2017年4月24日
みちのく情報伝達学セミナー

品質推定と疑似パラレルコーパス

首都大学東京　小町研
D3　梶原智之

https://sites.google.com/site/moguranosenshi/

梶原 智之 @moguranosenshi

- 新居浜工業高等専門学校（高校1年〜学部2年）
  - 電気情報工学科
  - 卒業論文：対話型遺伝的アルゴリズムによる自動作曲
- 長岡技術科学大学（学部3年〜修士）
  - 電気電子情報工学課程 → 電気電子情報工学専攻
  - 修士論文：文章読解支援のための語彙平易化
  - 研究インターン@富士通研究所　2012.10 - 2013.02
- 首都大学東京（博士）
  - システムデザイン研究科、情報通信システム学域
  - 博士論文：テキスト平易化の話を書く予定
  - NLP若手の会プログラム委員　2014.11 - 2017.03
  - NLP東京Dの会（主催）　2015.04 - 2017.04
  - 客員研究員@リバプール大学　2015.09 - 2015.12
  - 協力研究員@情報通信研究機構　2017.05 - 2017.08

テキスト平易化

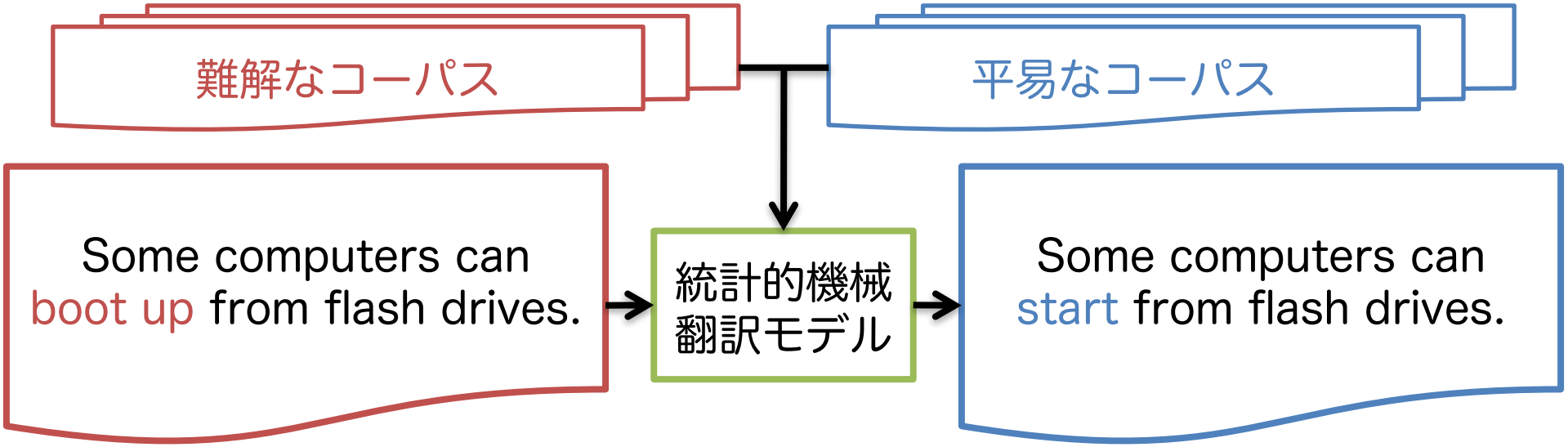
- English Wikipedia: Alfonso Perez
  - Alfonso Perez ~~Munoz, usually referred to as Alfonso,~~ is a former Spanish footballer~~, in the striker position~~.
- Simple English Wikipedia: Alfonso Perez
  - Alfonso Perez is a former Spanish football player.

読みやすくなるように文を書き換えるタスク

- 応用１：自然言語処理のために入力文の複雑さを減らす
- 応用２：言語学習者など人々の文章読解を助ける

テキスト平易化

- 同一言語内の翻訳問題（難解な英語 → 平易な英語）
- 難解なテキストと平易なテキストからなる
  パラレルコーパスを用意してトレーニングする



データない問題

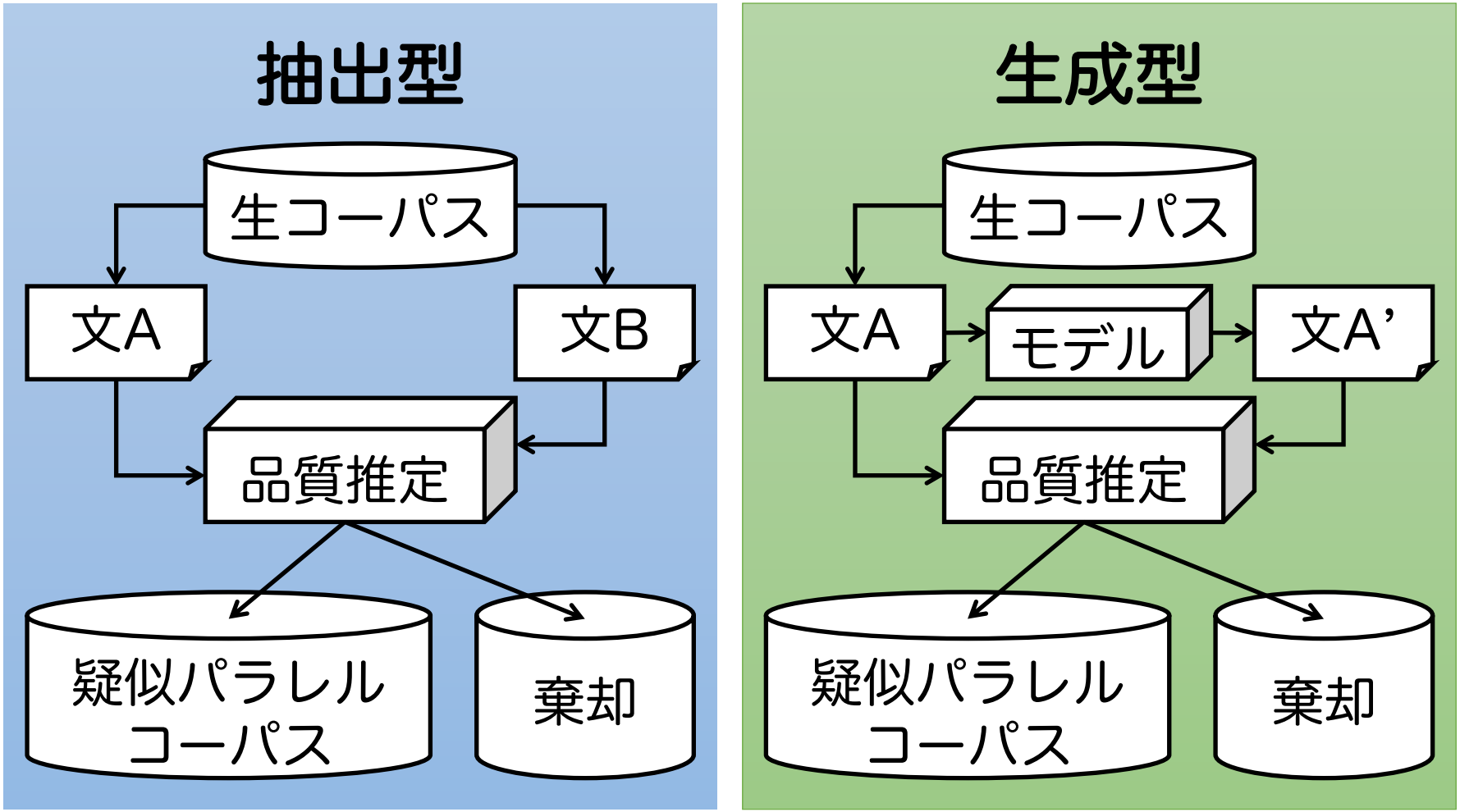
英語

- 平易なコーパス
  - Simple Wikipedia
- 難易のパラレルコーパス
  - Parallel Wikipedia
  - Newsela（人手で平易化）
- 英英パラレルコーパス
  - MSRP (News) 5K
  - TPC (Twitter) 18K
- 英仏パラレルコーパス
  - Europarl 2M
  - UN 25M

日本語

- 平易なコーパス
  - なし
- 難易のパラレルコーパス
  - なし
  - なし
- 日日パラレルコーパス
  - なし
  - なし
- 日英パラレルコーパス
  - 田中コーパス 150K
  - KFTT 443K

品質推定と疑似パラレルコーパス



品質推定と疑似パラレルコーパス

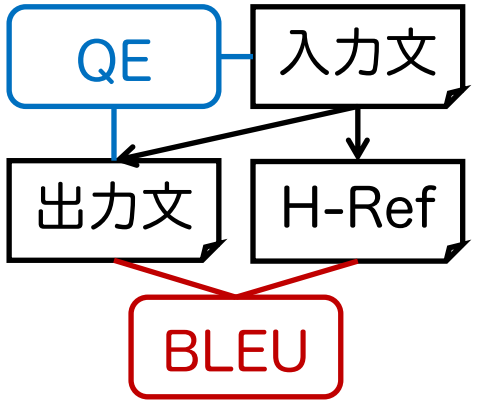
- 機械翻訳などのテキスト生成タスクにおいて、
  大規模なパラレルコーパスを利用できない問題がある

- 疑似パラレルコーパスを構築するために、
  品質推定を利用する新しいフレームワークを提案する
  - 抽出型
  - 生成型

- 生コーパスは実質無限に利用することができるため、
  大規模な疑似パラレルコーパスを獲得できる
  - 英語のテキスト平易化：既存のパラレルコーパスを
    使って訓練したモデルと同等の性能を達成
  - 露日の機械翻訳：Google翻訳を上回る性能を達成

品質推定（QE: Quality Estimation）

- Human Referencesに頼ることなく、
  入力文と出力文のみを用いて出力文の品質を推定する
- 語、句、文、文書、の各粒度で研究が進められている



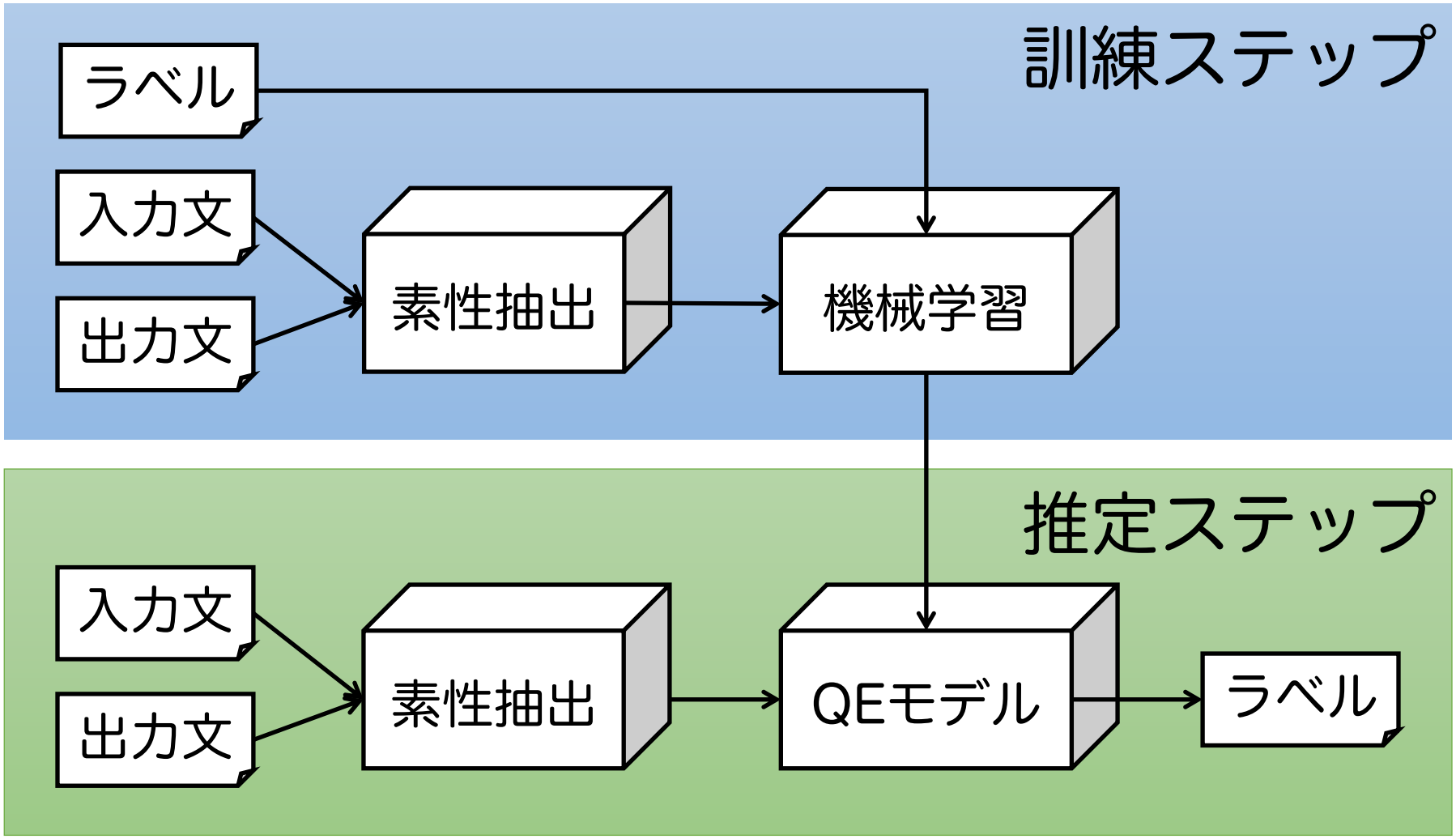
BLEUなどの従来の自動評価尺度

- Human Referencesだけが正解ではない
- 実際の運用時には利用できない

品質推定が有用なシナリオ

- 複数の機械翻訳から、より良い翻訳を選択して利用する
- 機械翻訳に対して、質の低い語句を後編集して利用する

品質推定（QE: Quality Estimation）



品質推定の先行研究

- Quality Estimation for Language Output Applications
  - COLING-2016のチュートリアル
  - http://coling2016.anlp.jp/tutorials/T4/
- WMT: Workshop on Statistical Machine Translation
  - 2012年以降、毎年QEのshared taskを開催
  - http://www.statmt.org/wmt12/quality-estimation-task.html
  - http://www.statmt.org/wmt17/quality-estimation-task.html
- Quality Assessment for Text Simplification
  - LREC-2016のワークショップ
  - http://qats2016.github.io/

品質推定の先行研究

機械翻訳

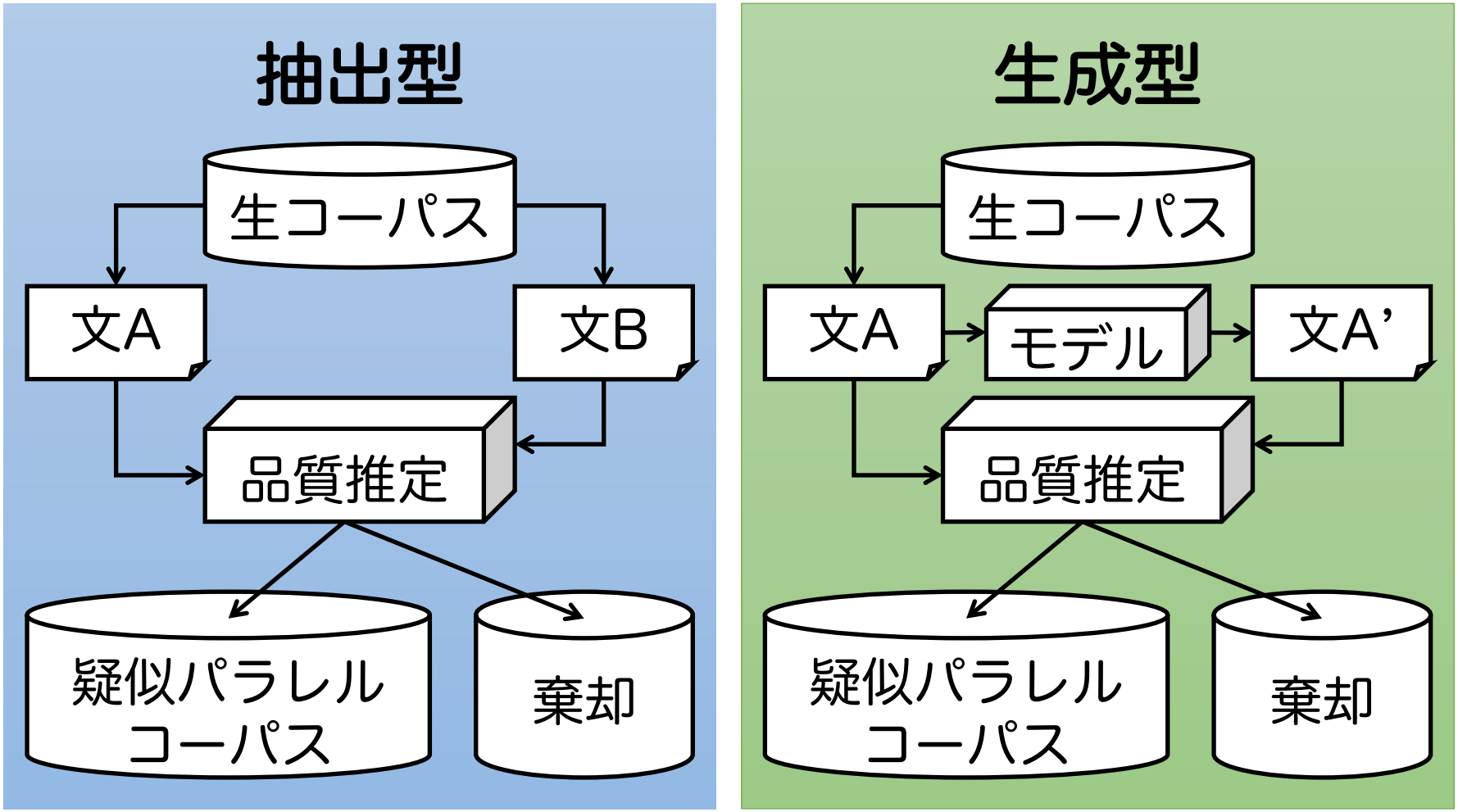
- Source Complexity Features:
  - 文長、平均単語長、構文木の深さ
  - N-gram言語モデルスコア（N=3）
- Target Fluency Features:
  - 文長、 (や”の開閉のミスマッチ率
  - N-gram言語モデルスコア（N=3）
- Adequacy Features:
  - 入出力の語数、品詞の割合、固有名詞の数
- Confidence Features:
  - 翻訳スコア、未知語の比率、N-best密度（語彙サイズ、平均文長）
- Others:
  - 語レベルのQE
  - 句レベルのQE

テキスト平易化

- Quality Features:
  - 機械翻訳と同じ
- Simplicity Features:
  - リーダビリティスコア

Task	Train/Dev/Test	Pearson
機械翻訳（WMT16、英→独）	12K/1K/2K	0.525
テキスト平易化	505/0/126	0.343

品質推定と疑似パラレルコーパス

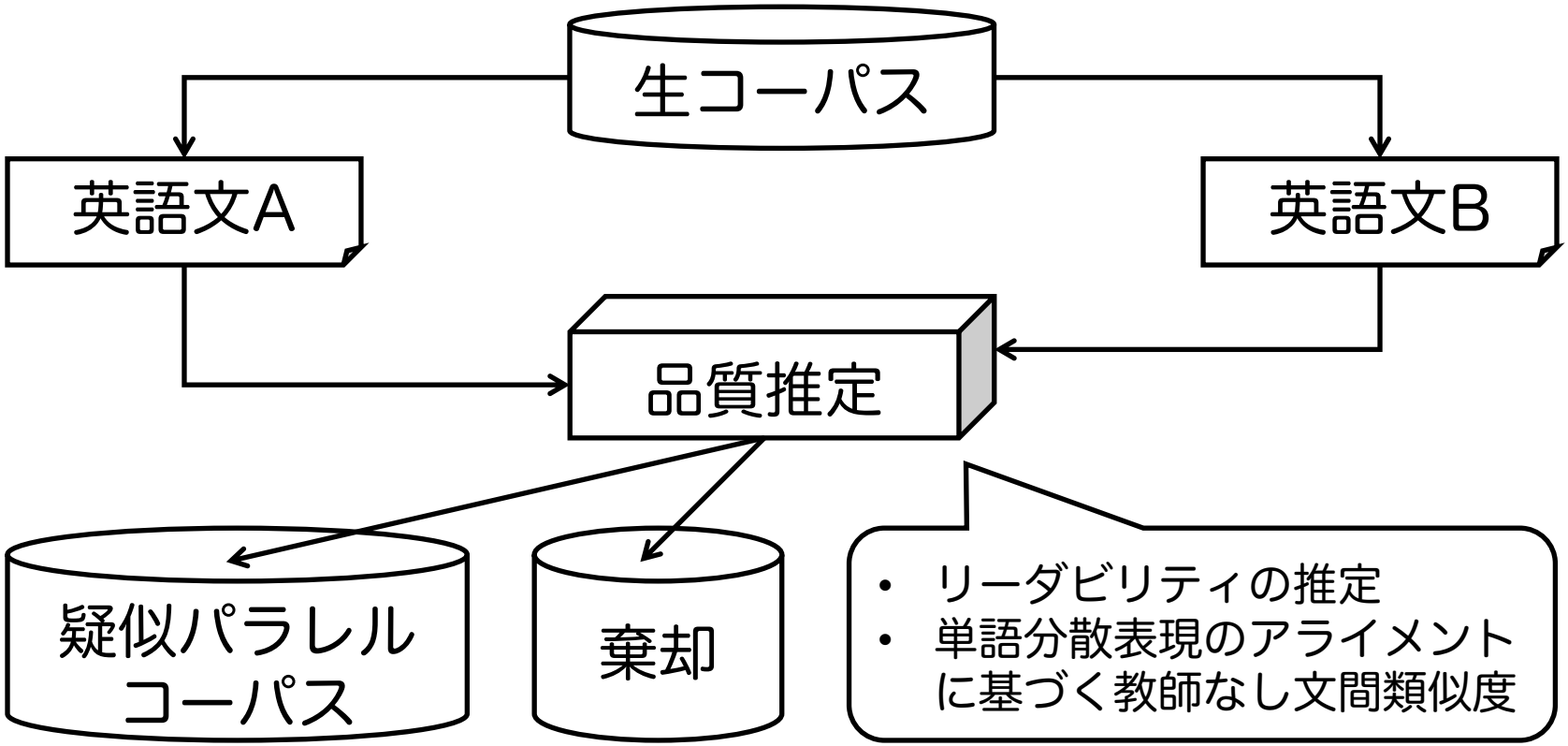


テキスト平易化　　言い換え　　機械翻訳

平易なコーパスを用いない テキスト平易化のための 単言語パラレルコーパスの構築

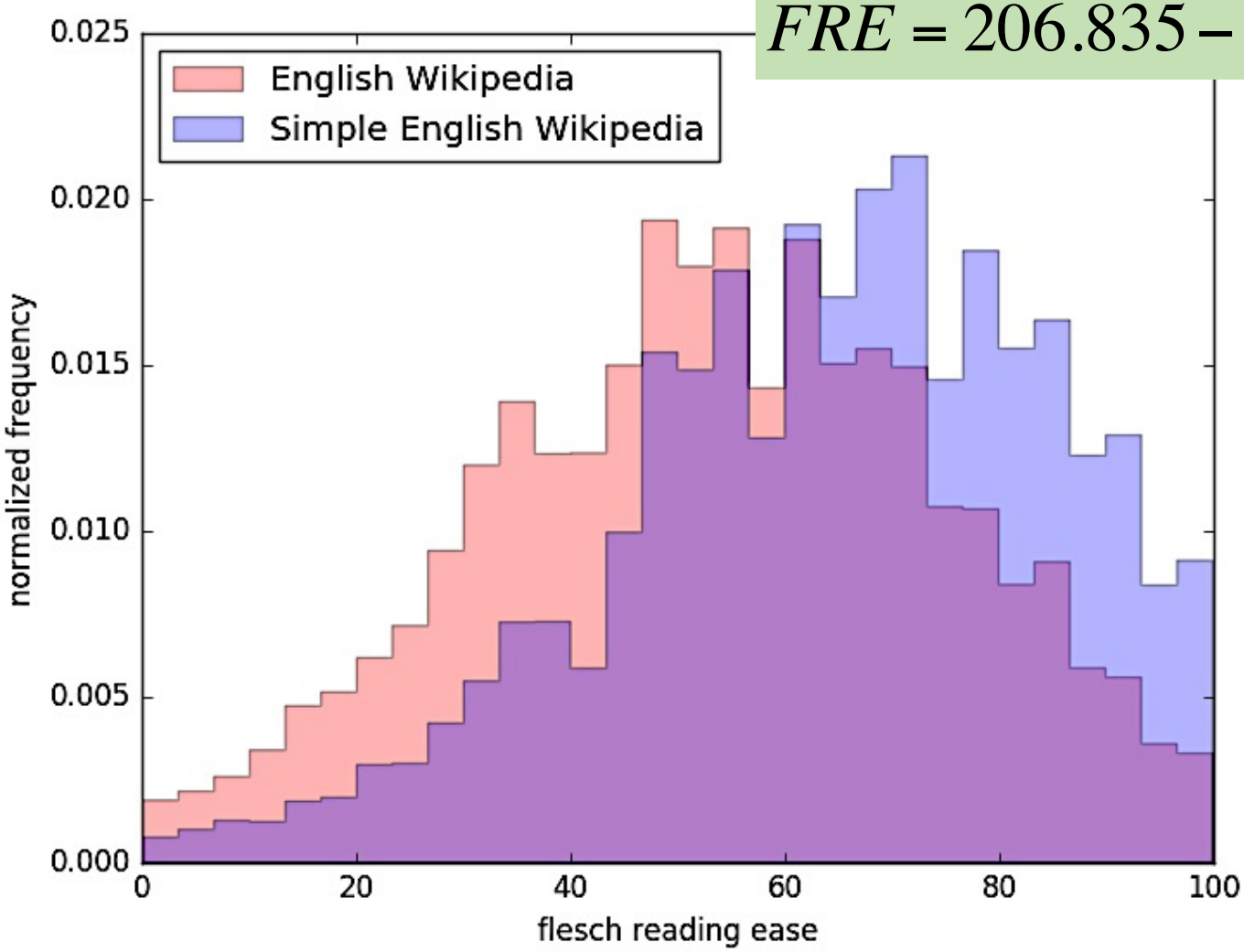
第229回情報処理学会自然言語処理研究会（2016/12）

難解な英語と平易な英語の疑似パラレルコーパスを構築



テキスト平易化の品質推定：リーダビリティと文間類似度

品質推定：リーダビリティ



$$FRE = 206.835 - 1.015\alpha - 84.6\beta$$

α：単語数
β：平均音節数

90 ~ 100	Very Easy
80 ~ 89	Easy
70 ~ 79	Fairly Easy
60 ~ 69	Standard
50 ~ 59	Fairly Difficult
30 ~ 49	Difficult
0 ~ 29	Very Difficult

品質推定：文間類似度

- 文xyの類似度を、アラインされた単語類似度の平均値で定義
- 単語類似度$\phi(x_i,y_j)$にはCBOWベクトルのCOS類似度を使う
- 各単語x_iに対して、最も類似度が高い単語y_jをアラインする
- S_{asym}は非対称なスコアなので、両方向の平均値を取る
- ノイズ軽減のため、ϕ < (閾値) の単語対はアラインしない

$$S_{asym}(x,y) = \frac{1}{|x|}\sum_{i=1}^{|x|}\max_j \phi(x_i, y_j)$$

$$S_{sym}(x,y) = \frac{1}{2}(S_{asym}(x,y) + S_{asym}(y,x))$$

Song and Roth (2015) Unsupervised Sparse Vector Densification for Short Text Similarity

文アライメントの性能

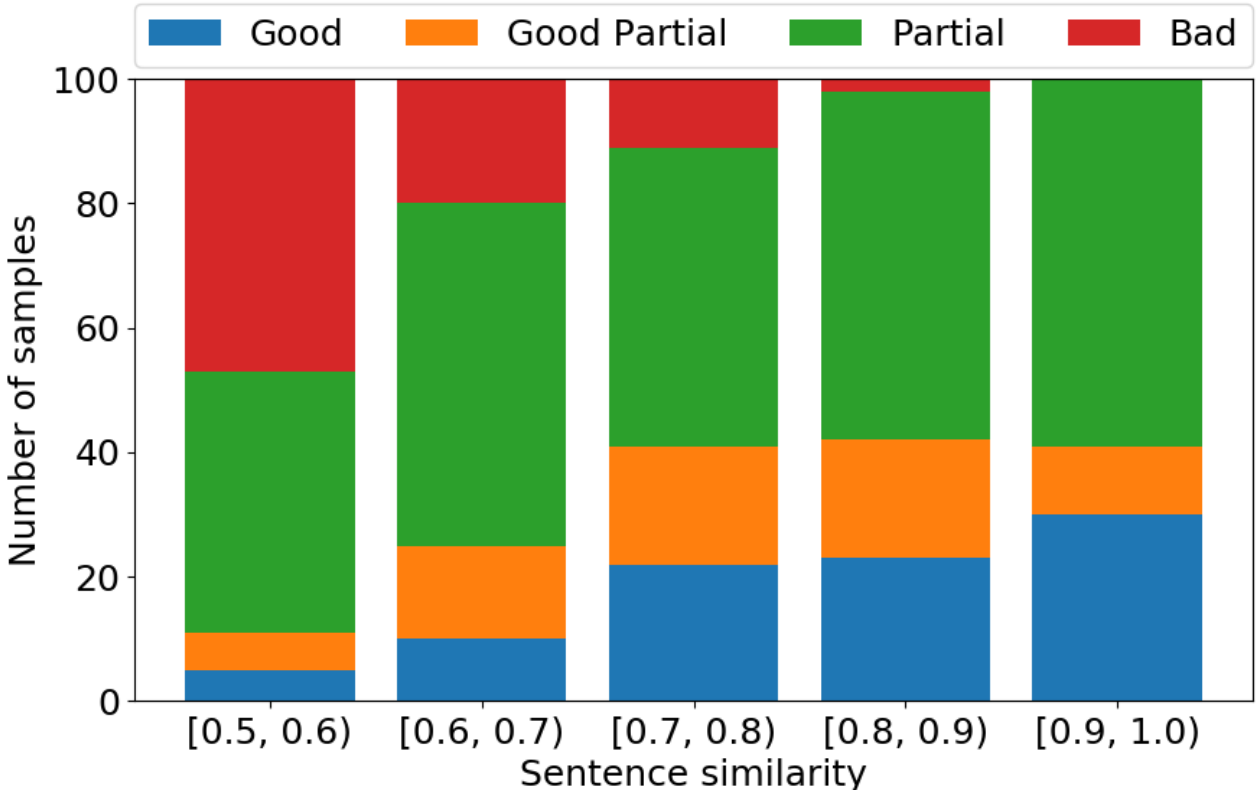
	同義 vs. その他		同義 & 含意 vs. その他	
	MaxF1	AUC	MaxF1	AUC
Zhu et al. (2010)	0.550	0.509	0.431	0.391
Coster and Kauchak (2011)	0.564	0.495	0.415	0.387
Hwang et al. (2015)	0.712	0.694	0.607	0.529
Additive Embeddings	0.691	0.695	0.518	0.487
Average Alignment	0.419	0.312	0.391	0.297
Maximum Alignment	**0.717**	**0.730**	**0.638**	**0.618**
Hungarian Alignment	0.524	0.414	0.354	0.499

- WikipediaとSimple Wikipediaから集めた文対に対して、人手で4段階のラベルを付与した評価用データセット
- 同義、含意、関連あり、関連なし

疑似パラレルコーパス

Wikipedia 628万文（Simple Wikipediaは使わない）

→ 200万文対の難易の疑似パラレルコーパスを獲得



既存のパラレルコーパスを使って訓練したモデルと同等の性能を達成

	文対数	規則数	FRE	BLEU	SARI
Baseline（入力文を書き換えずに出力する）	0	0/0	54.5	99.4	25.9
Zhu et al. (2010)	108k	7M/40k	59.7	84.7	34.7
Coster and Kauchak (2011)	137k	12M/49k	59.8	86.4	34.1
Hwang et al. (2015)	285k	25M/104k	61.0	81.3	34.5
品質推定：文間類似度のみ	493k	34M/129k	61.7	78.4	34.9
品質推定：リーダビリティ+文間類似度(>0.94)	100k	2M/4k	54.9	94.9	29.1
品質推定：リーダビリティ+文間類似度(>0.79)	500k	11M/18k	55.3	92.7	31.1
品質推定：リーダビリティ+文間類似度(>0.64)	1M	32M/50k	56.9	88.0	33.7
品質推定：リーダビリティ+文間類似度(>0.55)	1.5M	77M/103k	58.2	83.2	**34.4**
品質推定：リーダビリティ+文間類似度(>0.51)	2M	138M/161k	59.2	79.1	34.1
品質推定：リーダビリティ+文間類似度(>0.50)	2.1M	147M/170k	58.9	78.0	34.0

赤：WikipediaとSimple Wikipedia　　青：Wikipediaのみ
規則数：フレーズテーブルのエントリ / PPDBに含まれるフレーズペア

既存のパラレルコーパスを使って訓練したモデルと同等の性能を達成

Input	Offenbach's numerous operettas, such as *Orpheus in the Underworld*, and *La belle Hélène*, were extremely popular in both France and the English-speaking world during the 1850s and 1860s.
Ref 1	Offenbach's ~~numerous~~ operettas, such as *Orpheus in the Underworld*, and *La belle Hélène*, were ~~extremely~~ very popular in ~~both~~ France and the English-speaking world during the 1850's and 1860's.
Ref 2	Offenbach's ~~numerous~~ many operettas, such as *Orpheus in the Underworld*, and *La belle Hélène*, were ~~extremely~~ very popular in ~~both~~ France and in the English-speaking world during the 1850s and 1860s.
Kajiwara+ 2016	Offenbach's ~~numerous~~ many operettas, such as *Orpheus in the Underworld*, and *La belle Hélène*, were ~~extremely~~ very popular in both France and the English-speaking world during the 1850s and 1860s.
本研究	Offenbach's ~~numerous~~ many operettas, such as *Orpheus in the Underworld*, and *La belle Hélène*, were extremely popular in ~~both~~ France and the English-speaking world during the 1850s and 1860s.

日本語（BCCWJ）でも実験

品質推定

- リーダビリティ：単語難易度（Simple PPDB: Japanese）の平均
- 文間類似度：単語分散表現のアライメントに基づく文間類似度

	文対数	規則数	BLEU	SARI
Baseline（入力文を書き換えずに出力する）	0	0	58.5	24.3
品質推定：リーダビリティ+文間類似度(>0.80)	1k	14k	52.9	30.9
品質推定：リーダビリティ+文間類似度(>0.75)	2k	50k	46.5	32.1
品質推定：リーダビリティ+文間類似度(>0.70)	12k	307k	40.1	**34.4**
品質推定：リーダビリティ+文間類似度(>0.65)	91k	2,610k	28.6	31.7
品質推定：リーダビリティ+文間類似度(>0.60)	471k	15,129k	19.9	28.3

平易なコーパスを用いない テキスト平易化のための 単言語パラレルコーパスの構築

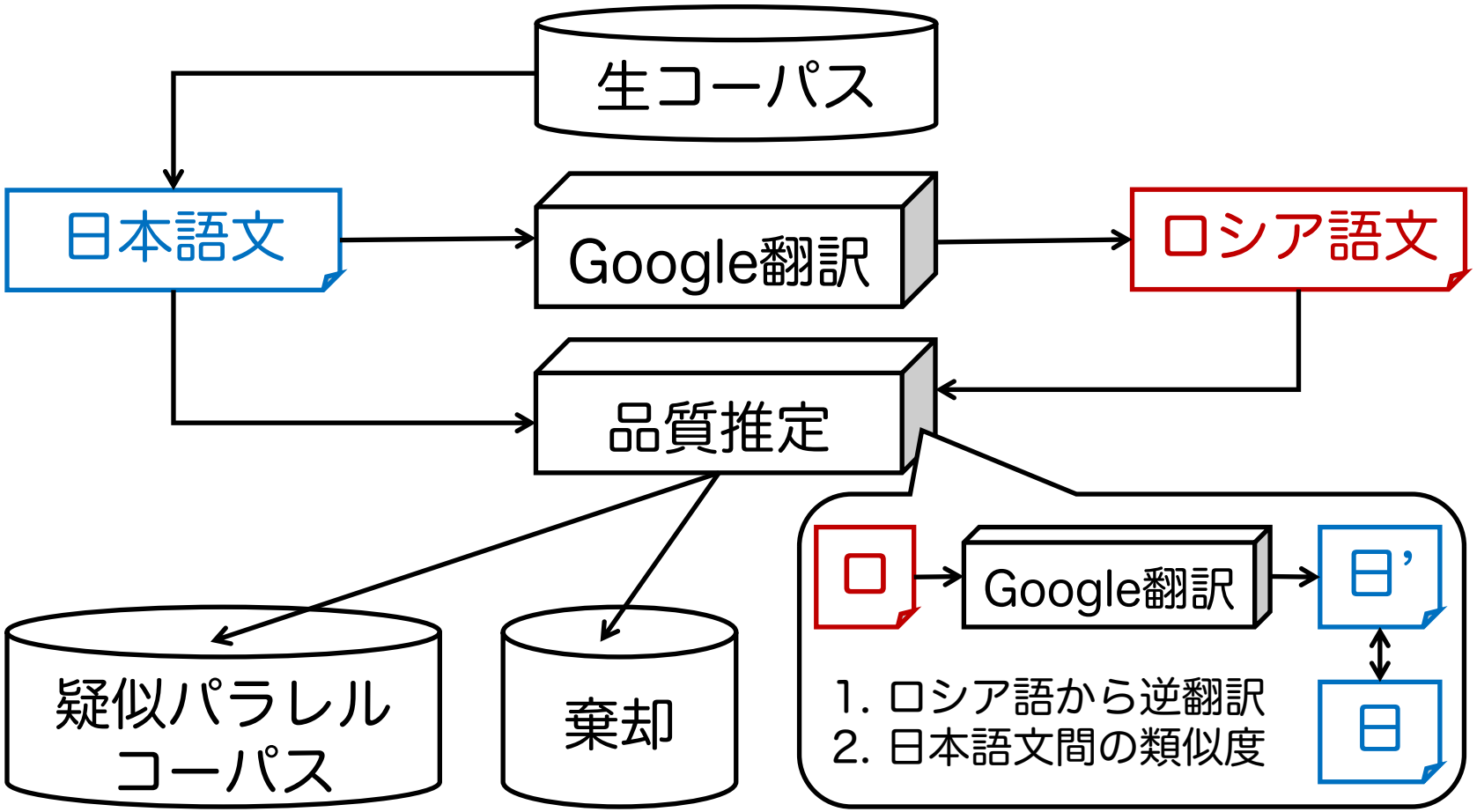
- 先行研究：Coster and Kauchak (2011) など
  - WikipediaとSimple Wikipediaのコンパラブルコーパス
  - TF-IDFベクトルのCOS類似度で文アライメントを取る

- 本研究：
  - 生コーパスと教師なし品質推定
  - リーダビリティ+単語アライメントに基づく文間類似度
  - Simple Wikipediaを使うのと同等の性能を達成
  - 言語非依存（英語と日本語の両方で性能を確認した）

逆翻訳による高品質な大規模疑似対訳コーパスの作成

言語処理学会第23回年次大会（2017/03）

日本語とロシア語$_{MT}$の疑似パラレルコーパスを構築



機械翻訳の品質推定：入力文と逆翻訳文のBLEU（WMT-2016のベスト）

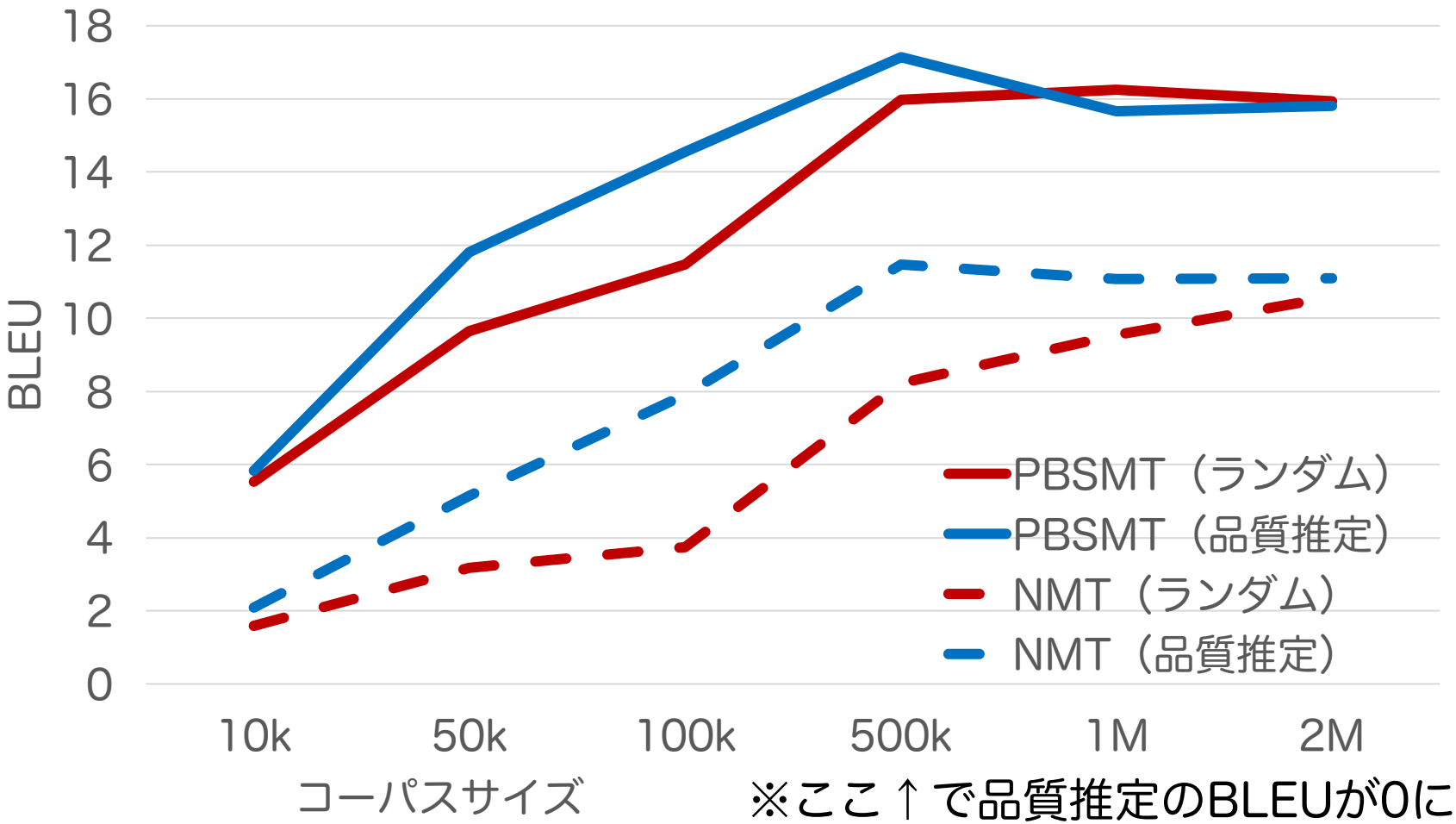
実験設定

- ベースライン
  - Google翻訳
  - 小規模コーパスを用いるPBSMTとNMT
  - 中規模コーパスを用いるピボット翻訳（PBSMT/NMT）
- データ（Tatoeba Project: 会話文）
  - 露日　Train:　1万文、Dev: 500文、Test: 500文
  - 露英　Train: 9.5万文、Dev: 500文、Test: 500文
  - 英日　Train: 9.5万文、Dev: 500文、Test: 500文
- 翻訳器
  - PBSMT: Moses
  - NMT: Bahdanau et al. (2015) の Attentionモデル

露→日の翻訳結果（BLEU）

手法	文対数	PBSMT（ランダム）	NMT（ランダム）	PBSMT（品質推定）	NMT（品質推定）
Google翻訳		19.44			
Baseline	10k	19.10	9.75		
ピボット	95k	11.51	11.10		
疑似コーパス	10k	14.66	4.11	17.19	7.90
疑似コーパス	50k	21.73	10.08	23.43	13.44
疑似コーパス	95k	24.65	13.67	25.15	13.73
疑似コーパス ＋ 対訳コーパス	95k	27.87	9.78	28.77	15.80

BCCWJを用いた大規模な実験



※ここ↑で品質推定のBLEUが0になる

逆翻訳による高品質な大規模疑似対訳コーパスの作成

- 先行研究：Sennrich et al. (2016)
  - 品質推定なしの疑似パラレルコーパス構築
  - 大規模なパラレルコーパスに追加して性能改善
- 本研究
  - 品質推定を導入
  - 疑似パラレルコーパスのみで
    オリジナルの翻訳器を上回る性能を達成
- 今後の課題
  - もっと弱い翻訳器でスタートできるか？
  - もっと強い翻訳器でスタートしても上回れるか？
  - コーパスを増やせばNMTがPBSMTに勝てるか？

複数の機械翻訳を用いた言い換え認識の評価用コーパス構築に向けて

言語処理学会第23回年次大会（2017/03）

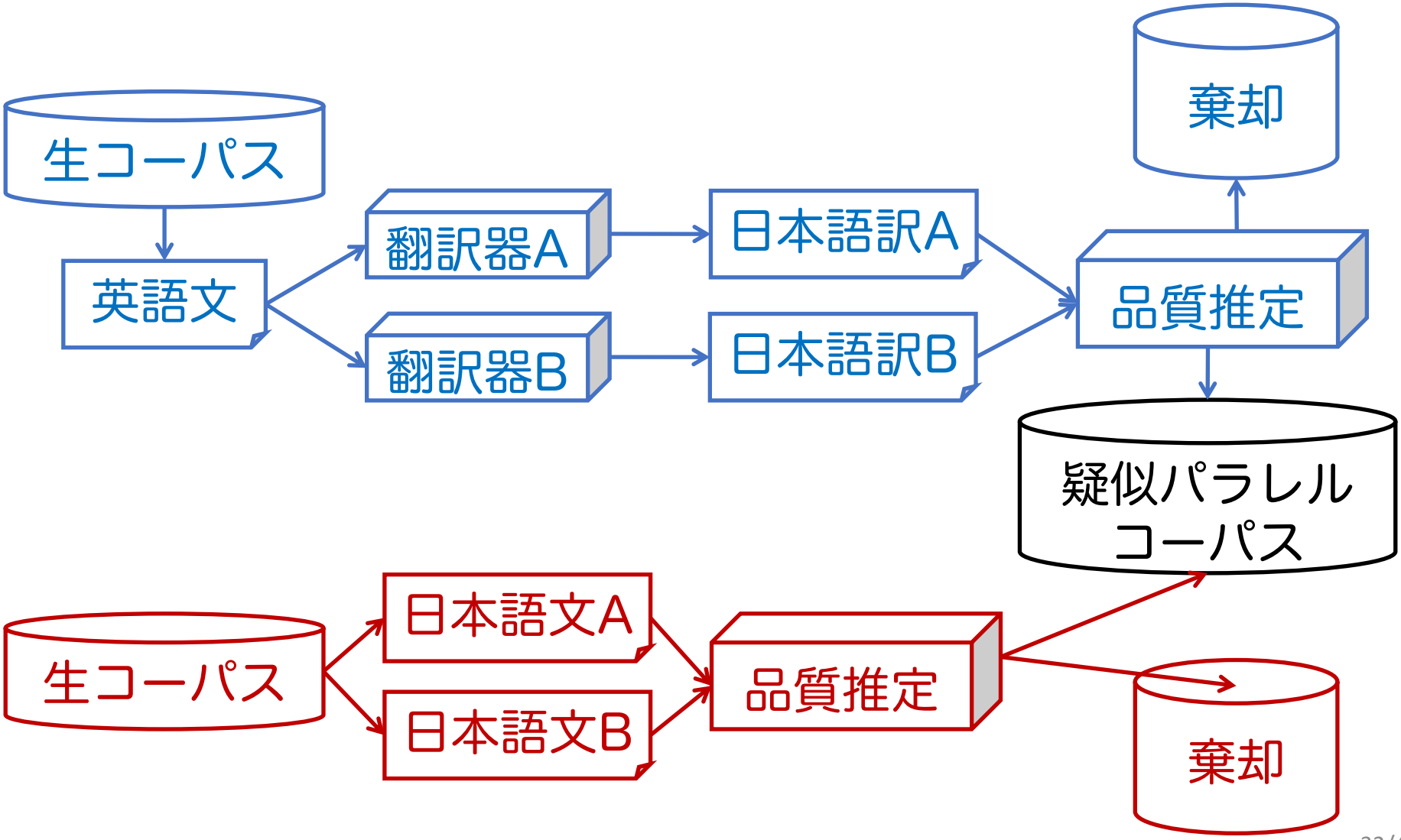
言い換え認識の評価用データ

- 機械翻訳やテキスト平易化などの
  テキスト生成タスクのための訓練用データとは異なる
  - 正例だけではダメ
  - 自明な例ばかりではダメ

- **非自明な正例** と **非自明な負例** を効率的に集めたい
  - 非自明な正例：多くの単語が違うのに言い換え
  - 非自明な負例：多くの単語が同じなのに非言い換え

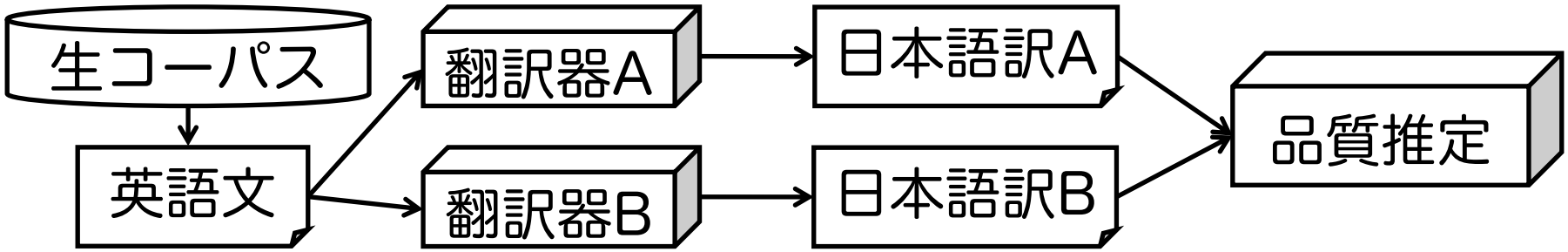
- 人間には生成させたくない
  - 一部のパターンの言い換えに偏りそう

正例を生成　負例を抽出


生コーパス
英語文
翻訳器A
翻訳器B
日本語訳A
日本語訳B
品質推定
棄却
疑似パラレル
コーパス
生コーパス
日本語文A
日本語文B
品質推定
棄却

正例の生成

- 50万文の英語文を用意（Wikipedia）
- 2つの翻訳器でそれぞれ日本語訳を生成する
  - 旧Google翻訳（PBSMT）
  - 新Google翻訳（NMT）
- 品質推定１：入力文（英）と逆翻訳（日→英）のBLEU
- 品質推定２：単語一致率ごとに一様サンプリング
- 最終的に人間が2,000文をチェック



自動生成された言い換え文対

Jaccard	文対数	サンプル	正例	負例	非文	その他
[0.0, 0.1)	228	200	2	1	80	117
[0.1, 0.2)	2,117	200	11	14	147	28
[0.2, 0.3)	14,080	200	20	9	162	9
[0.3, 0.4)	51,316	200	24	15	161	0
[0.4, 0.5)	100,674	200	27	16	151	6
[0.5, 0.6)	134,101	200	34	16	142	8
[0.6, 0.7)	100,745	200	38	13	129	20
[0.7, 0.8)	55,610	200	53	12	131	4
[0.8, 0.9)	26,884	200	81	3	94	22
[0.9, 1.0)	8,071	200	73	3	56	68
[1.0, 1.0]	6,174	0	0	0	0	0
Total	500,000	2,000	363	102	1,253	282

負例の抽出

- 単語一致率が高いのに言い換えではない
  「非自明な負例」がほしい
- ランダムに抽出した文対 → おそらく言い換えではない

- Wikipediaから2文を無作為抽出（どうせ言い換えではない）
- 品質推定：単語一致率が高い文対だけ残す

生成+抽出で得られた言い換え文対

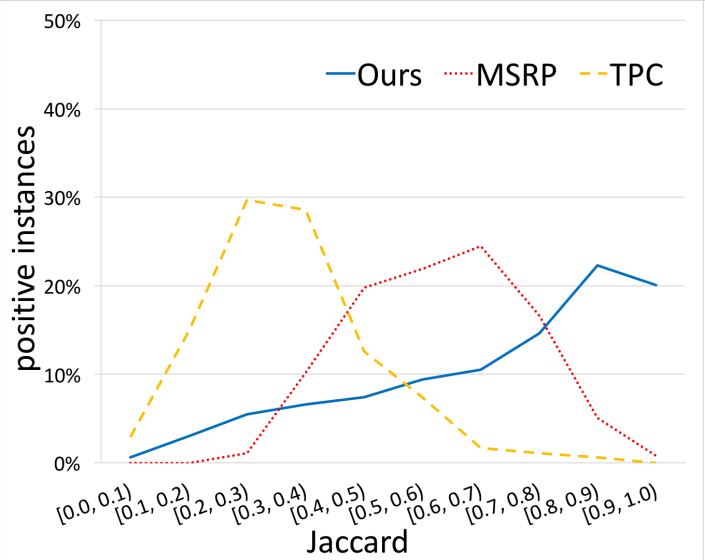
Jaccard	文対数	サンプル	正例	負例	非文	その他
[0.0, 0.1)	228	200	2	1	80	117
[0.1, 0.2)	2,117	200	11	14	147	28
[0.2, 0.3)	14,080	200	20	9	162	9
[0.3, 0.4)	51,316	200	24	15	161	0
[0.4, 0.5)	100,674	200	27	16	151	6
[0.5, 0.6)	134,101	200	34	16	142	8
[0.6, 0.7)	100,745	200	38	13	129	20
[0.7, 0.8)	55,610	200	53	52	131	4
[0.8, 0.9)	26,884	200	81	83	94	22
[0.9, 1.0)	8,071	200	73	73	56	68
[1.0, 1.0]	6,174	0	0	0	0	0
Total	500,000	2,000	363	292	1,253	282

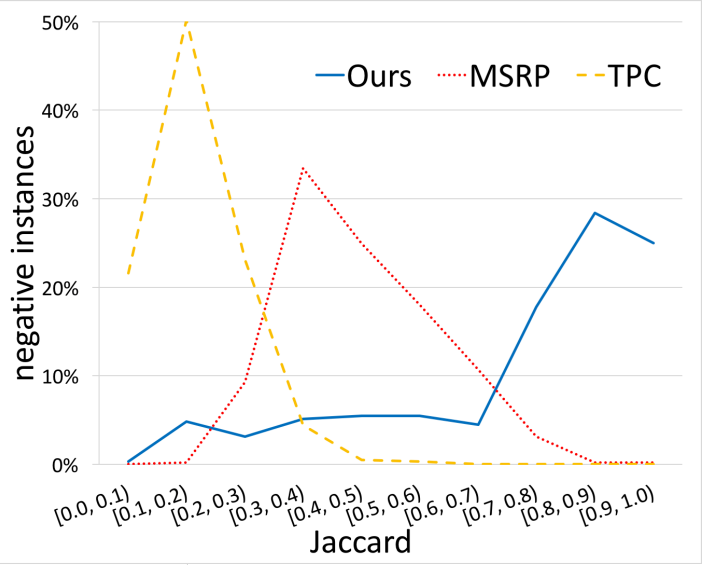
生成した負例と抽出した負例

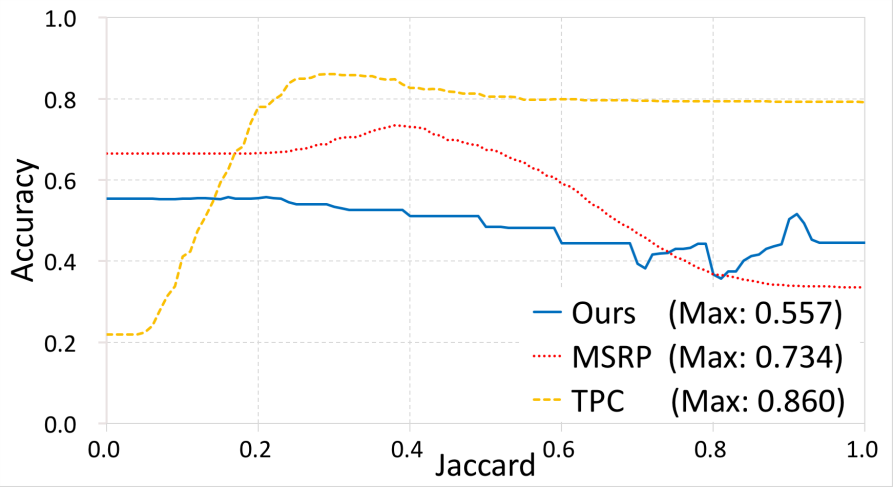
- 生成した負例30文対、抽出した負例70文対を3人に見せた
- 生成 / 抽出を人間が当てる2値分類 → F値: 0.34
- 生成した負例と抽出した負例の見分けがつかない → 良い

- 生成した負例
  - 他の役員のほとんどが海に戻りました。
  - 他の将校のほとんどは海に戻った。
- 抽出した負例
  - ミスターシービーは日本の競走馬。
  - シービークインは日本の競走馬。

既存の言い換え認識コーパスとの比較



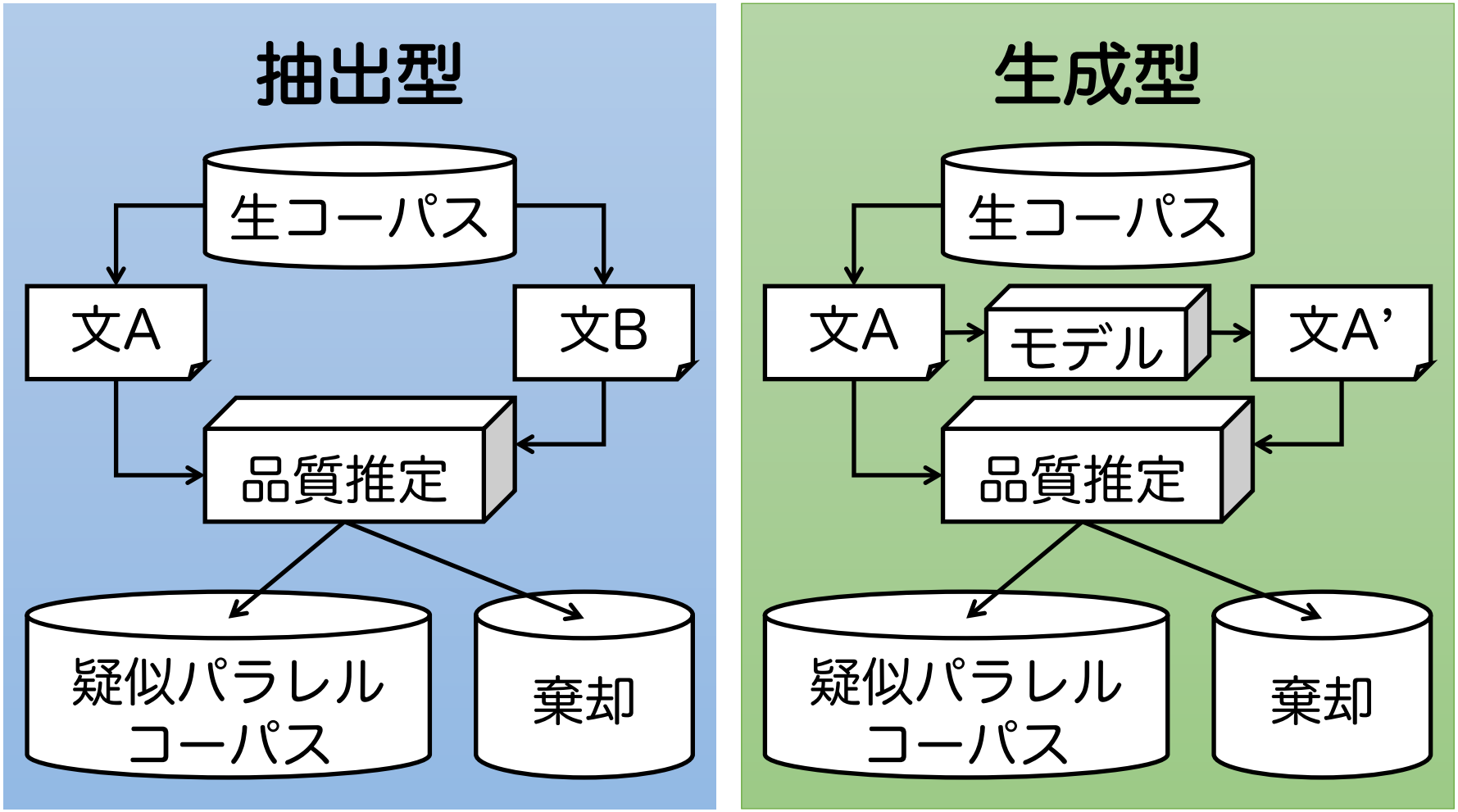




↑ 我々のコーパスは偏りが少ない

← 単語一致率だけでは解けない
Accuracy≒0.5
評価用コーパスなので、
簡単に解けないのが良い

品質推定と疑似パラレルコーパス



テキスト平易化　　言い換え　　機械翻訳

まとめ・今後の課題

- 入出力の比較によって出力文を評価する**「品質推定」**を用いて、生コーパスから疑似パラレルコーパスを構築する**「抽出型」「生成型」**の新しいフレームワークを提案した

- 英語のテキスト平易化と露日の機械翻訳における実験から既存のパラレルコーパスを用いるのと同等以上の性能を疑似パラレルコーパスのみで達成できることを示した

- 今後は、教師あり学習を用いて真剣に品質推定をしたい
- 生成型のアプローチでは、良い疑似コーパスを作ることでモデルが改善されると、より良い疑似コーパスを作ることができると期待される → 繰り返し適用して効果を見たい