語彙獲得のための過分割未知語の検出

2009年3月3日

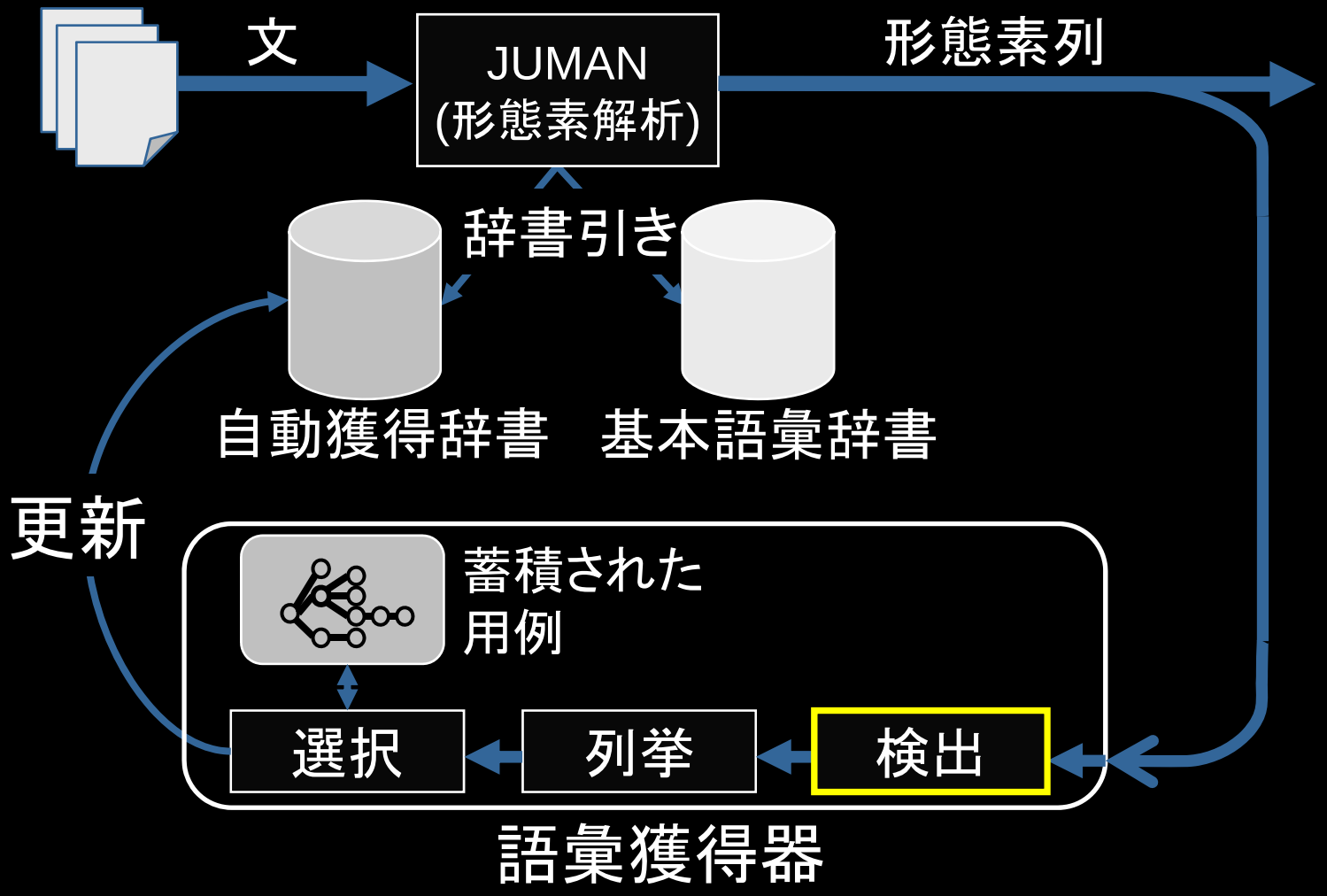
京都大学大学院 情報学研究科

村脇 有吾　　黒橋 禎夫

背景

- 形態素解析はNLPの最初の処理
  - ほぼ完璧な精度が求められる
- 辞書引きによる候補列挙で高い精度
  - 辞書にない未知語が問題
- テキストから未知語を獲得して自動で辞書に追加
  - 実用を想定

オンライン未知語獲得



[Murawaki and Kurohashi 2008]

概要

- 目的: テキスト中の未知語を検出
- 問題: 既知語と解釈されて検出できない
  - うざい ⇒ う (卯/雨/鵜) + ざい (剤/在/材/罪/財)
- 解決策: 表記ゆれ情報の利用
  - う + ざい ⇔ 卯 + ざい, 卯 + 剤, 雨 + ざい, etc?

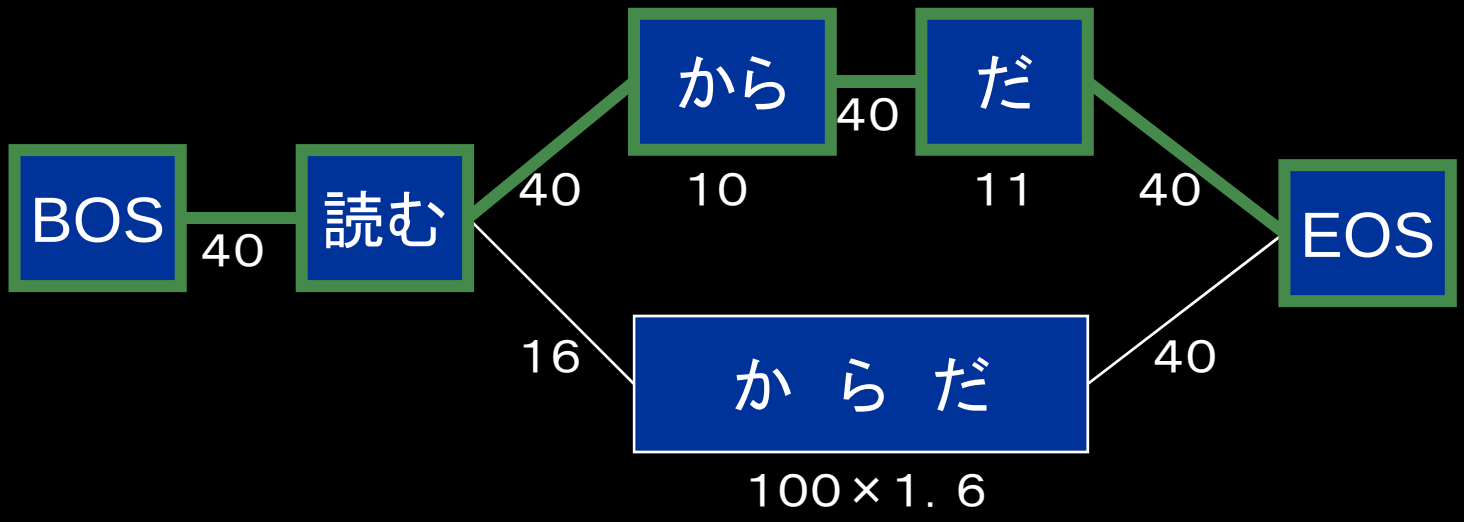
Outline

- 形態素解析の未知語処理
- 未知語検出のベースライン手法
- 過分割問題
- 表記ゆれを利用する提案手法
- 実験

Outline

- 形態素解析の未知語処理
- 未知語検出のベースライン手法
- 過分割問題
- 表記ゆれを利用する提案手法
- 実験

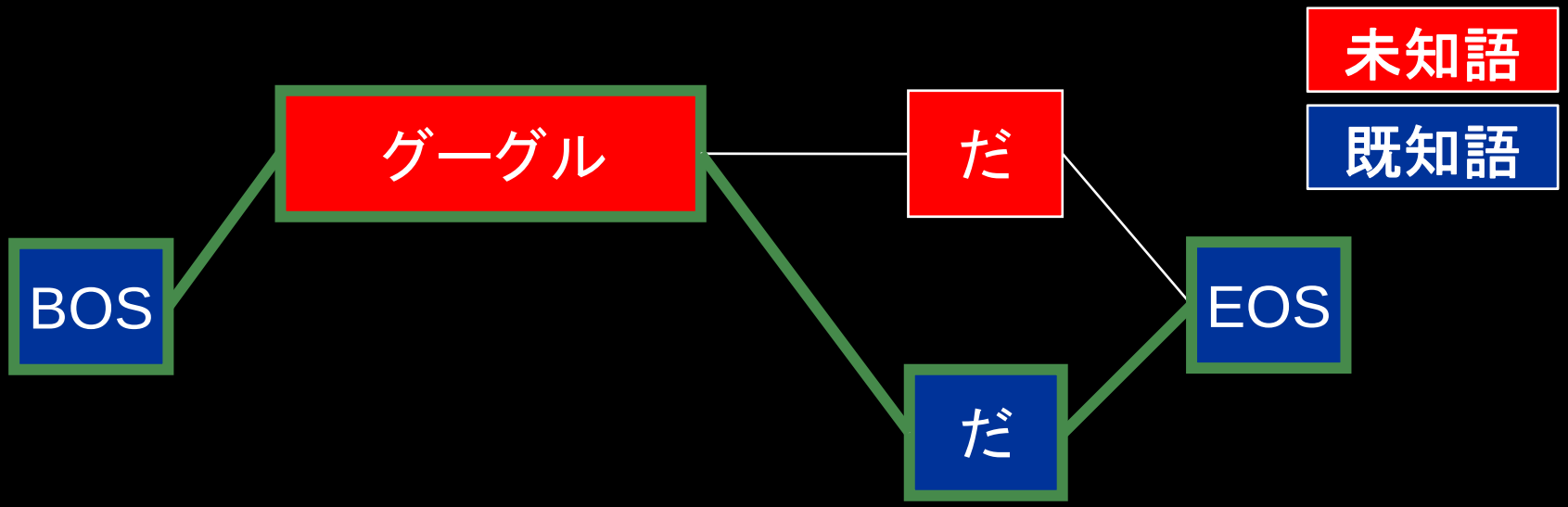
辞書引きによる候補列挙



1. 予め定義された辞書を引いて候補を列挙
2. コストにより最適なパスを選択

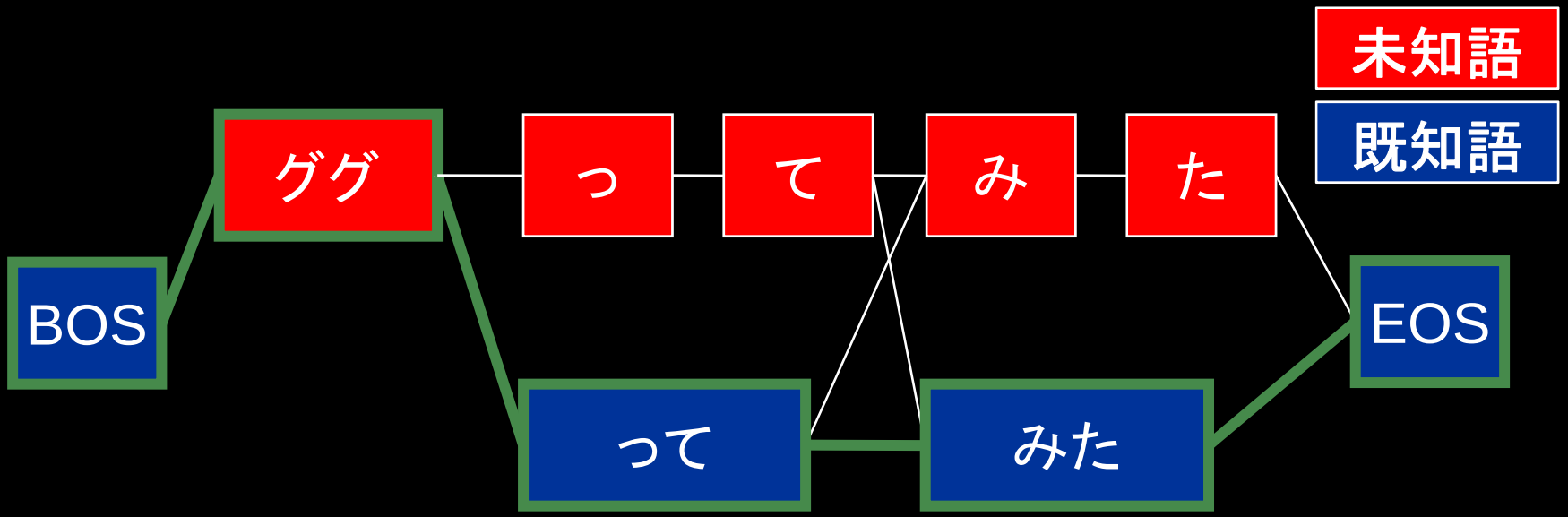
[Kurohashi et al. 1994] [Asahara and Matsumoto 2000] [Kudo et al. 2004]

形態素解析の未知語処理 1/3



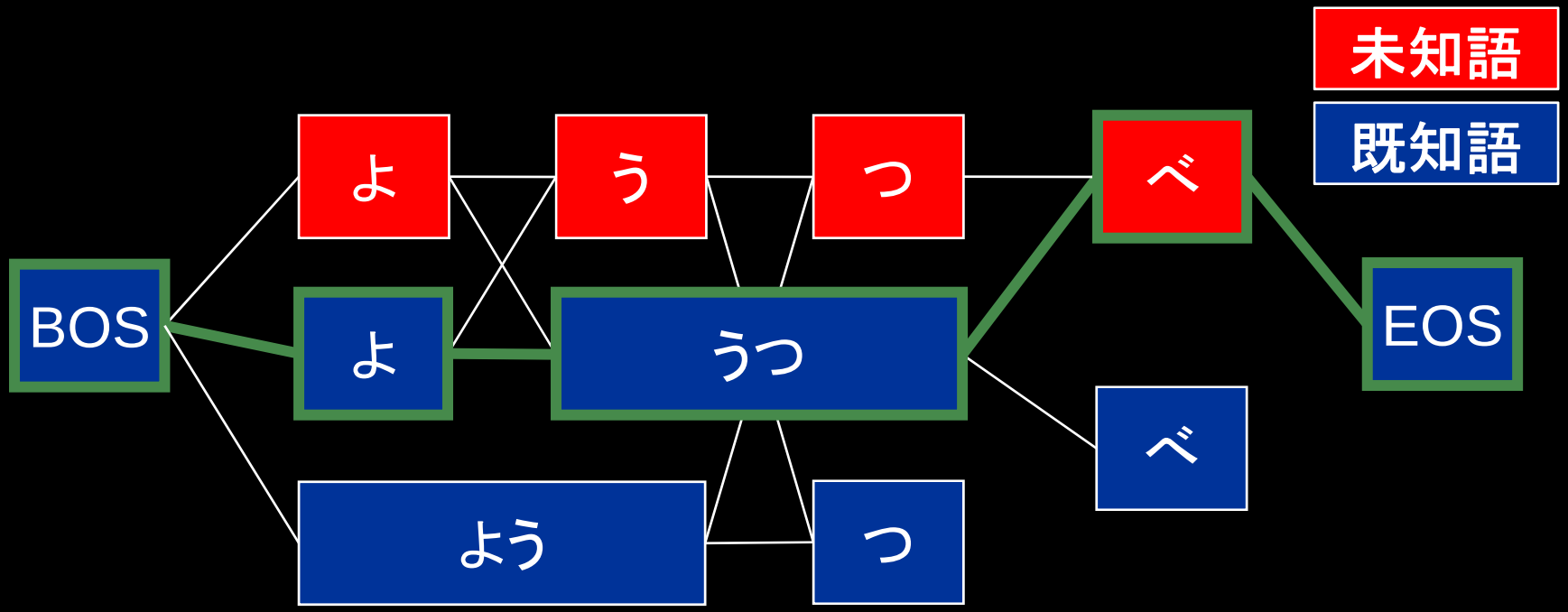
- 字種に基づくヒューリスティクスによる候補列挙
  - カタカナ連続を1候補に
  - ひらがな・漢字は一文字ずつ候補に (JUMANの場合)

形態素解析の未知語処理 2/3



- 問題点 1: 動詞の解析を誤る

形態素解析の未知語処理 3/3



- 問題点 2: 長いひらがな未知語を認識できない
  - e.g. ようつべ (YouTube)

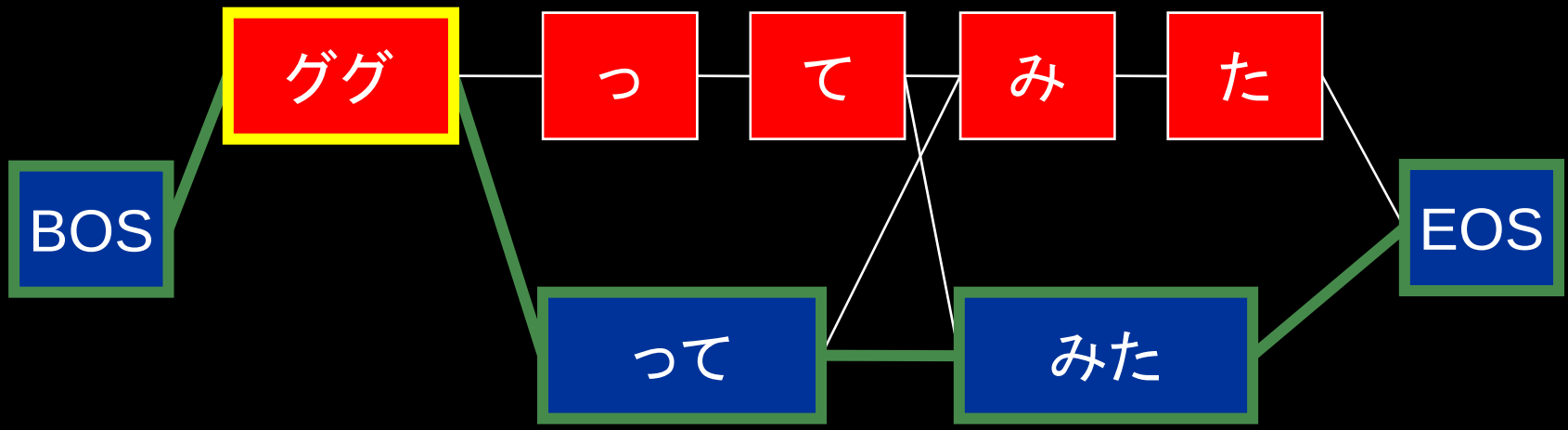
Outline

- 形態素解析の未知語処理
- 未知語検出のベースライン手法
- 過分割問題
- 表記ゆれを利用する提案手法
- 実験

未知語用例の検出タスク

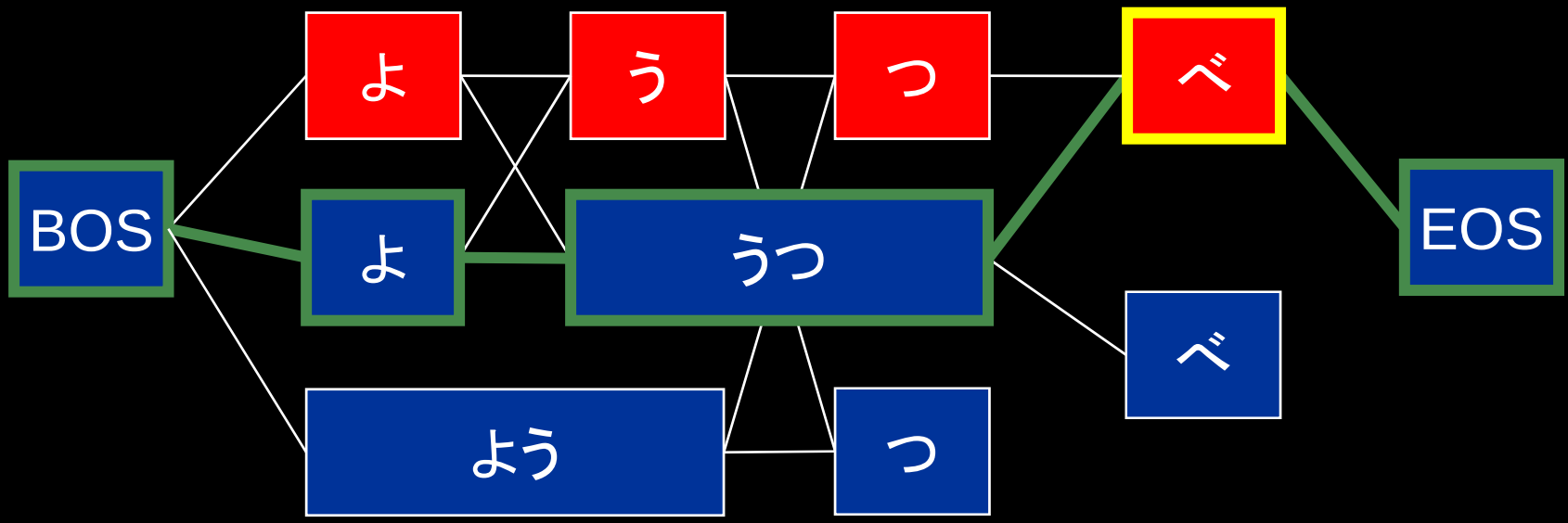
- テキスト中の未知語を見つける
  - 正確な境界同定は不要
- 未定義語に着目するベースライン手法
  1. 文を形態素解析結果
  2. 未定義語が含まれる箇所を検出

未定義語に着目する ベースライン手法 1/2



- 解析器が同定した未定義語に着目
  - 多くの場合、未知語 (の一部) を未定義語とする

未定義語に着目する ベースライン手法 2/2



- ようつべ (YouTube) も検出できる

Outline

- 形態素解析の未知語処理
- 未知語検出のベースライン手法
- 過分割問題
- 表記ゆれを利用する提案手法
- 実験

過分割問題

- 解析器が未知語を
  め、検出できない

人間が見ると2形態素の連続 (bigram) だけで、明らかに怪しい

うざい ⇒ う (卯/雨/鵜) + ざい (剤/在/材/罪/財)

うざくて ⇒ う + ざ (座) + く (区

めんどかった ⇒ めん (免/綿/

統語的にも自然で検出が難しそう

+ かった (買った/飼った/勝った/刈った/狩った/駆った)

かぐや姫 ⇒ かぐ (家具) + や + 姫

N-gram のシステマティックな誤り

P(ざい |う) は大きな値になってしまう

- テキストの形態素解析結果から形態素 bigramをカウント
- 解析誤りをシステマティックに蓄積
  - 「うざい」は常に「う + ざい」

Outline

- 形態素解析の未知語処理
- 未知語検出のベースライン手法
- 過分割問題
- 表記ゆれを利用する提案手法
- 実験

表記ゆれの利用: アイデア

- 知識: システマティックな誤りを含む N-gram
- 仮説: 形態素の異表記は同じように振る舞う
- 過分割候補に対して異表記の出現をチェック
  - う (卯/雨/鵜) + ざい (剤/在/材/罪/財)?
  - 卯 + ざい，卯 + 剤，雨 + ざい，etc?
  - 異表記が出現しないからこの分割は怪しい

表記ゆれの利用: 定式化

前向きbigramのチェック

- 「う + ざい」の「う」と「ざい」の連鎖について
  - 形態素解析結果の bigram から尤度比を計算

$$L_f = \log \frac{\mathrm{P}(\{\text{ざい}, \text{剤}\} \mid \text{う})}{\mathrm{P}(\{\text{ざい}, \text{剤}\} \mid \{\text{烏}, \text{雨}, \text{鵜}\})}$$

  - L_f が閾値以上であれば検出

表記ゆれの利用: 表記ゆれ知識

- JUMANの代表表記を利用
  - JUMAN 5.0 以降で基本語彙に付与
  - 表記ゆれを吸収するキー
  - 曖昧性解消の候補
- 卯/う = {う, 卯};　雨/う = {う, 雨}
- う ∈ 卯/う;　う ∈ 雨/う;　う ∈ 鵜/う

未知語検出の実行

解析結果の形態素列を前からスキャン

1. ルールで検出 (ベースライン手法)
   - 未定義語
   - 1文字漢字の3連続, etc
2. 検査の対象表記について
   1. 前向きbigramの検査
   2. 後ろ向きbigramの検査

Outline

- 形態素解析の未知語処理
- 未知語検出のベースライン手法
- 過分割問題
- 表記ゆれを利用する提案手法
- 実験

実験

1. 未知語検出の評価
2. オンライン未知語獲得による獲得未知語の評価
3. 獲得未知語による形態素解析の改善の評価

検出実験

- 適合率: システム出力をサンプリングして人手で評価
- 再現率: 近似的に正解データを作成
  1. 過分割候補のみを簡単なルールで抽出
  2. 人手でアノテーション
- ベースライン手法: ルール
- 提案手法: ルール + 表記ゆれチェック

検出実験: 結果

		ベースライン	提案手法 (smoothing なし)	提案手法 (smoothing あり)
再現率		288/1004 (28.7%)	**688**/1004 (**68.5**%)	648/1004 (64.5%)
適合率	全体	12,616	14,256	13,957
	カタカナ除く	1,478	3,204	2,884
	精度	**173**/200 (**86.5**%)	140/200 (70.0%)	156/200 (78.0%)

検出実験: 新たな検出例

- かもめ ⇒ かも (鴨) + め (目)
- ちゃち-い ⇒ ちゃ (茶) + ちい (地位/地異)
- よさこい ⇒ よ- (良/善) + さ + こい (恋/故意/鯉)

検出実験: 検出されない未知語

1. そもそも対象表記でない
   - あずみ ⇒ あ (感動詞) + ずみ (接尾辞)
2. bigram では自然な連接
   - 助詞による誤分割が多い
     - めも ⇒ め (目) + も
     - でか-い ⇒ で (出) + かい (終助詞)

検出実験: 文脈に依存する検出

- 前後の文字列によって検出が依存
  - はてな ⇒ はて (果て) + な (終助詞)
  - はてなが ⇒ はて (感動詞) + な (名/菜) + が
  - はてなを ⇒ はて (感動詞) + な (名/菜) + を

獲得実験: 結果

クエリ	ベースライン手法		提案手法	
捕鯨問題	99.5%	(190/191)	98.5%	(194/197)
赤ちゃんポスト	100%	(78/78)	100%	(80/80)
JASRAC	98.1%	(473/482)	97.8%	(499/510)

- 獲得未知語数がやや増加
  - 数ではカタカナ名詞が圧倒的
  - ひらがな未知語の方が形態素解析の改善に重要

獲得実験: 獲得できた正解例

ベースラインから新たに獲得された例

- はてな
- わんこ
- かがみん (人名)
- ドラえもん
- ねとらじ
- めんどくさ-い:イ形容詞

課題

- bigramでは検出されない未知語
- カタカナ未知語の過分割問題
  - モニタリング ⇒ モニタ + リング
- 漢語未知語の過分割問題
  - 主に2文字漢語
- 成果の還元

未知語による誤りの連鎖

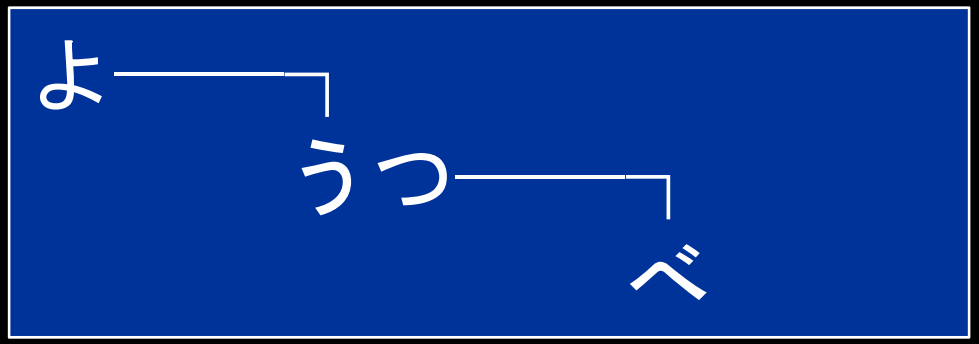
入力文

ようつべ

形態素解析

よ（夜）+うつ（打つ）+べ

構文解析



格解析

打つ（べ:ガ，よ:時間）

その他の過分割問題

- カタカナ外来語
  - カースト ⇒ カー + スト
  - サーバー ⇒ サー + バー
  - モニタリング ⇒ モニタ + リング
- 元言語の知識が有効?

定式化 1/2

- 形態素列 $\ldots w_{i-1}, w_i, w_{i+1}, \ldots$ について

  前向きbigram w_i, r_{i+1}

  L_f が閾値以上なら w_i を検出

  V_{wi} は w_i の異表記の集合

  $V_{う} = \{$卯, 雨, 鵜$\}$

  r_{i+1} は w_{i+1} の代表表記

  ざい → 剤/ざい, 在/ざい, etc

定式化 2/2

- 形態素列 $\ldots w_{i-1}, w_i, w_{i+1}, \ldots$ について
- 後ろbigram r_{i-1}, w_i もチェック

確率は最尤推定

- 形態素解析結果から頻度をカウント

$$P(v_{i+1} \mid v_i) = \frac{f(v_i v_{i+1})}{f(v_i)}$$

$$P(v_{i+1} \mid v_i) = \frac{\sum f(v_i v_{i+1})}{\sum f(v_i)}$$

smoothing

- 文節をまたぐ場合に smoothing
  - 日本語の語順は柔軟で、文節境界をまたぐ連接は自由度が高い

対象表記の選定

- 表記ゆれチェックの対象表記 w_i
  - 現在はひらがなと混ぜ書きのみ
- 対象表記から異表記へのマッピング
  - う → {卯, 雨, 鵜}
- JUMANの辞書からマッピングを構築
  - ルール + 人手で修正
  - 8,509個の対象表記 (原形のみカウント)

検出実験: N-gram 構築

- ウェブコーパス約1億ページから構築
- 頻度10で足切り

検出実験: 正解データの作成 1/2

- 京都テキストコーパスは使えない
  - JUMANの開発に使われてきたから未知語が極端に少ない
- コーパス全体へのタグ付けは非現実的
  - 未知語は出現頻度が低いので大規模なデータへのタグ付けが必要
- 過分割の可能性のある部分に絞り込んでタグ付け

検出実験: 正解データ作成 2/2

- 形態素解析結果で、以下のルールにマッチする箇所を抽出
  - ひらがな1文字 + ひらがな1文字
  - ひらがな2文字 + ひらがな1文字
  - ひらがな1文字 + ひらがな2文字
- フィルタリング
  - 京都コーパスから同じルールでひらがな連続を抽出しておき、それらとマッチするものを除外
- 人手で抽出箇所の未知語語幹にタグを付与

獲得実験: 設定

- テキストから未知語を獲得
  - TSUBAKIの検索結果上位1,000ページ
- 獲得された未知語の正誤を人手で評価
  - 語幹と品詞の両方が正しければ正解