「blue forest」が対となり、ネットワークが形成される。

CN に基づく手法は、単語の並びを決定するスケルトンの選択に依存している。例えば、図 1(a) の例では、最初の三つの仮説が能動態であるのに対し、最後の仮説は受動態であり、文法的に大幅に異なる構造を持つ。この結果、図 1(c) のように極端に長い仮説を生

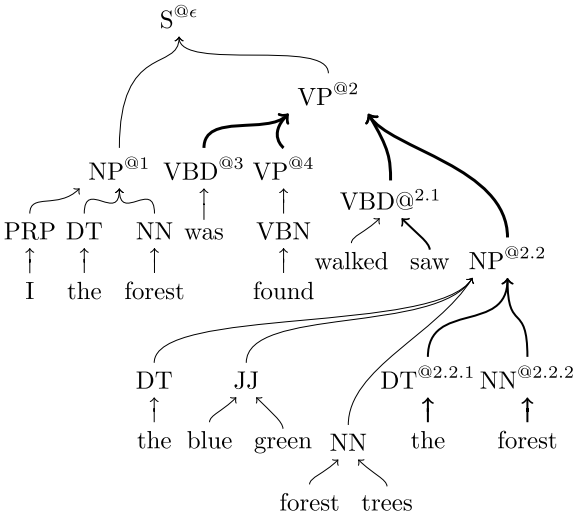
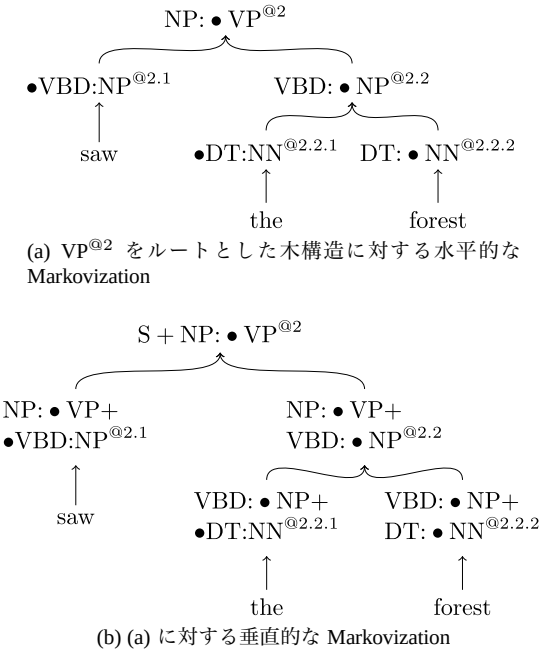


図 2: 図 1(a) を基に作成された Confusion Forest の例

成することが可能となる。これを部分的に解決する手法として、各出力それぞれがスケルトンとして複数のネットワークを作成し、そのネットワークを結合して大きなネットワークを作成する手法が提案されているが、根本的な解決とはなっていない。

3 Confusion Forest

複数の仮説を CN として表現するのに対し、本稿では、Confusion Forest(CF) という複数の構文木をコンパクトに表現した構文森として表現する。統語的な合意は構文解析木の木構造の断片を共有することにより実現する。構文森は構文解析や機械翻訳で使用されるハイパーグラフとして表現される [10]。具体的には、ハイパーグラフは $\langle V, E\rangle$ という二つ組で表現され、V、E はそれぞれノードの集合、ハイパーエッジの集合とする。V の各ノードは $X^{@p}$ として表現され、X $\in \mathcal{N}$ は非終端記号であり、p は各ノードの ID を親からの相対的な位置として表されるアドレスとする。例えば、ルートノードには、ϵ のアドレスが割り当てられ、p の最初の小ノードには、$p.1$ のアドレスが割り当てられる。各ハイパーエッジ $e \in E$ は文脈自由文法のルールのインスタンスであり、$\langle head(e), tails(e)\rangle$ という二つ組で表される。推論規則とみなした場合、$head(e) \in V$ は後件を表したノードであり、$tails(e) \in V^*$ 前件の集合である。図 1(a) の仮説に対する構文森の例を図 2 に示す。例えば、$\text{VP}^{@2}$ の二つのハイパーエッジ $\langle \text{VP}^{@2}, (\text{VBD}^{@3}, \text{VP}^{@4})\rangle$ と $\langle \text{VP}^{@2}, (\text{VBD}^{@2.1}, \text{NP}^{@2.2})\rangle$ により、前者は受動態、後者は能動態という文法的に異なる導出が可能である。



(a) $\text{VP}^{@2}$ をルートとした木構造に対する水平的な Markovization

(b) (a) に対する垂直的な Markovization

図 3: 非終端記号の書き換え例

機械翻訳システムの出力が与えられたとき、以下のような文法的な手法により CF を生成する。まず、機械翻訳システムの出力を構文解析する。次に、各構文解析木の各ハイパーエッジを文脈自由文法のルールのインスタンスとみなして文法を獲得する。入力文に特化した局所的な文法を基にして、開始記号から始め、各非終端記号を書き換えていくことにより構文森を生成する。新しい仮説は、生成された構文森から最適な導出木を計算することにより得られる。

3.1 ルールの獲得

ルール獲得時に、構文木の各ノードに割り当てられている非終端記号に元々の木構造の形を符号化することにより、ルールの曖昧性を減らす。まず、水平的な Markovization[11] により、各ノードの非終端記号にその左右の兄弟ノードの非終端記号を符号化する。図 3(a) では、$\text{VP}^{@2}$ をルートとした木構造に対するラベルの書き換え例を示す。例えば $\text{NP}^{@2.2}$ に対して、その左にある兄弟ノード $\text{VBD}^{@2.1}$ のラベルを組み合わせ、• で元々のノードのラベルの位置を示す。続いて、垂直的な Markovization[11] により、親ノードのラベルを組み合わせる。図 3(b) では、@2.2 のノードがその親である @2 のノードのラベルと組み合わされ、(NP: • VP + VBD: • NP) のようなラベルが得られる。このようにラベルの書き換えを行った後に、各ハイパーエッジをルールとみなして文法を学習する。

Initialization:

$$\frac{}{[\text{TOP} \rightarrow \bullet \text{S}, 0] : \overline{1}}$$

Scan:

$$\frac{[\text{X} \rightarrow \alpha \bullet x\beta, h] : u}{[\text{X} \rightarrow \alpha x \bullet \beta, h] : u}$$

Predict:

$$\frac{[\text{X} \rightarrow \alpha \bullet \text{Y}\beta, h]}{[\text{Y} \rightarrow \bullet\gamma, h+1] : u} \quad \text{Y} \xrightarrow{u} \gamma \in \mathcal{G}, h < H$$

Complete:

$$\frac{[\text{X} \rightarrow \alpha \bullet \text{Y}\beta, h] : u \quad [\text{Y} \rightarrow \gamma\bullet, h+1] : v}{[\text{X} \rightarrow \alpha\text{Y} \bullet \beta, h] : u \otimes v}$$

Goal:

$$[\text{TOP} \rightarrow \text{S}\bullet, 0]$$

図 4: 推論規則によるアーリー法生成アルゴリズム

3.2 構文森の生成

3.1 節で得られた文法に対して、アーリー法 [5] を適用し構文森を生成する。推論規則 [7] として表現された生成アルゴリズムを図 4 に示す。$\text{X} \in \mathcal{N}$ は非終端記号とし、$x \in \mathcal{T}$ を終端記号とする。α と β、γ は終端記号、非終端記号の記号列 $(\mathcal{T} \cup \mathcal{N})^*$ であり、u と v は、各項目に割り当てられる重みである。

一般的なアーリー法とは異なり、各非終端記号に割り当てられるスパンの情報は無視され、各導出に対する高さを h として保持する。Scan ステップは必ず成功し、このため、深い構文森が生成される。この深さは、Predict ステップにおける $h < H$ により制限される。本稿では、H は構文解析されたシステムの出力のうち最大の深さの 1.5 倍としている。

3.3 構文森のリスコア

構文森 F から、構文森に基づく k-best 構文解析アルゴリズム [8] を使用して、k-best の導出 d を求める。全ての可能な導出 D から最適な導出 $\hat{d}$ を求めるため、複数の素性を線形結合した目的関数を使用する。

$$\hat{d} = \arg\max_{d \in D} \mathbf{w}^\top \cdot \mathbf{h}(d, F) \tag{1}$$

ここで、$\mathbf{h}(d, F)$ は素性の集合であり、$\mathbf{w}$ により重み付けされる。Cube Pruning により、n グラム言語モデルなどの非局所的な素性との近似的な結合を行う [4, 9]。そして、k-best 導出には、Huang と Chiang のアルゴリズム 3 を用いる [8]。

表 1: WMT10 システムコンビネーションのデータ

	cz-en	de-en	es-en	fr-en
システム数	6	16	8	14
平均文字数 tune	10.6K	10.9K	10.9K	11.0K
test	50.5K	52.1K	52.1K	52.4K

4 実験

WMT10 における、チェコ語およびドイツ語、スペイン語、フランス語の各言語から英語への機械翻訳システムコンビネーションタスクにて実験を行った [3]。tune/test それぞれ 455/2,034 文からなるデータを表 1 に示す。各システムの出力は Stanford parser[11] にて構文解析し、全ての単語を小文字へと変換した。

本稿では、同期的文脈自由文法に基づく機械翻訳システム [4] のために作成された、ハイパーグラフに基づくツールキットを用いて実験した。各システムの出力は 3 節における文法に基づく手法により、ハイパーグラフで表された CF が生成される。また、ベースラインとして、2 節の CN に基づく手法を用いた。具体的には、TER[17] に基づいてネットワークに対するアライメントを逐次的に計算し、各システムの出力をスケルトンとして作成された複数のネットワークを結合し、大きなネットワークを作成する [15]。ネットワークは上記のツールキットを用い、フレーズに基づく推論規則 [9] を用いてハイパーグラフへと変換する。

本実験では以下の素性を用いた。$h^i_{lm}(d)$:English Gigaword[1]、10^9 コーパス、news commentary [2]の三つの 5-gram 言語モデル。$h_t(d)$:終端記号の数。$h_e(d)$:ハイパーエッジの数。$h^m_s(d)$:各システムの信頼度を表し、d の中で m 番目のシステムから得られたルールの数。$h^m_b(d)$:各システムの出力 $\mathbf{e}_m$ を参照訳として計算された d の BLEU 値 [13]。この素性は BLEU によるシステム間の相関を表すことができる [12]。$h_p(m)$: CN のみに用いられる素性であり、m 番目のシステムをスケルトンとして得られるネットワークの編集距離の合計をそのノード数で割った値 [14]。

異なる水平的 ($h = 1, 2, \infty$)、垂直的 ($v = 4, 5, \infty$) なオーダによる CF と、CN、システム出力の最大最小の BLEU 値を表 2 に示す[3]。最大の BLEU と統計的に優位な差が見られなかった結果を太字で示す。CF は $v = \infty$、$h = \infty$ としたとき、CN とほぼ同様な結果が得られた。これは、各システムの出力のルートからの

[1] LDC2009T13

[2] http://www.statmt.org/wmt10/

[3] 例えば、$h = 1$ の場合、左右のコンテキストを見て、最大三つのラベルが結合される。

表 2: BLEU による翻訳結果

言語対	cz-en	de-en	es-en	fr-en
システム 最小	14.09	15.62	21.79	16.79
最大	23.44	24.10	29.97	**29.17**
CN	23.70	24.09	**30.45**	**29.15**
$CF_{v=\infty,h=\infty}$	**24.13**	24.18	**30.41**	**29.57**
$CF_{v=\infty,h=2}$	**24.14**	**24.58**	**30.52**	28.84
$CF_{v=\infty,h=1}$	**24.01**	23.91	**30.46**	**29.32**
$CF_{v=5,h=\infty}$	**23.93**	23.57	29.88	28.71
$CF_{v=5,h=2}$	**23.82**	22.68	29.92	28.83
$CF_{v=5,h=1}$	**23.77**	21.42	30.10	28.32
$CF_{v=4,h=\infty}$	23.38	23.34	29.81	27.34
$CF_{v=4,h=2}$	23.30	23.95	30.02	28.19
$CF_{v=4,h=1}$	23.23	21.43	29.27	26.53

表 3: ハイパーエッジの平均数 ($h = 1$)

言語対	cz-en	de-en	es-en	fr-en
CN	2,222.68	47,231.20	2,932.24	11,969.40
lattice	1,723.91	41,403.90	2,330.04	10,119.10
$CF_{v=\infty}$	230.08	540.03	262.30	386.79
$CF_{v=5}$	254.45	651.10	302.01	477.51
$CF_{v=4}$	286.01	802.79	349.21	575.17

導出を記憶し、その並びを保ったまま、木構造の断片を組み合わせることに相当する。$v = \infty$、$h = 2$ の時、CF は木単位の多少の並び替えが行われており、三つの言語対において最も良い結果がえられ、また cz、de では、CN と比較して統計的に有意な差が見られた。また、オーダを小さくすると、大幅な終端記号の並び替えが行われ、BLEU が小さくなることが確認された。これは、木構造を捉えた素性を導入することにより解決できると考えられ、今後の課題としたい。

表 3 にて平均ハイパーエッジ数で示されるハイパーグラフの大きさを示す。このように、CF は CN と比べ、桁違いに小さいことがわかる。

5 おわりに

本稿は、CF に基づく機械翻訳システムコンビネーションを提案した。単語の近さに基づいて複数の出力を CN で表現するのに対し、統語的な近さを利用して構文森である CF で表現した。構文森は、システムの出力の構文解析木から局所的な文法を獲得することにより、生成される。実験結果から非常にコンパクトなデータ構造で CN とほぼ同様な結果を得られた。

参考文献

[1] Srinivas Bangalore, German Bordel, and Giuseppe Riccardi. Computing consensus translation from multiple machine translation systems. In *Proc. of ASRU*, pp. 351 – 354, 2001.

[2] Sylvie Billott and Bernard Lang. The structure of shared forests in ambiguous parsing. In *Proc. of ACL*, pp. 143–151, June 1989.

[3] Chris Callison-Burch, Philipp Koehn, Christof Monz, Kay Peterson, Mark Przybocki, and Omar Zaidan. Findings of the 2010 joint workshop on statistical machine translation and metrics for machine translation. In *Proc. of WMT*, pp. 17–53, July 2010.

[4] David Chiang. Hierarchical phrase-based translation. *Computational Linguistics*, Vol. 33, No. 2, pp. 201–228, 2007.

[5] Jay Earley. An efficient context-free parsing algorithm. *Communications of the Association for Computing Machinery*, Vol. 13, pp. 94–102, February 1970.

[6] J.G. Fiscus. A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (rover). In *Proc. of ASRU*, pp. 347–354, December 1997.

[7] Joshua Goodman. Semiring parsing. *Computational Linguistics*, Vol. 25, pp. 573–605, December 1999.

[8] Liang Huang and David Chiang. Better k-best parsing. In *Proc. of IWPT*, pp. 53–64, October 2005.

[9] Liang Huang and David Chiang. Forest rescoring: Faster decoding with integrated language models. In *Proc. of ACL*, pp. 144–151, June 2007.

[10] Dan Klein and Christopher D. Manning. Parsing and hypergraphs. In *Proc. of IWPT*, pp. 123–134, 2001.

[11] Dan Klein and Christopher D. Manning. Accurate unlexicalized parsing. In *Proc. of ACL*, pp. 423–430, July 2003.

[12] Wolfgang Macherey and Franz J. Och. An empirical study on computing consensus translations from multiple machine translation systems. In *Proc. of EMNLP-CoNLL*, pp. 986–995, June 2007.

[13] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proc. of ACL*, pp. 311–318, July 2002.

[14] Antti-Veikko Rosti, Spyros Matsoukas, and Richard Schwartz. Improved word-level system combination for machine translation. In *Proc. of ACL*, pp. 312–319, June 2007.

[15] Antti-Veikko Rosti, Bing Zhang, Spyros Matsoukas, and Richard Schwartz. Incremental hypothesis alignment for building confusion networks with application to machine translation system combination. In *Proc. of WMT*, pp. 183–186, June 2008.

[16] K.C. Sim, W.J. Byrne, M.J.F. Gales, H. Sahbi, and P.C. Woodland. Consensus network decoding for statistical machine translation system combination. In *Proc. of ICASSP*, Vol. 4, pp. IV–105 –IV–108, April 2007.

[17] Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. A study of translation edit rate with targeted human annotation. In *Proc. of AMTA*, pp. 223–231, 2006.