頻出文脈に基づく分野依存入力支援

日本アイ・ビー・エム株式会社
〇海野 裕也，坪井 祐太
{yunno,yutat}@jp.ibm.com

背景:ビジネス文書などで表現の統一を行いたい

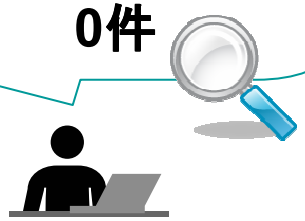
文書中で適切な表現が行われていないと不都合が生じる

例

検索が難しくなる

人事管理システムを開いて下さい。

人事システム:
0件



読み手に誤解を生じさせる

ここで、**システム出力**を確認して下さい。

システム出力って何？
ファイル出力？
画面出力？

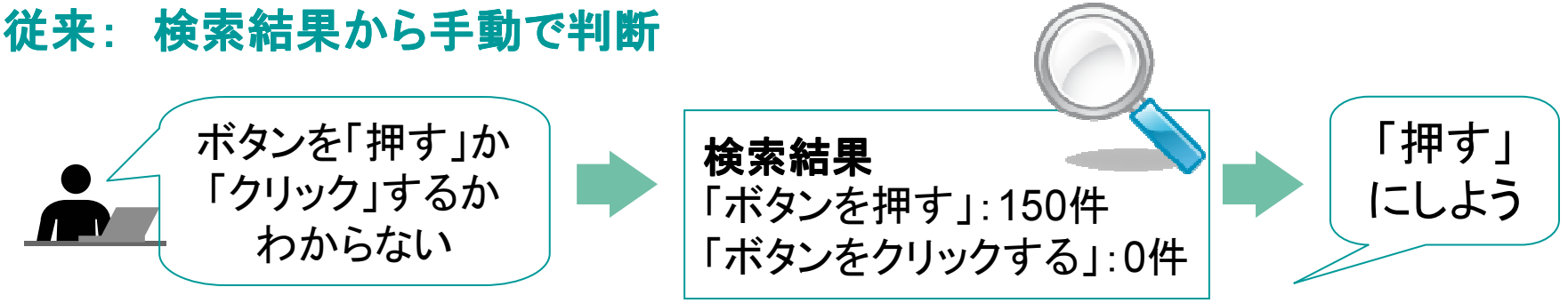
非母語話者にとって適切な表現は困難

これを**使用ない**で下さい。

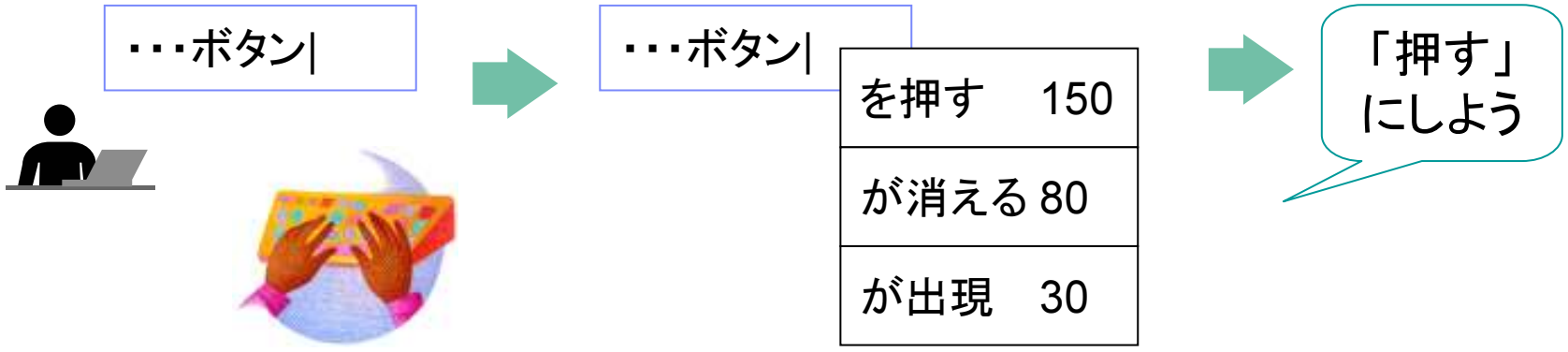
「使用しない」の間違い？

既存文書中での頻度に基づいて利用する表現を選択する

従来： 検索結果から手動で判断



目標： 高頻度表現を自動で提示



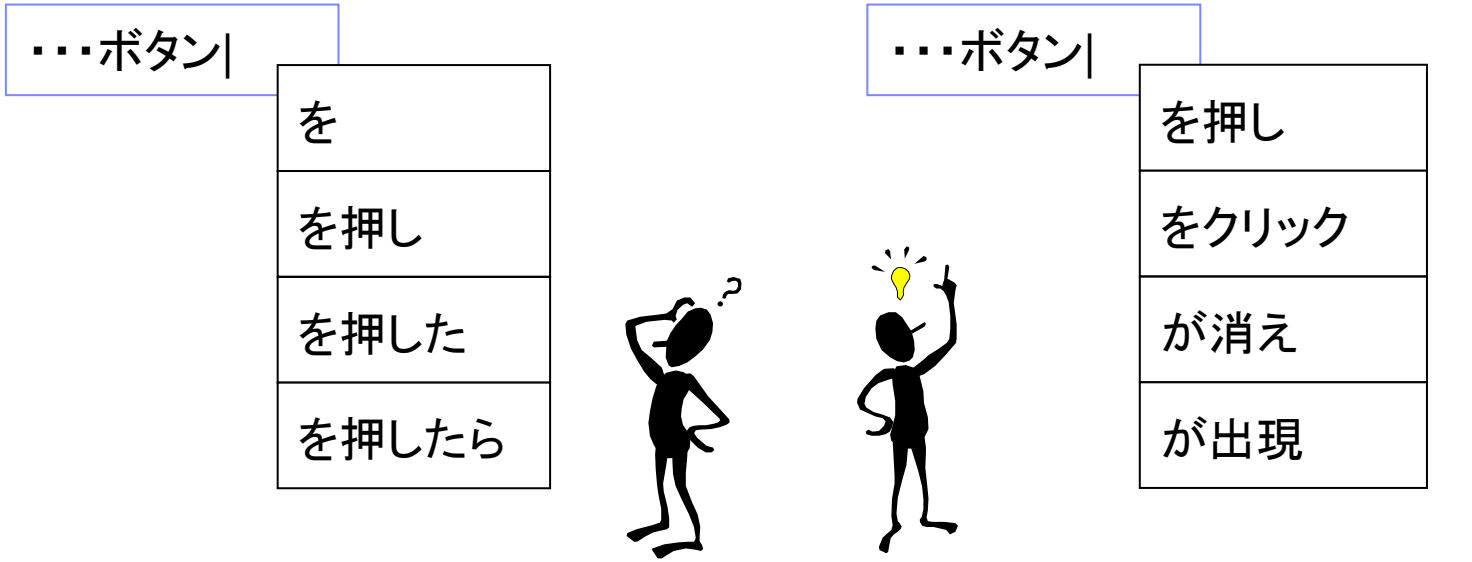
目的は異なるが予測入力技術と手法が類似している

- 本研究でやりたいこと
  - –表現の統一を図りたい
  - –既存文書と同一の表現を利用して欲しい
  - –そのため**既存文書中での高頻度表現を提示する**のがよい

- 予測入力でやりたいこと
  - –入力の手間を省きたい
  - –よく入力される候補を提示したい
  - –そのため**既存文書中での高頻度表現を提示する**のがよい

貪欲に高頻度表現の上位を提示すると類似候補が複数選択されやすい

- 用語統一のためには**候補を概観**できないと都合が悪い
- 長い候補を提示しなくても、短い候補を選択すれば続きを見ることはできる



なるべく多様な表現が選ばれる方がよい

候補が重複しないようにKWICを圧縮して表示する [海野+10] (1/2)

- 表示する文字列でカバーされる面積を求める
- K個の文字列で**カバーされる面積の合計を最大化**させる

「ボタン」の後続文字列

全文脈(KWIC)

ボタンが大きくて・・・
ボタンが赤い．・・・
ボタンという表・・・
ボタンに書いてあ・・・
ボタンをクリックしたら・・・
ボタンをクリックして下・・・
ボタンをクリックしよう・・・
ボタンをクリックできな・・・
ボタンをクリックできま・・・
ボタンをクリック．・・・
ボタンを押したら・・
ボタンを押しては・・・
ボタンを押せませ・・・
ボタンを押そうと・・・

「を」のカバー範囲
「を押」のカバー範囲
「を押し」のカバー範囲

長さ　頻度

$$S^* = \arg\max_S \sum_{s \in S} \mathrm{Len}(s) \times \mathrm{Pref}(s, C)$$

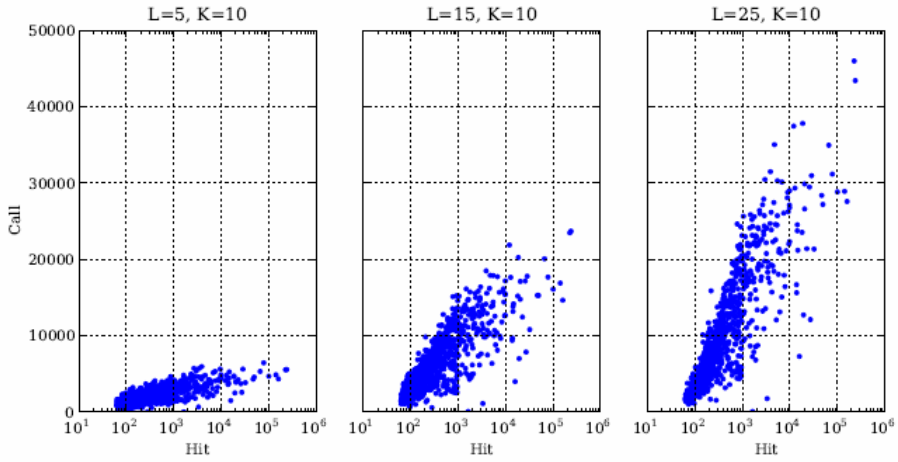
S: 文字列集合 （ただし、各要素は他の要素の接頭辞にならない）
C: 文脈文字列集合
Len(s): sの文字列長
Pref(s, C): sを接頭辞とするC中の要素数

提案手法(K=3)

ボタンをクリックし
ボタンをクリックでき
ボタンを押

候補が重複しないようにKWICを圧縮して表示する [海野+10] (2/2)

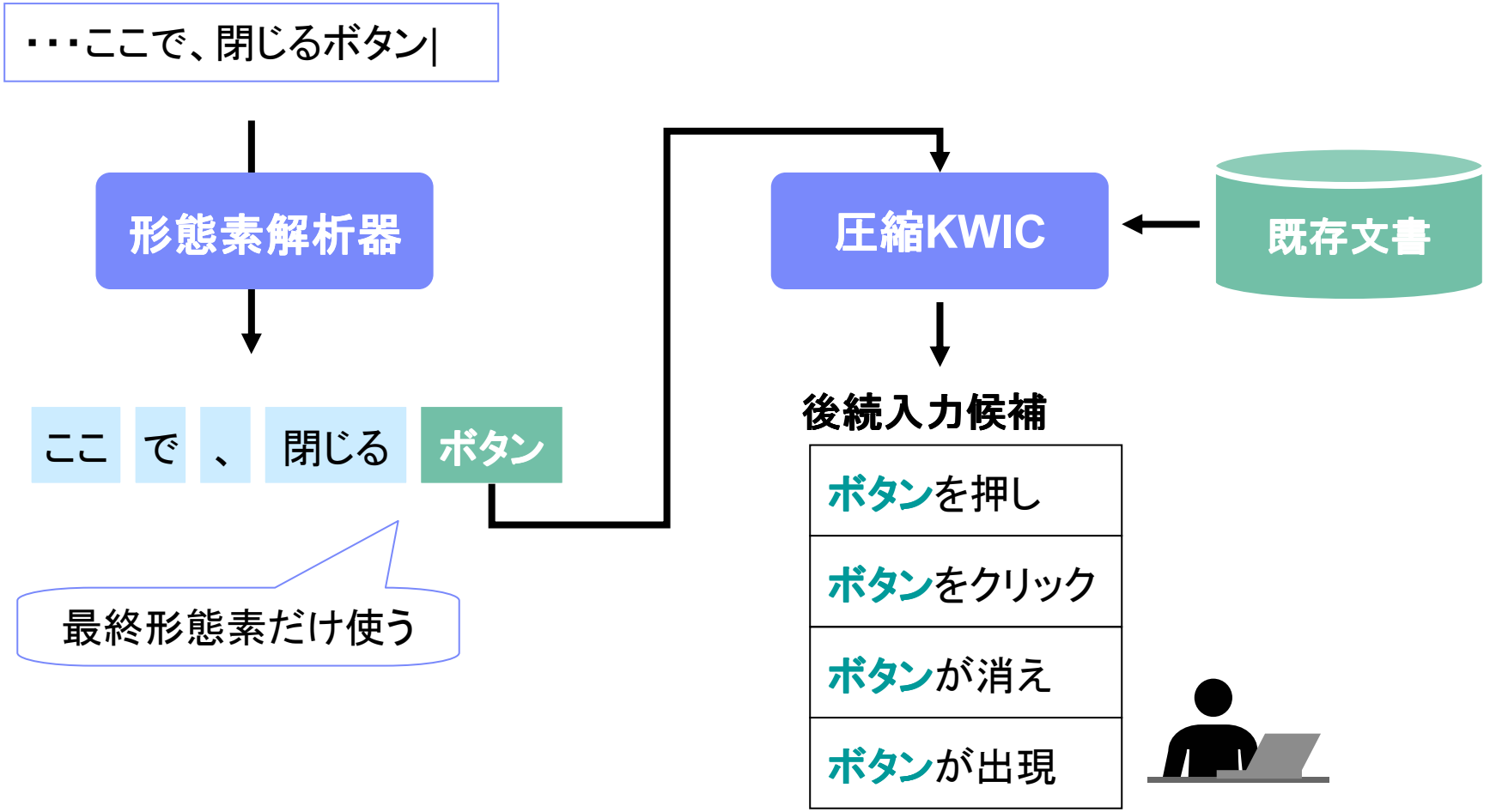
- 後方文脈集合に対する**TRIE中のノードを選択する**問題とみなせる
- 動的計画法を使うと**検索ヒット数に比例した時間**で探索できる
- 最大値の上限を見積もる**枝刈り**で、**実験的に検索ヒット数の対数時間**程度で済む
- 子ノードを頻度順に並べた**接尾辞木**を作っておくとTRIEの構築時間も省ける



横軸：ヒット数の対数　　縦軸：実行時間

詳細は論文参照

圧縮KWICを入力支援に応用する



従来の予測入力手法との違い

2つの点で大きく異なる

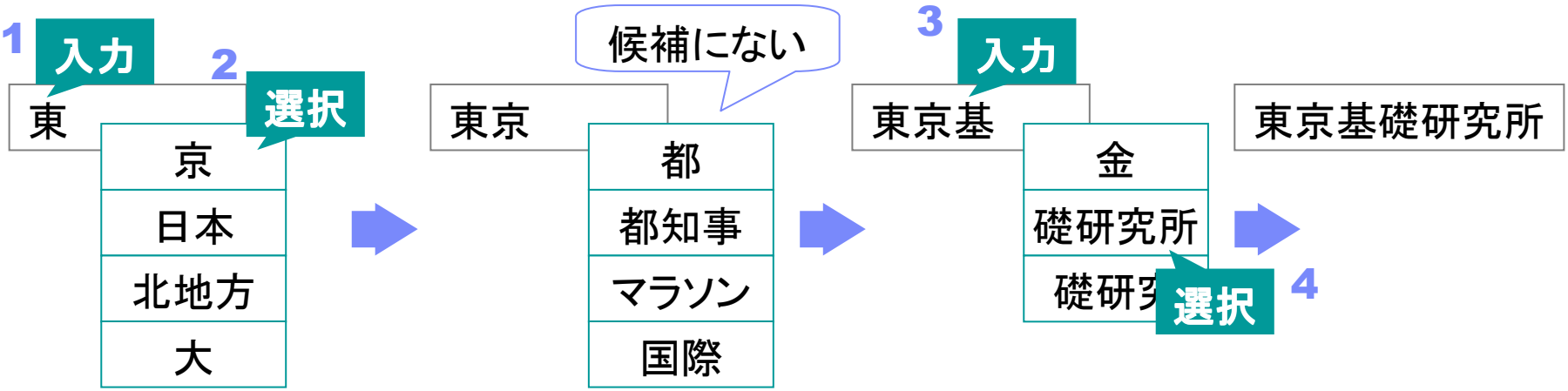
- **候補の組み合わせを探す**問題に定式化している
  - –従来の手法はいずれもスコアの高い上位K件を貪欲に提示する
- **全部分文字列**が提示候補である
  - –記号列のようなデータに対しても処理できる

表 1: 各手法の特徴の比較.

	手法	候補選定	スコア *	複数候補
	提案	全部分文字列	LF	総和最大化
従来の予測入力手法	Masui[2]	単語単位	F	貪欲
	市村 [4]	辞書	$cF + L$	貪欲
	奥野 [6]	頻出単語列	$(L - c)F$	貪欲
	山本 [3]	後続文字種数	$\log(L + 1)F$	貪欲

評価方法：実データの入力シミュレーションによる打鍵数の比較

入力対象：「東京基礎研究所」



7文字に対して合計4回入力
(7 - 4)÷7 = 43%の削減率

- 最初の1文字を入力する
- 入力候補に入力対象があれば必ず選択する（途中文字列までも入力可能とする）
- 無いときは直接入力する
- 「入力」と「選択」回数を入力回数とし、削減率を計算する

打鍵数削減率の比較結果

頻度計測用　入力対象

参照	入力	提案	kiwi	freq
san	san	24.7 %	20.9 %	24.4 %
twi	twi	25.0 %	20.9 %	23.4 %
san	twi	8.8 %	7.4 %	8.5 %
twi	san	13.5 %	9.6 %	13.6 %

- 提案手法は他手法より若干削減率が優れるが、差は小さい
- 同一分野を参照した方が倍近く削減でき、分野固有の表現を提示できていることを示唆している

- データ
  - san: 産経新聞1年分 （100MB）
  - twi: twitterクロールデータ （53MB）
- 比較手法
  - kiwi:後続文字種数が増える部分で単語分割, log(L + 1) x F の上位10件 [山本+03]
  - freq: 単語1~4gramの内, L x F の上位10件 [奥野+09]

既存手法では類似候補に偏りやすい傾向が確認される

- 産経新聞データで「安全」を入力
- 既存手法では、「安全保障」に候補が偏りすぎて全体の傾向がつかめない

提案		Kiwi		Freq	
安全・保安委	80	安全・保安院	80	安全確保	151
安全を確保する	46	安全を確保する	46	安全管理	104
安全委員会	41	安全確保	151	安全性を	120
安全確保	151	安全確保のため	30	安全対策	136
安全管理	104	安全性を	120	安全保障	1309
安全基準	61	安全対策	136	安全保障の	112
安全性に	73	安全保障	1309	安全保障会議	111
安全性を	120	安全保障上の	60	安全保障問題	94
安全対策	136	安全保障上級代表	22	安全保障理事	101
安全保障	1309	安全保障理事会	100	安全保障理事会	100

Twitterで不自然言語を入力支援

- Twitter データに対して「 (」を入力
- 分割単位の不明な記号列でも意味のありそうな結果がでている

提案		Kiwi		Freq	
(´ ▽ `)	809	(´ ▽ `)	809	(*´	3214
(*´	3214	(´ ▽ `)ノ	710	(^^	2810
(^^	2810	(*´∀`*)	573	(^o	3117
(^−^)	1605	(^−^)	1605	(^o^	3114
(^_^	1534	(^o^)	2951	(^0	2703
(^o^)	2951	(^o^)／	1853	(^0^	2698
(^0^)	2690	(^0^)	2690	(´・ω	1990
(´・ω・	1961	(^0^)／	1329	(>_	2386
(>_<)	2306	(´・ω・`)	1290	(>_<	2384
(笑)	5841	(>_<)	2306	(笑)	5841

重複のある候補　既存のいずれかで提示されない

考察：個別に見るとよさそうだが、数字に表れていない理由は？

- 低頻度の語を予測できても数字には表れない
  - –頻度が低い語はテストデータにもほとんど現れない
  - –ほとんど現れない語を予測してもスコアはよくならない
- では、低頻度語を出す必要はない？
  - –NO
  - –候補がひとつだけだと他の候補と比較できない
- 表現の統一に貢献しているか測るには別の指標が必要かもしれない

今後の課題： 実際に使ってもらうために

- IMEとつなげたい
  - –IMEを一から作るのは大変
  - –予測入力部分の切り分け？
  - –プロトコルの整備？

まとめ

- 表現統一のために既存文書中の**高頻度表現を提示**する入力支援システムを提案した
- 既存文書中に対する検索結果の**後方文脈を圧縮**して入力候補として提示する
- 従来の予測入力技術と比べて以下の2点で大きく異なる
  - –重複しない**最適な候補の組み合わせを探索**する
  - –事前の**単語単位に依存しない**候補を提示できる
- 入力シミュレーションによる比較では大きな差が得られなかった
- 個別の事例では**多様な候補**を提示していることを確認した