Document Alignment based on Overlapping Fixed-Length Segmentation using Optimal Transport

○王 小天[1]　宇津呂 武仁[1]　永田 昌明[2]

1. 筑波大学 システム情報工学研究群 自然言語処理研究室

2. NTTコミュニケーション科学基礎研究所

目次

目次

研究背景

◆ウェブクロールデータ

- 機械翻訳に関する研究では、ウェブクローリングした多言語データを利用して、パラレルコーパスを構築することが非常に重要
- 代表的には、ParaCrawl Dataset (Bañón et al., ACL 2020)、JParaCrawl Dataset (Morishita et al., LREC 2022)など
- ウェブクロールデータの作成手順：クローリング、テキスト抽出、二言語間の文書対応付け、二言語間の文対応付け、文ペアのフィルタリング

研究背景

◆二言語間の文書対応付け

■定義

異なる文書集合の中でお互いに対訳になっている文書を結びつけること

■目的

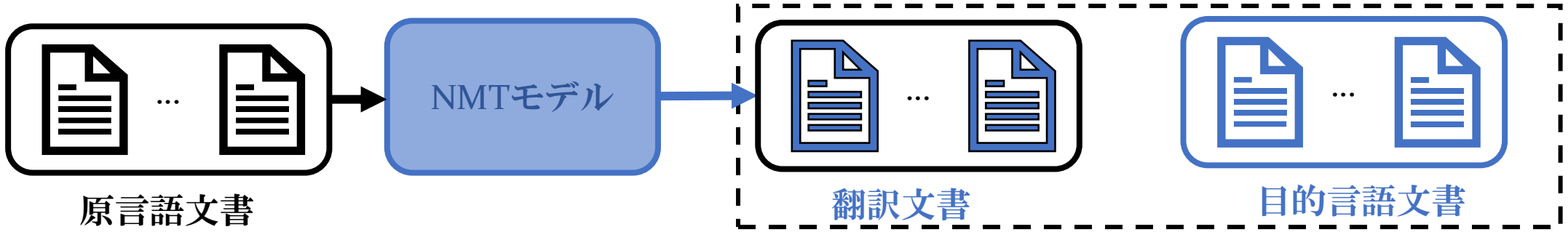
ウェブクロールから取得した多言語の文書に対して、対訳コーパスを作るために、

文書の二言語間対応関係を作る

目次

◆機械翻訳に基づく二言語間の文書対応付け (Buck and Koehn, WMT2016)

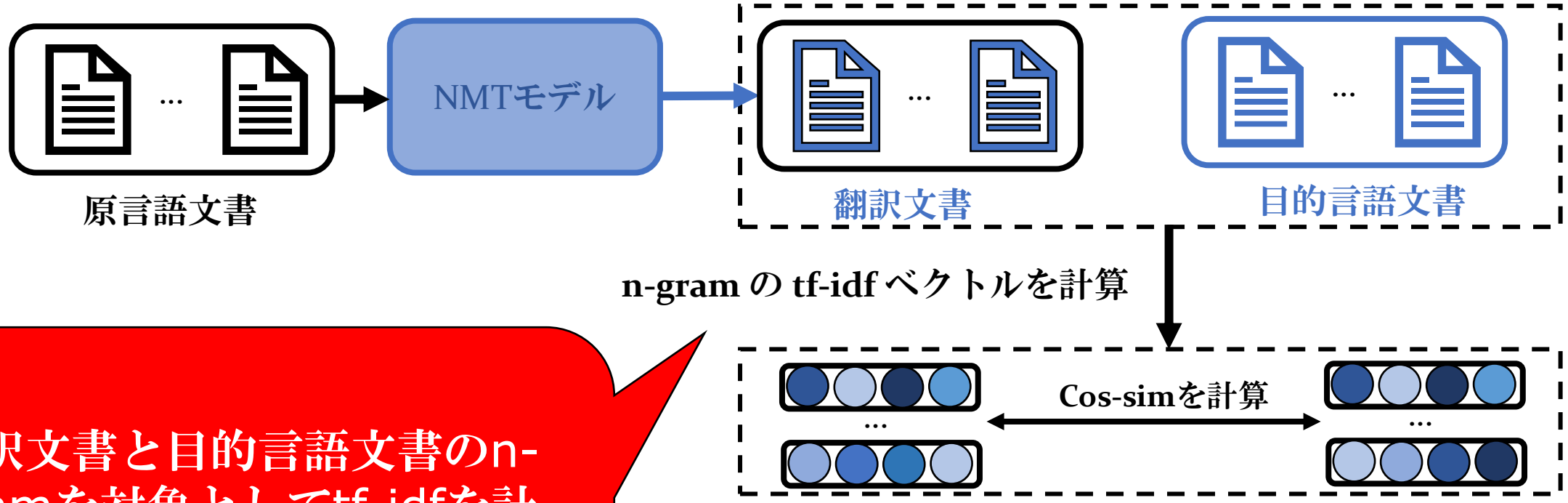
■手順



原言語文書の文単位分割された文を対象として、目的言語に機械翻訳する

◆機械翻訳に基づく二言語間の文書対応付け (Buck and Koehn, WMT2016)

■手順

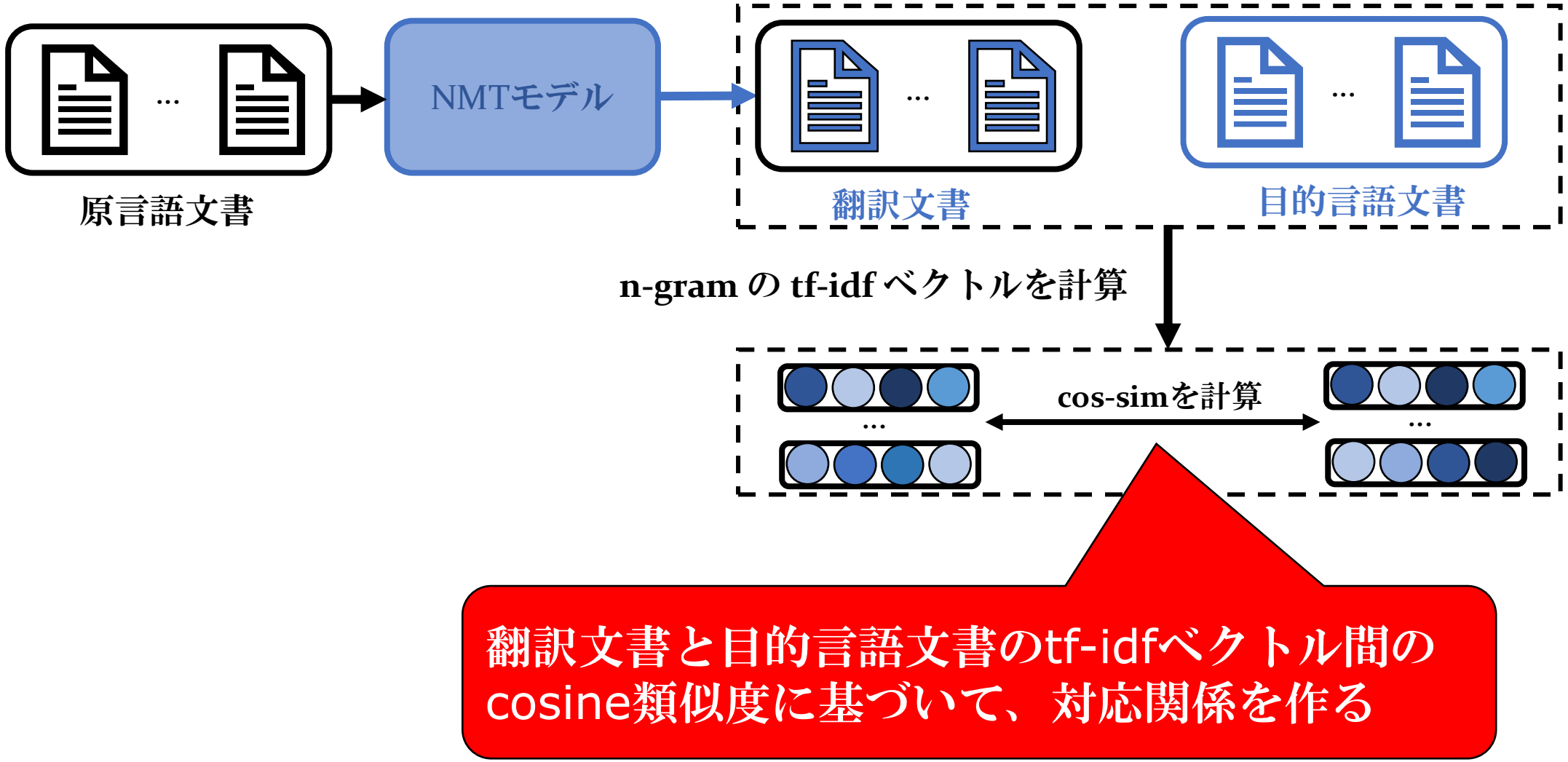


翻訳文書と目的言語文書のn-gramを対象としてtf-idfを計算し、各文書に対してtf-idfベクトルを生成する

先行研究|機械翻訳に基づく二言語間の文書対応付け (Buck and Koehn, WMT2016)

◆機械翻訳に基づく二言語間の文書対応付け (Buck and Koehn, WMT2016)

■手順



◆機械翻訳に基づく二言語間の文書対応付け (Buck and Koehn, WMT2016)

■対応関係の作成ルール (1-1ルール)

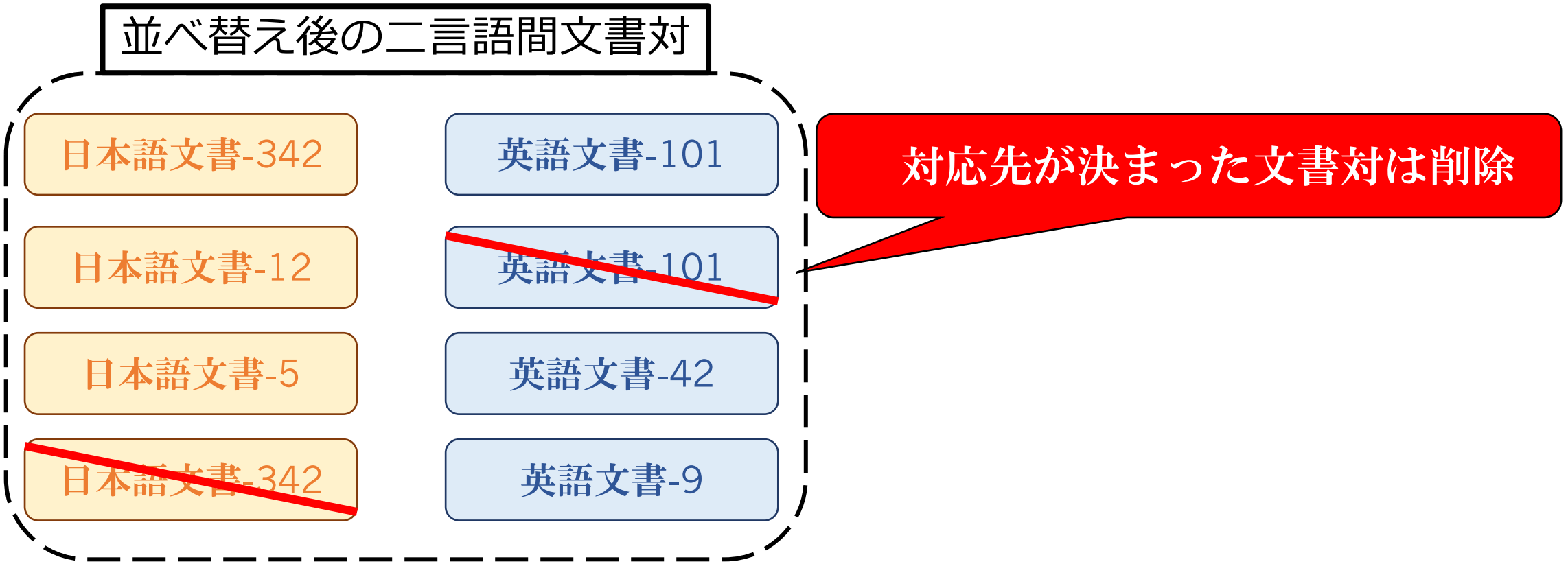
- 二言語間文書対の全候補を類似度の高い順に並べ替え、対応先が決まった文書対は削除
- 各文書が最大で一つの対応文書しか持たないようにする

先行研究|機械翻訳に基づく二言語間の文書対応付け (Buck and Koehn, WMT2016)

◆機械翻訳に基づく二言語間の文書対応付け (Buck and Koehn, WMT2016)

■対応関係の作成ルール (1-1ルール)

- 二言語間文書対の全候補を類似度の高い順に並べ替え、対応先が決まった文書対は削除
- 各文書が最大で一つの対応文書しか持たないようにする

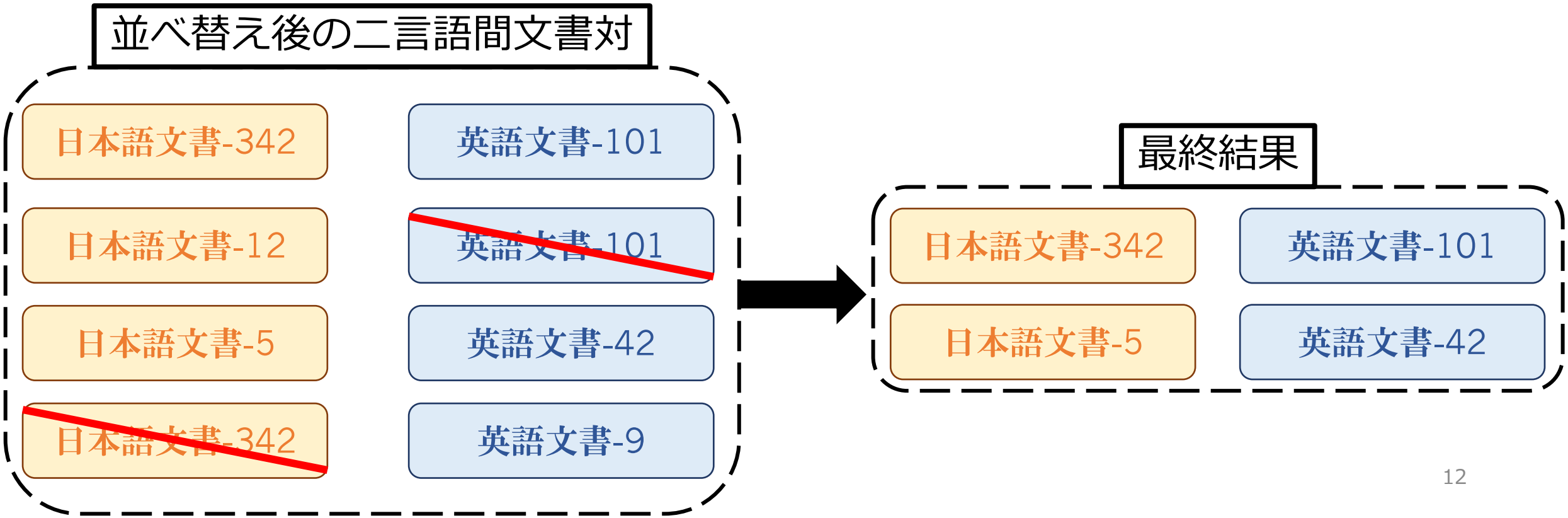


先行研究|機械翻訳に基づく二言語間の文書対応付け (Buck and Koehn, WMT2016)

◆機械翻訳に基づく二言語間の文書対応付け (Buck and Koehn, WMT2016)

■対応関係の作成ルール (1-1ルール)

- 二言語間文書対の全候補を類似度の高い順に並べ替え、対応先が決まった文書対は削除
- 各文書が最大で一つの対応文書しか持たないようにする



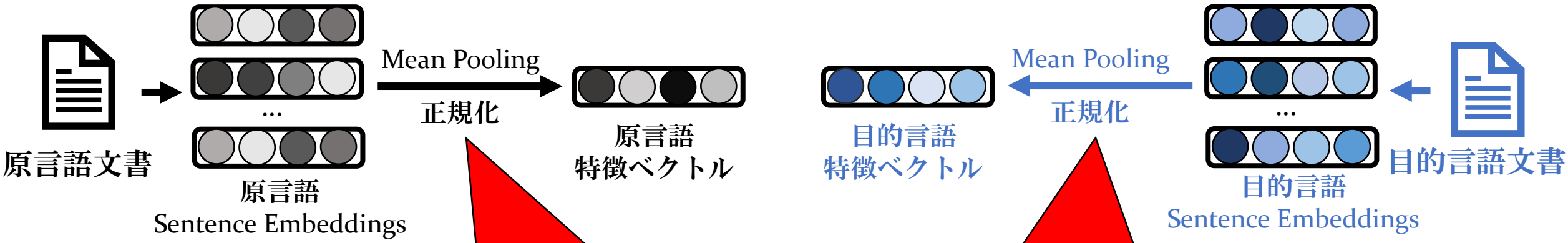
◆Sentence Embeddingに基づく二言語間の文書対応付け (Mean Pooling)

■手順

原言語文書と目的言語文書の分割された文を対象として、事前訓練済み多言語Sentence Embeddingモデルにより、それぞれに文埋め込みに変換

◆Sentence Embeddingに基づく二言語間の文書対応付け (Mean Pooling)

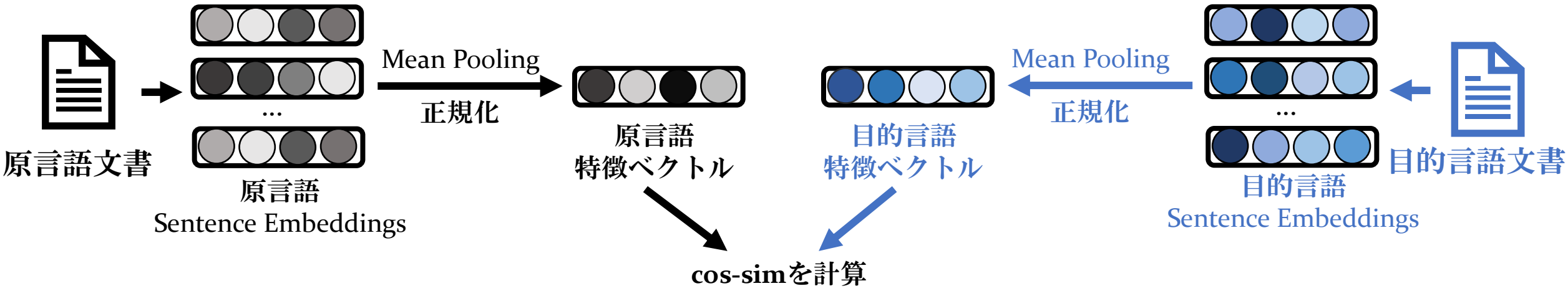
■手順



各文書の系列ベクトルをMean Poolingし、次に正規化して、文書の特徴ベクトルを取得する

◆Sentence Embeddingに基づく二言語間の文書対応付け (Mean Pooling)

■手順



原言語文書と目的言語文書間の特徴ベクトルのcosine類似度に基づいて、対応関係を作る (1-1ルールに従う)

先行研究| Sentence Embeddingに基づく二言語間の文書対応付け (Optimal Transport)

◆Sentence Embeddingに基づく二言語間の文書対応付け (Optimal Transport)

(Clark et al., ACL 2019; El-Kishky and Guzmán, AACL 2020)

■手順

1. 原言語文書と目的言語文書に対して、事前訓練済み多言語Sentence Embeddingモデルで、それぞれの系列ベクトルを生成する
2. Sentence-Levelの文書間の最適輸送距離 (Optimal Transport Distance)、Sentence Mover's Distance (SMD)を計算する

(最適輸送距離 (OT)の計算方式は次スライド)

◆Sentence Embeddingに基づく二言語間の文書対応付け (Optimal Transport)

(Clark et al., ACL 2019; El-Kishky and Guzmán, AACL 2020)

■OTの計算方式

- あるドキュメントDoc A, Doc B
- Doc A の s_i の重みは $d_{A,i}$ とする
- Doc B の s'_j の重みは $d_{B,j}$ とする
- **$c(i,j)$は s_i のEmbeddingと s'_jの Embedding の距離**
- T_{ij} の i 行目の和は$d_{A,i}$、j 列目の和は$d_{B,j}$が条件として、最小の$\sum_{i,j=1}^{n} T_{ij}c(i,j)$ を見つけるのが目的
- 見つかった minimum $\sum_{i,j=1}^{n} T_{ij}c(i,j)$ が最終の文間の最適輸送距離
- 重み付け方式の候補は右記

目標関数：

$$\min_{T \geq 0} \sum_{i=1}^{n} \sum_{j=1}^{m} T_{ij}\, c(i,j)$$

$$\text{Subject to:} \quad \sum_{j=1}^{m} T_{ij} = d_{A,i}$$

$$\sum_{i=1}^{n} T_{ij} = d_{B,j}$$

重み付け方式

- **文頻度による重み**
- **文長による重み**
- **IDFによる重み**
- **文長、IDFによる重み**

目次

◆提案手法：Sentence Embeddingに基づく二言語間の文書対応付け

1. ２特徴ベクトルによる文書対応

- Mean Poolingベクトルを利用するだけではなく、
  文書の系列ベクトルの中の第１文ベクトルも使用
- 文書間の類似度を計算する際に、第１文ベクトルとMean Poolingベクトルの間の類似度
  をそれぞれに計算して、それらに重みを与えて合計する

2. Overlapping Fixed-Length Segmentation (OFLS)による文書対応

- 文単位分割ではなく、固定長 *L* の Sliding Window で、
  ウェブクロールされた文書を重複ありの Segments に分割する
- OFLSをもとに、言語の影響を考慮して、原言語文書と目的言語文書を異なる固定長
  (言語間で一定の比率)で分割する (Language-Pair Dependent OFLS, LD-OFLS)
- 従来手法・提案手法において、「文単位分割」と「OFLS」の性能を比較する

◆提案手法：Sentence Embeddingに基づく二言語間の文書対応付け

1. ２特徴ベクトルによる文書対応

- Mean Poolingベクトルを利用するだけではなく、
  文書の系列ベクトルの中の第１文ベクトルも使用
- 文書間の類似度を計算する際に、第１文ベクトルとMean Poolingベクトルの間の類似度
  をそれぞれに計算して、それらに重みを与えて合計する

2. Overlapping Fixed-Length Segmentation (OFLS)による文書対応

- 文単位分割ではなく、固定長 *L* の Sliding Window で、
  ウェブクロールされた文書を重複ありの Segments に分割する
- OFLSをもとに、言語の影響を考慮して、原言語文書と目的言語文書を異なる固定長
  (言語間で一定の比率)で分割する (Language-Pair Dependent OFLS, LD-OFLS)
- 従来手法・提案手法において、「文単位分割」と「OFLS」の性能を比較する

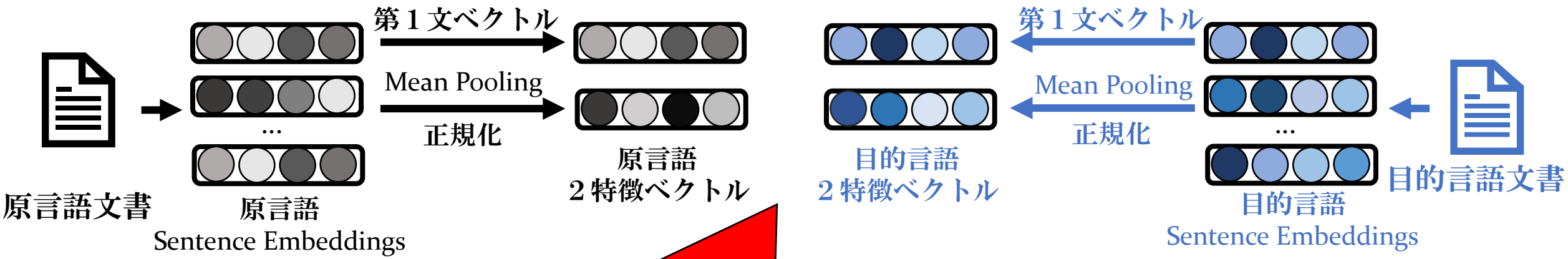
◆Sentence Embeddingに基づく二言語間の文書対応付け (2 特徴ベクトル)

■ 手順

**原言語文書と目的言語文書に対して、
それぞれの系列ベクトルを生成する**

◆Sentence Embeddingに基づく二言語間の文書対応付け (2 特徴ベクトル)

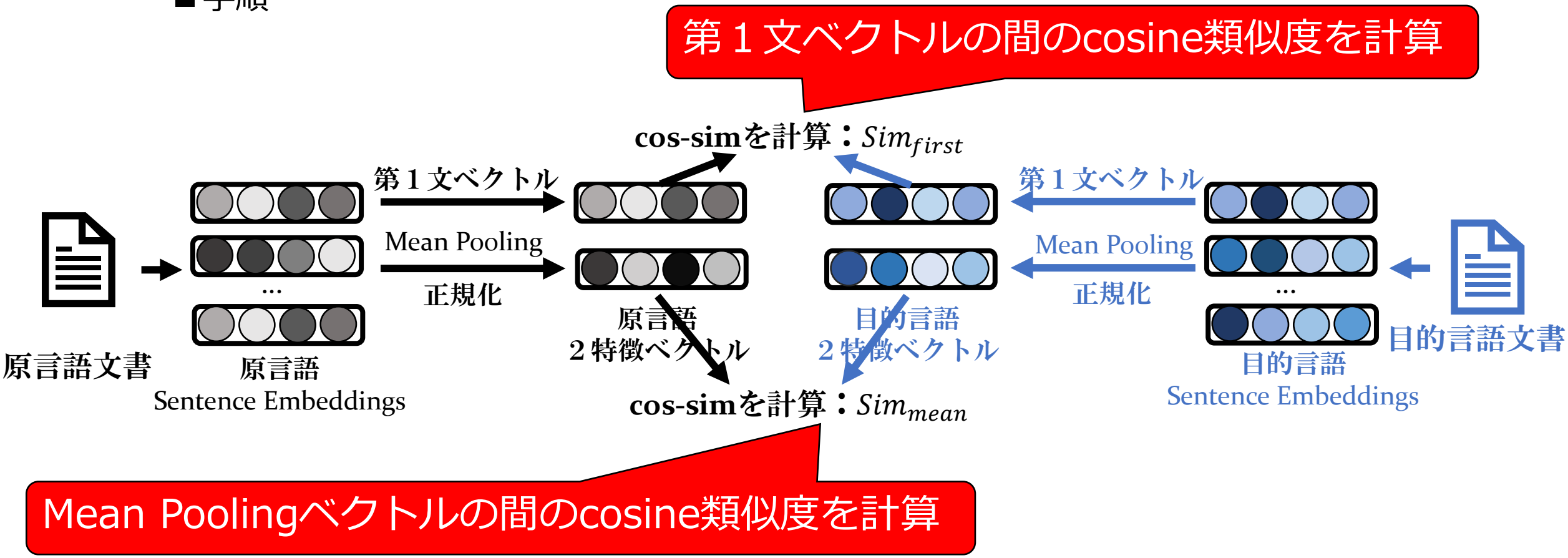
■ 手順



各文書の系列ベクトルのMean Poolingベクトルと第１文ベクトルを取得する

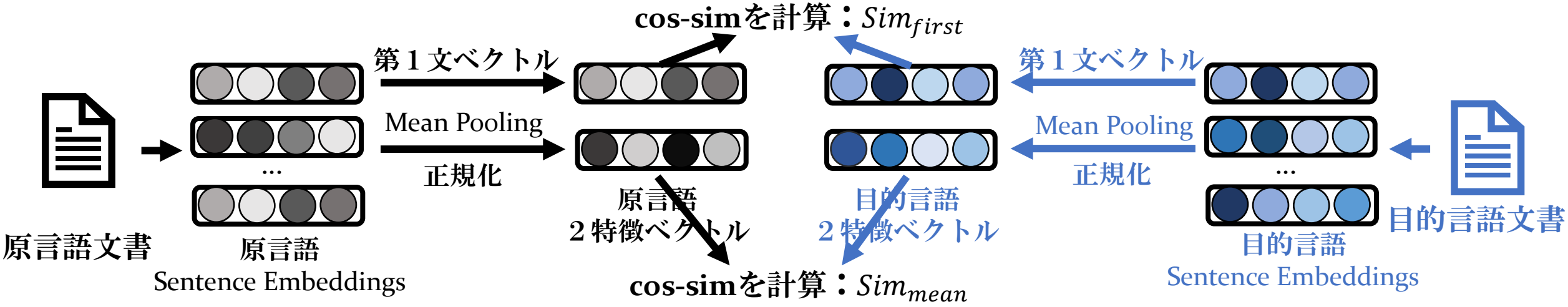
◆Sentence Embeddingに基づく二言語間の文書対応付け (2 特徴ベクトル)

■ 手順



◆Sentence Embeddingに基づく二言語間の文書対応付け (2 特徴ベクトル)

■ 手順



類似度の重み付き和を計算：$Sim_{final} = \lambda Sim_{first} + (1 - \lambda) Sim_{mean}$

◆提案手法：Sentence Embeddingに基づく二言語間の文書対応付け

1. ２特徴ベクトルによる文書対応

- Mean Poolingベクトルを利用するだけではなく、
  文書の系列ベクトルの中の第１文ベクトルも使用
- 文書間の類似度を計算する際に、第１文ベクトルとMean Poolingベクトルの間の類似度
  をそれぞれに計算して、それらに重みを与えて合計する

2. Overlapping Fixed-Length Segmentation (OFLS)による文書対応

- 文単位分割ではなく、固定長 *L* の Sliding Window で、
  ウェブクロールされた文書を重複ありの Segments に分割する
- OFLSをもとに、言語の影響を考慮して、原言語文書と目的言語文書を異なる固定長
  (言語間で一定の比率)で分割する (Language-Pair Dependent OFLS, LD-OFLS)
- 従来手法・提案手法において、「文単位分割」と「OFLS」の性能を比較する

◆Overlapping Fixed-Length Segmentation (OFLS)による文書対応

■定義

・固定長 L の Sliding Window を利用して Segments に分割
・隣接するSegmentsの間では、指定された割合が重複

■OFLSの例

$L = 20$
重複割合: 0.5

今日 は 晴れ て いて、 気温 は 25 度 ほど で、 風 も 穏やか です。 公園 で 友達 と ピクニック を 楽しむ 予定 です。 お弁当 に は おにぎり、 サンドイッチ、 フルーツ、 そして 冷たい ジュース が 入っ て い ます。 みんな で 楽しい 時間 を 過ごし、 笑顔 が 絶え ない こと を 願っ て います。

Seg1: 今日 は 晴れ て いて、 気温 は 25 度 ほど で、 風 も 穏やか です。 公園 で
Seg2: ほど で、 風 も 穏やか です。 公園 で 友達 と ピクニック を 楽しむ 予定 です。 お弁当
Seg3: 友達 と ピクニック を 楽しむ 予定 です。 お弁当 に は おにぎり、 サンドイッチ、 フルーツ、 そして 冷たい
Seg4: に は おにぎり、 サンドイッチ、 フルーツ、 そして 冷たい ジュース が 入っ て い ます。 みんな で 楽しい 時間
Seg5: ジュース が 入っ て い ます。 みんな で 楽しい 時間 を 過ごし、 笑顔 が 絶え ない こと を 願っ て います。

◆Language-Pair Dependent OFLS (LD-OFLS)による文書対応

■考慮点

一般的に、異なる言語で同じ意味を表現するために必要なトークン数は異なる

例：「私 は 犬 が 好き だ」と「I like dogs」

日本語：6 トークン、英語：3 トークン

⇒固定長分割において、原言語文書・目的言語文書で異なる固定長を使用

■手法

- ρ (language-pair dependent proportion)：言語間の文書長(トークン数)の比率
- 目的言語文書を分割する固定長＝原言語文書を分割する固定長＊ρ

◆Language-Pair Dependent OFLS (LD-OFLS)による二言語間文書対応付け

■具体例

The expected annual power generation will be 87,000,000kWh, corresponding to the annual electricity consumption of 30,000 ordinary houses.

「setu4993/LaBSE」でtokenize

The | expected | annual | power | generation | will | be | 87 | , | 000 | , | 000 | ##kw | ##h | …

総トークン数：28

総トークン数：42

年 | 間 | 予 | 想 | 発 | 電 | 量 | は | ##8 | , | 700 | 万 | キ | ##ロ | ##ワ | ##ット | 時 | で | 、 | 一 | 般 | 家 | …

「setu4993/LaBSE」でtokenize

年間予想発電量は8,700万キロワット時で、一般家庭約3万世帯分の年間消費電力量に相当します。

目次

デートセット

◆テストデータ

■デートセットの構築

- 4つのweb-domain **丸紅**、**西新宿**、**楽天**、**NTT CS研究**に対して、各ドメインからランダムに100 件ずつ日本語文書をサンプリングし、4つの手法で対訳となる英語文書を検索する
  - 機械翻訳＋BM25
  - 機械翻訳＋tf-idf
  - URL Matching
  - CC-Aligned (El-Kishky et al., EMNLP 2020)
- 人手で参照用日英対訳文書対を作成
- 検索結果の英語文書集合を、対応目的言語(英語)文書の候補集合として利用

4つのweb-domain
- www.marubeni.com
- nishishinjuku.co.jp
- corp.rakuten.co.jp___global.rakuten.com
- www.kecl.ntt.co.jp

	丸紅	西新宿	楽天	NTT CS研究	ALL
対象： 対訳あり日本語文書数	73	16	75	68	**232**
検索結果の 候補英語文書数	251	42	319	319	**931**

デートセット

◆テストデータ

■デートセットの情報

- 丸紅、西新宿、楽天、NTT、これらの4つのドメイン、およびそれらの混合データ (4つのドメインの混合データ)
- 各文書の長さはHugging Faceの「setu4993/LaBSE」でtokenizeした後のトークン数
- ρ = 正解の英語文書の平均トークン数/対訳あり日本語文書の平均トークン数（Ja-En方向）

	丸紅	西新宿	楽天	NTT CS研究	ALL
対象： 対訳あり日本語文書数	73	16	75	68	**232**
検索結果の 候補英語文書数	251	42	319	319	**931**
対訳あり日本語文書の 平均トークン数	2447.36	340.56	3541.55	726.37	**2151.35**
正解の英語文書の 平均トークン数	1598.51	217.69	2174.49	441.88	**1350.47**
ρ	0.65	0.64	0.61	0.61	**0.63**

目次

実験設定

◆実験設定

■Baseline: 機械翻訳に基づく二言語間の文書対応付け (Buck and Koehn, WMT2016)

- 翻訳機：JParaCrawl v3.0 ja-en model
- 翻訳方向：日本語から英語
- tf-idf で文書対応：Bitextorの「Docalign」ツールを使用 (bigramベースで実行)

■Sentence Embeddingに基づく二言語間の文書対応付け

- 文埋め込みモデルとtokenizer：Hugging Face の「setu4993/LaBSE」
- 対訳候補の検索：OTの計算において計算時間を要する ⇒

  Mean Pooling ベクトルのcosine類似度による「Faiss」で20個の対訳候補を検索
- OTの重み付け方式：文頻度

■GPU：NVIDIA RTX A6000一枚 を使用

実験設定

◆実験設定

■検索方向と範囲

- 方向：日英方向 (日本語文書に対応する英語文書を検索)
- 範囲：4つのweb-domainを混合して、全範囲で検索する

■評価指標

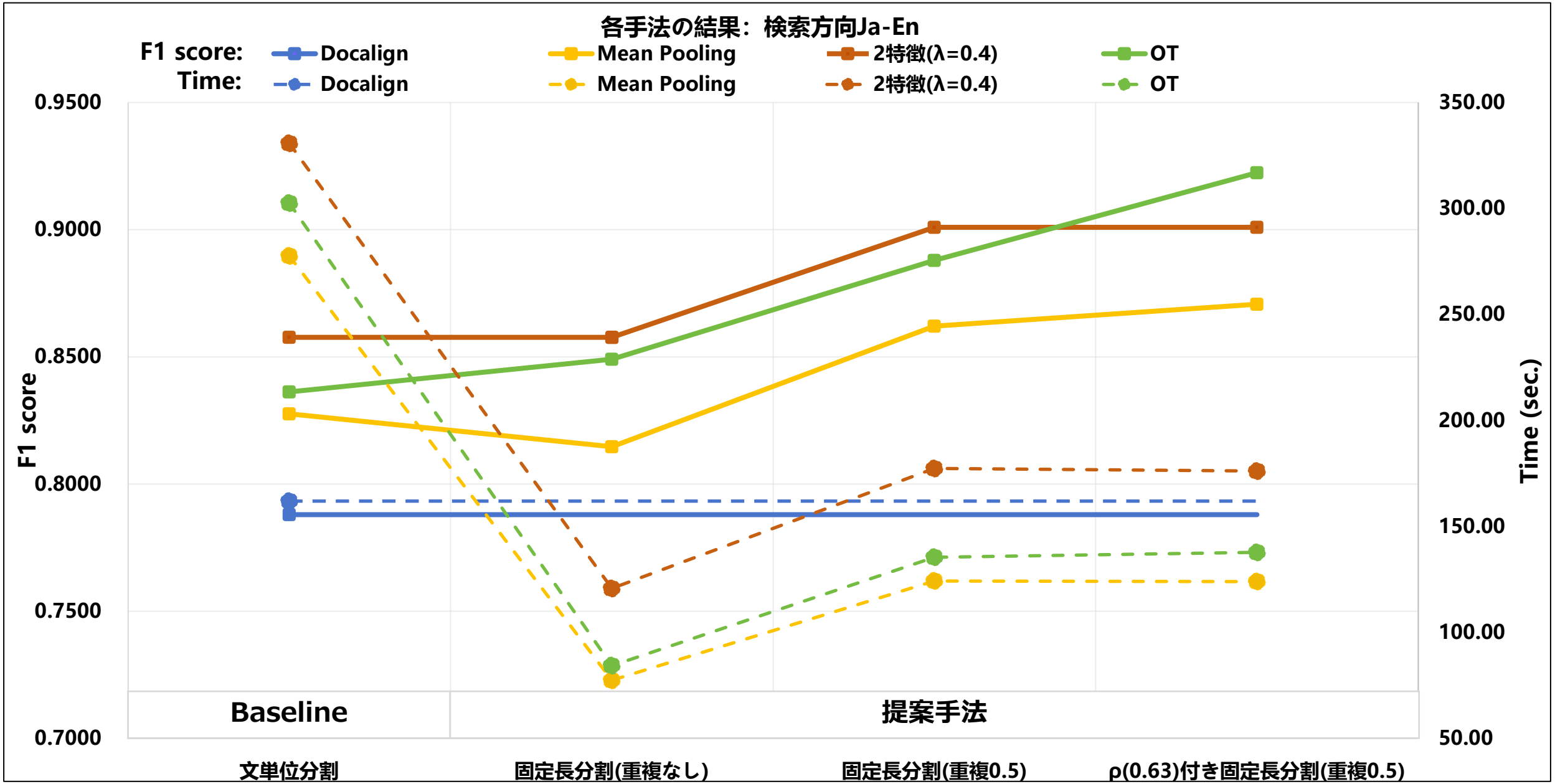
- 適合率 (Precision)
- 再現率 (Recall)
- F1_score：$\frac{2*Precision*Recall}{Precision + Recall}$

■計算時間の評価

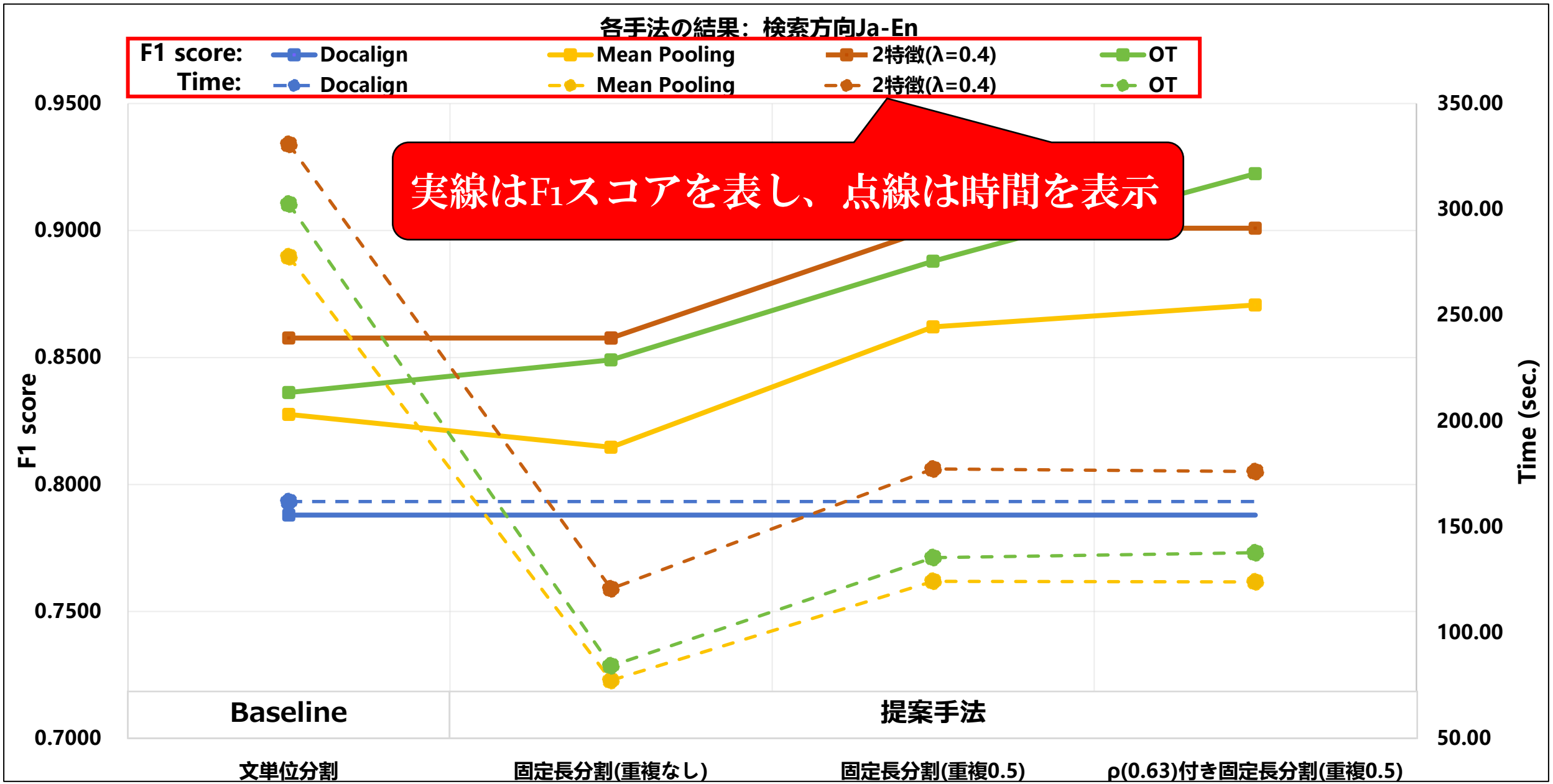
- 機械翻訳に基づく二言語間の文書対応付け
  - 機械翻訳時間
  - 「Docalign」による二言語間文書対応付け時間
- Sentence Embeddingに基づく二言語間の文書対応付け
  - 系列ベクトルの生成時間、またはそれに加えて特徴ベクトルの生成時間
  - 二言語間文書対応付け時間 (類似度の計算と候補文書の検索時間が含まれる)

目次

実験結果

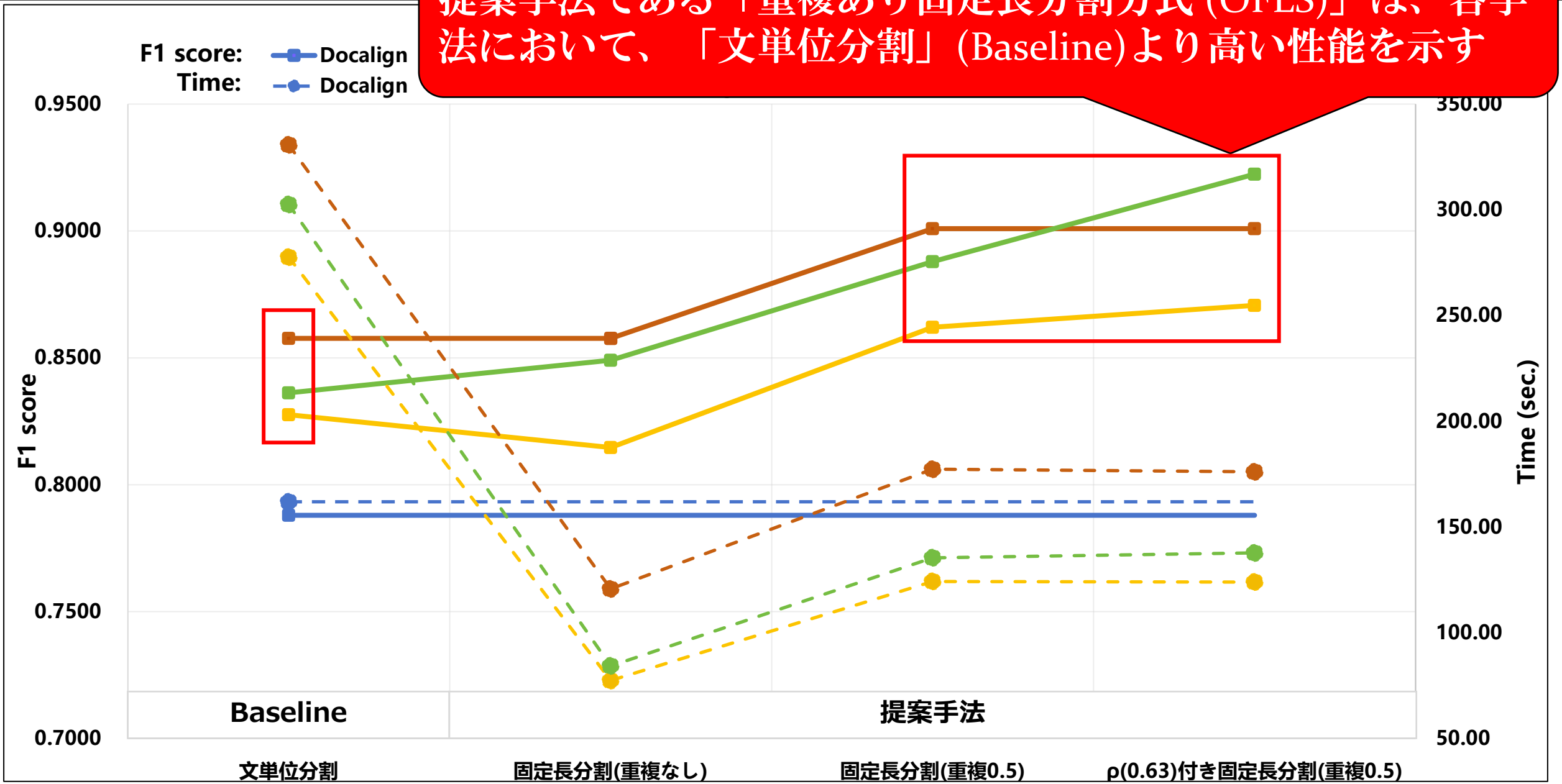

各手法の結果：検索方向Ja-En
F1 score:
Time:
Docalign
Mean Pooling
2特徴(λ=0.4)
OT
0.9500
0.9000
0.8500
0.8000
0.7500
0.7000
F1 score
350.00
300.00
250.00
200.00
150.00
100.00
50.00
Time (sec.)
Baseline
提案手法
文単位分割
固定長分割(重複なし)
固定長分割(重複0.5)
ρ(0.63)付き固定長分割(重複0.5)

実験結果


各手法の結果：検索方向Ja-En
F1 score:
Time:
Docalign
Mean Pooling
2特徴(λ=0.4)
OT
実線はF1スコアを表し、点線は時間を表示
0.9500
0.9000
0.8500
0.8000
0.7500
0.7000
F1 score
350.00
300.00
250.00
200.00
150.00
100.00
50.00
Time (sec.)
Baseline
提案手法
文単位分割
固定長分割(重複なし)
固定長分割(重複0.5)
ρ(0.63)付き固定長分割(重複0.5)

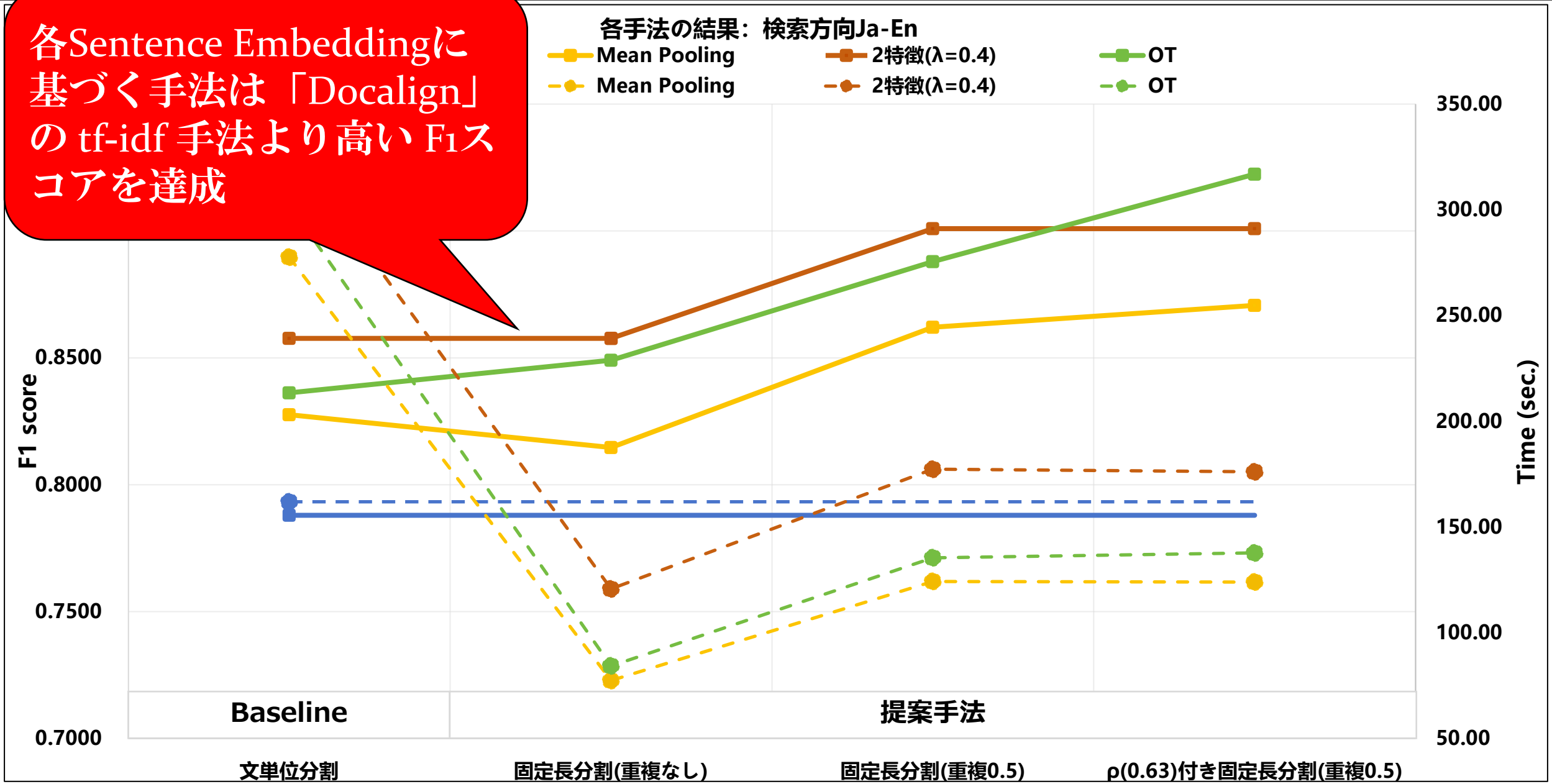
実験結果

提案手法である「重複あり固定長分割方式(OFLS)」は、各手法において、「文単位分割」(Baseline)より高い性能を示す

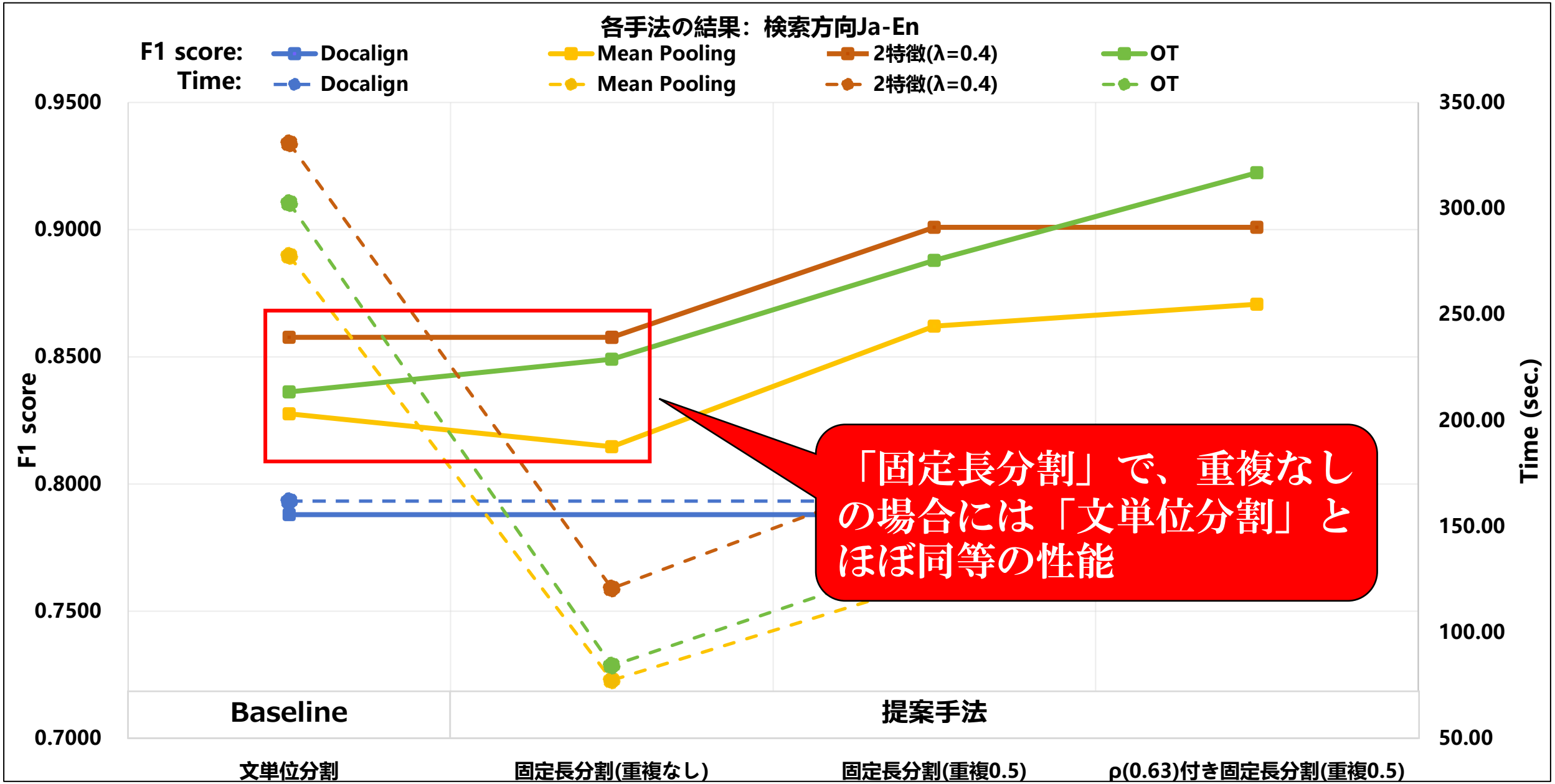


実験結果

各Sentence Embeddingに基づく手法は「Docalign」の tf-idf 手法より高い F1スコアを達成

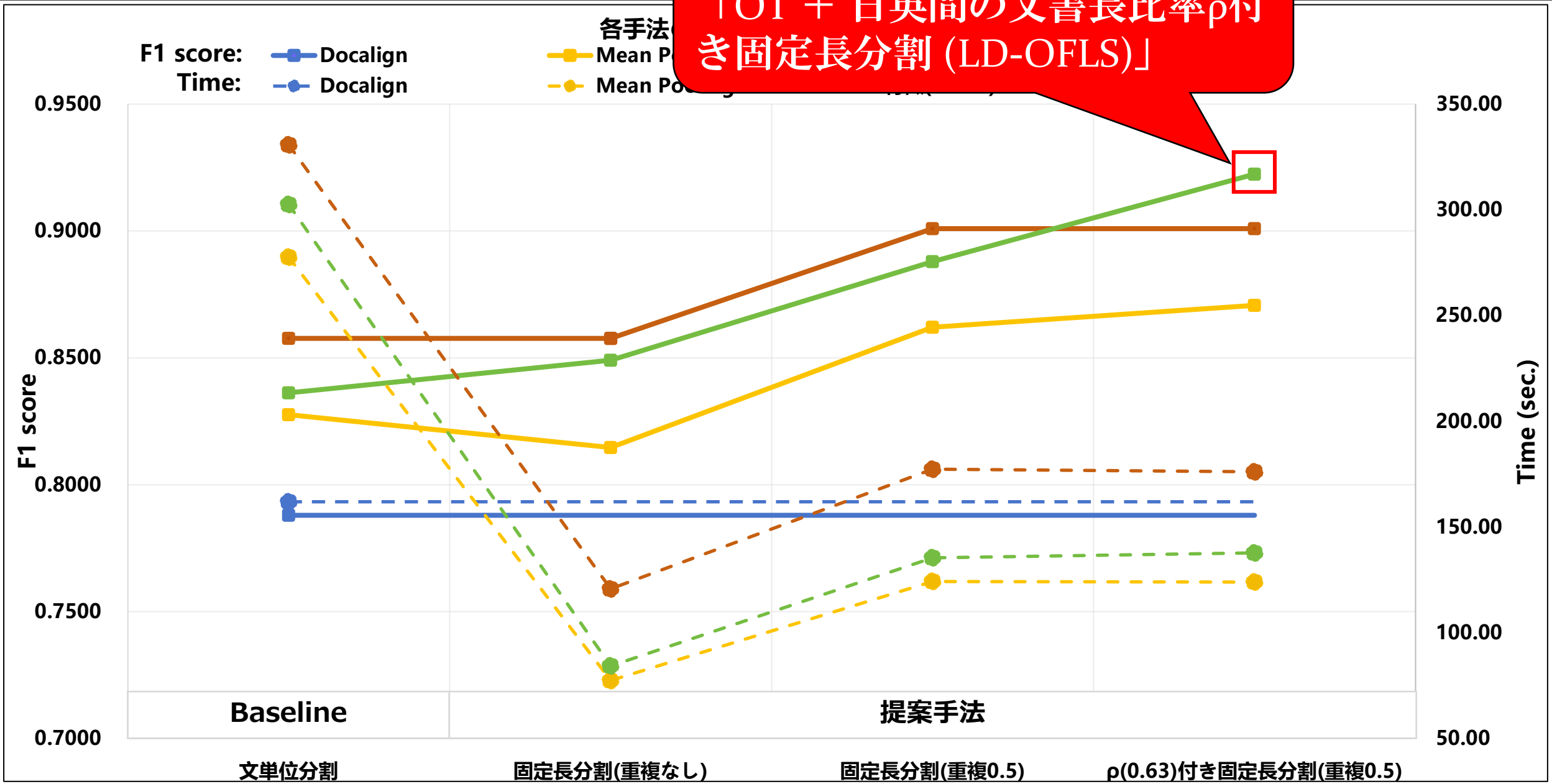


実験結果

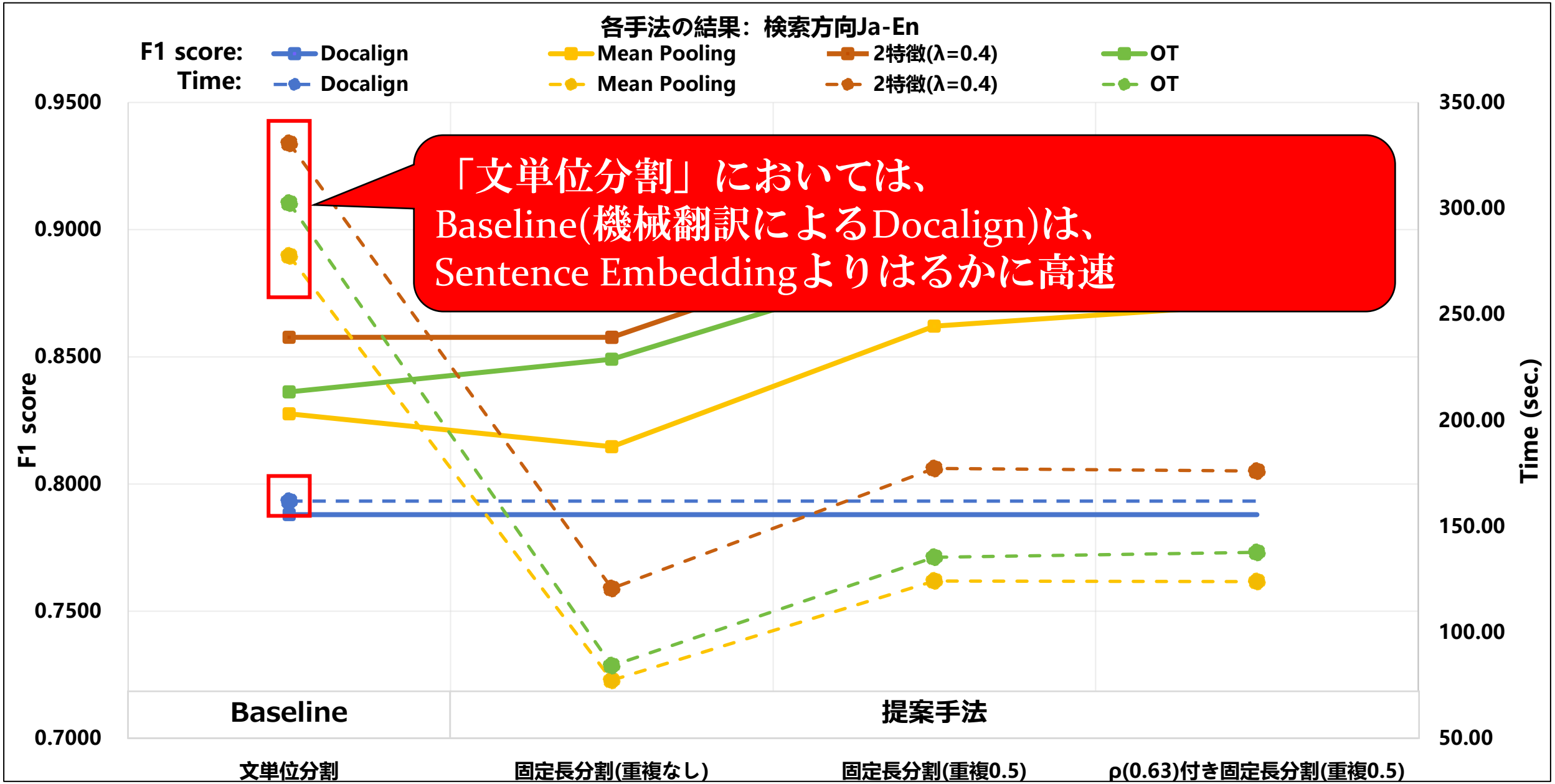

各手法の結果：検索方向Ja-En
F1 score:
Time:
Docalign
Mean Pooling
2特徴(λ=0.4)
OT
0.9500
0.9000
0.8500
0.8000
0.7500
0.7000
F1 score
350.00
300.00
250.00
200.00
150.00
100.00
50.00
Time (sec.)
「固定長分割」で、重複なしの場合には「文単位分割」とほぼ同等の性能
Baseline
提案手法
文単位分割
固定長分割(重複なし)
固定長分割(重複0.5)
ρ(0.63)付き固定長分割(重複0.5)

実験結果

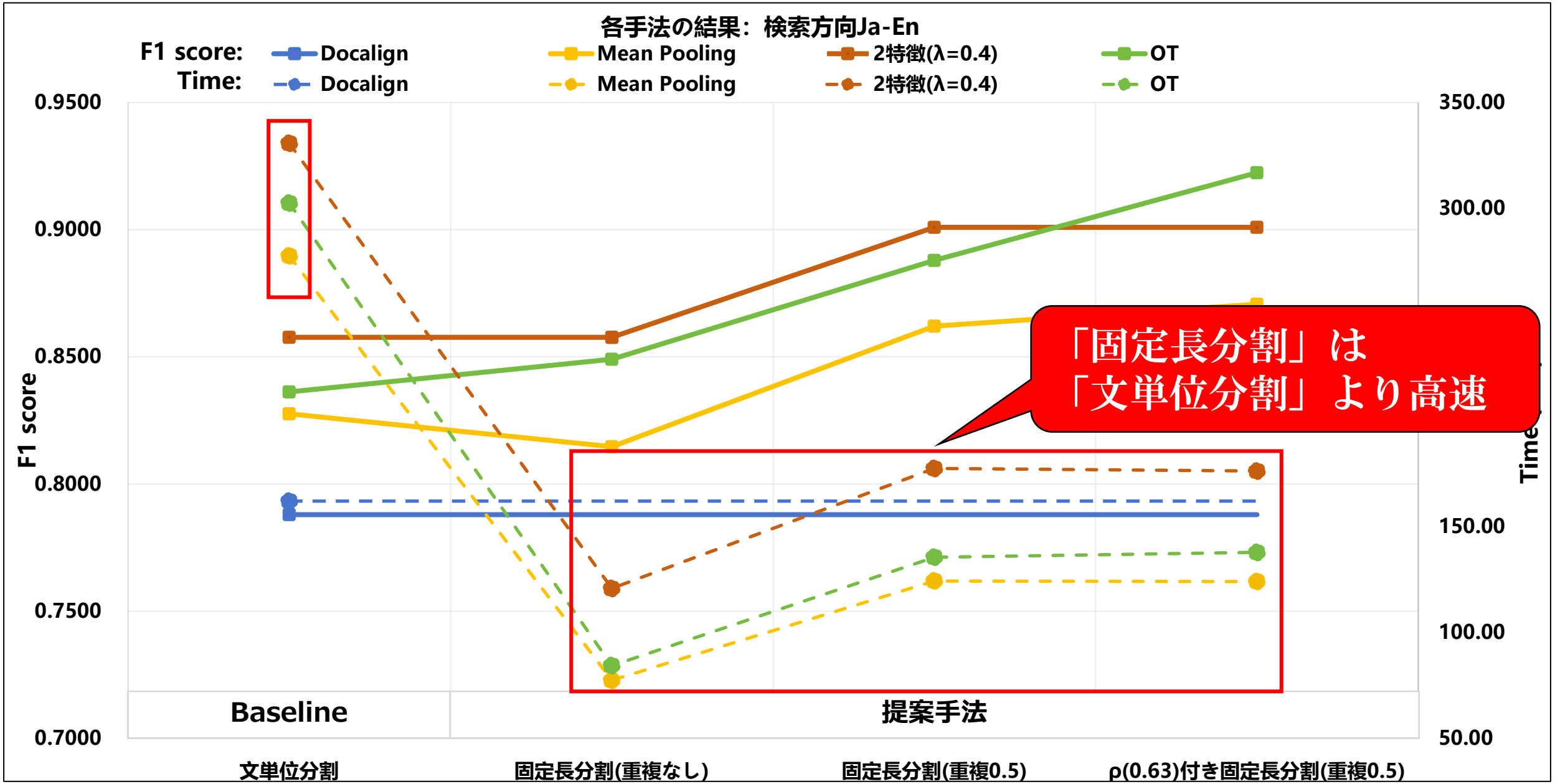
F1スコア最大は、
「OT ＋ 日英間の文書長比率ρ付き固定長分割 (LD-OFLS)」



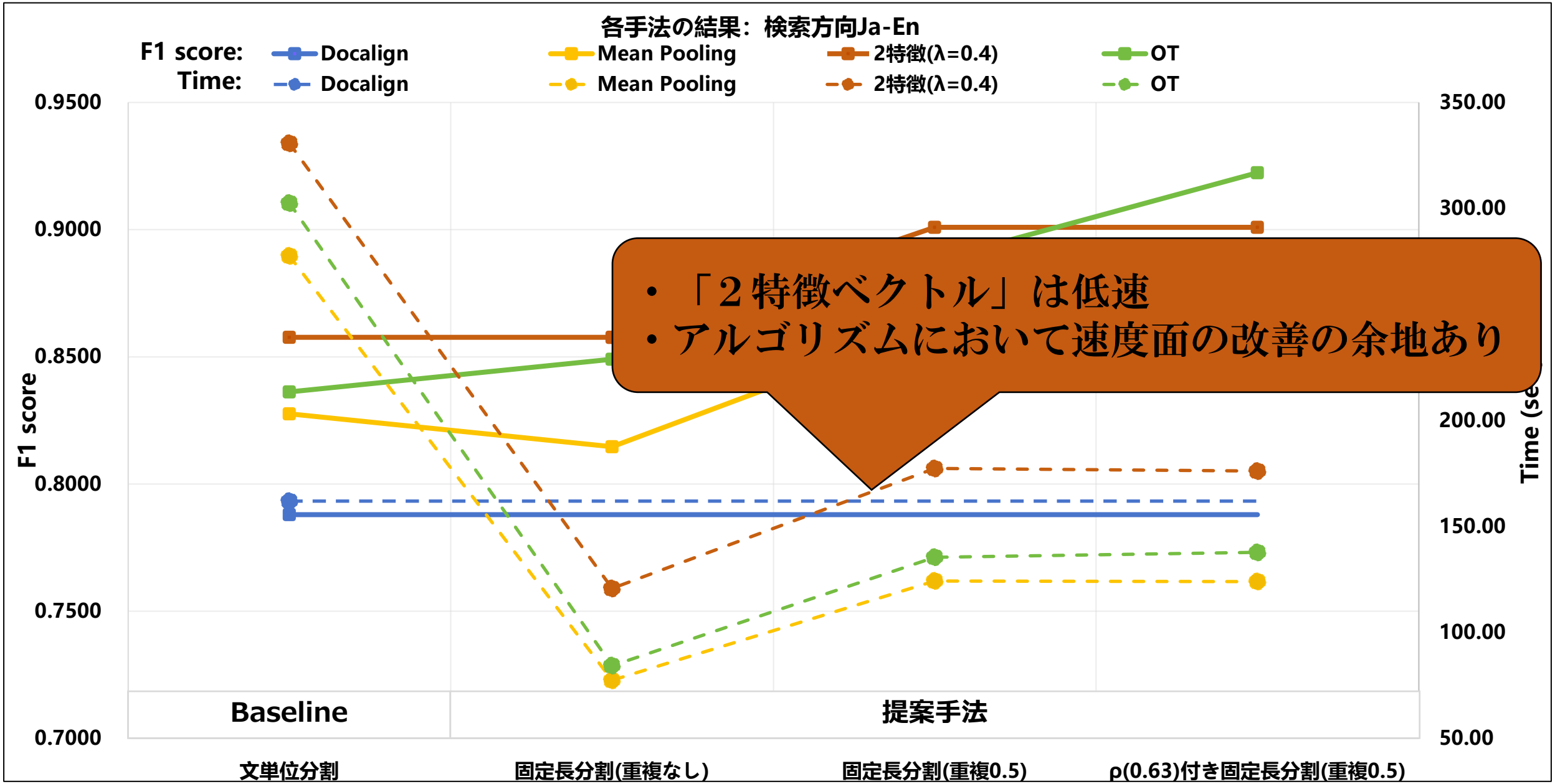
実験結果


各手法の結果：検索方向Ja-En
F1 score:
Time:
Docalign
Mean Pooling
2特徴(λ=0.4)
OT
「文単位分割」においては、
Baseline(機械翻訳によるDocalign)は、
Sentence Embeddingよりはるかに高速
F1 score
Time (sec.)
0.9500
0.9000
0.8500
0.8000
0.7500
0.7000
350.00
300.00
250.00
200.00
150.00
100.00
50.00
Baseline
提案手法
文単位分割
固定長分割(重複なし)
固定長分割(重複0.5)
ρ(0.63)付き固定長分割(重複0.5)

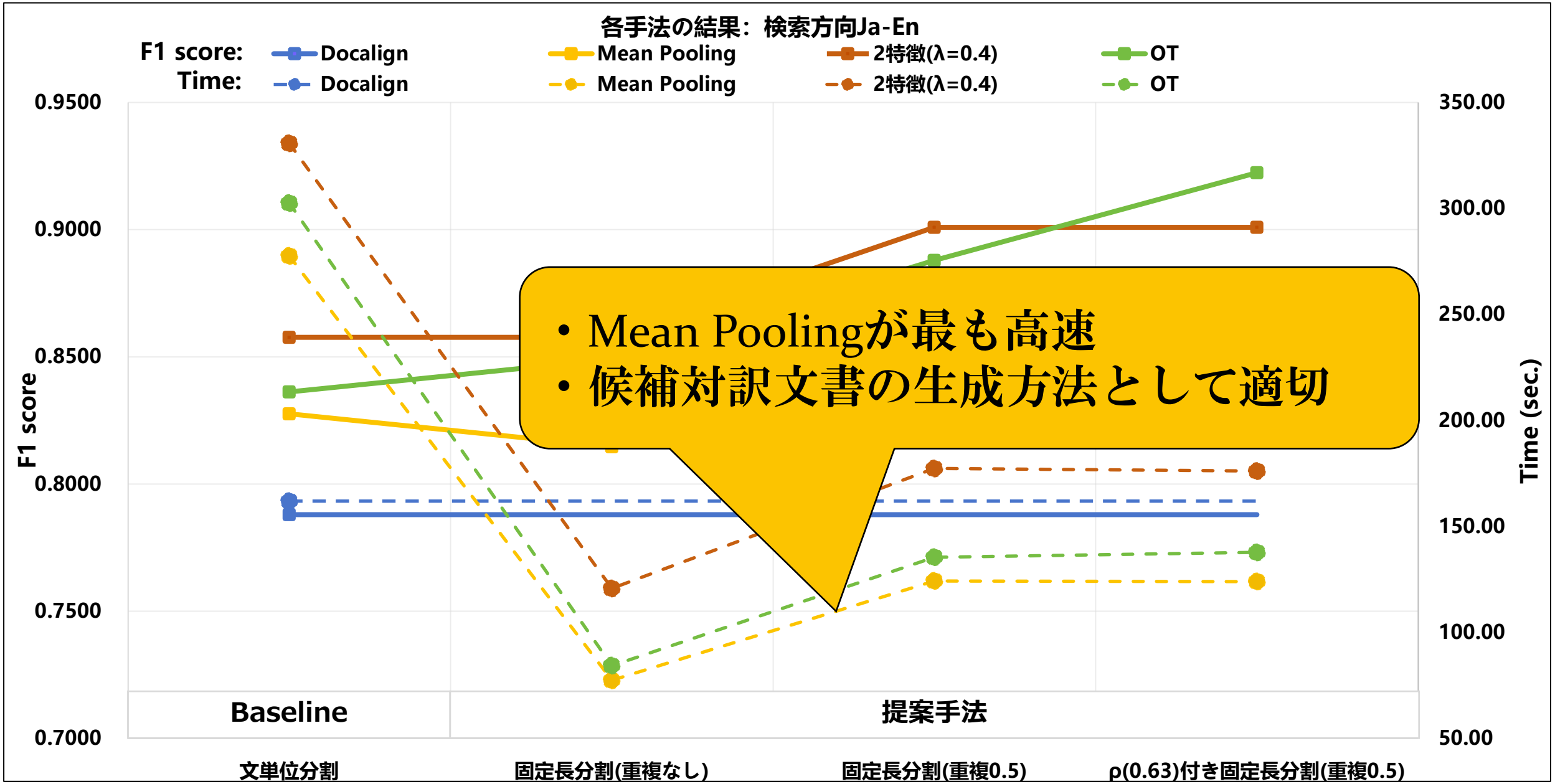
実験結果


各手法の結果：検索方向Ja-En
F1 score:
Time:
Docalign
Mean Pooling
2特徴(λ=0.4)
OT
0.9500
0.9000
0.8500
0.8000
0.7500
0.7000
F1 score
350.00
300.00
150.00
100.00
50.00
Time
「固定長分割」は
「文単位分割」より高速
Baseline
提案手法
文単位分割
固定長分割(重複なし)
固定長分割(重複0.5)
ρ(0.63)付き固定長分割(重複0.5)

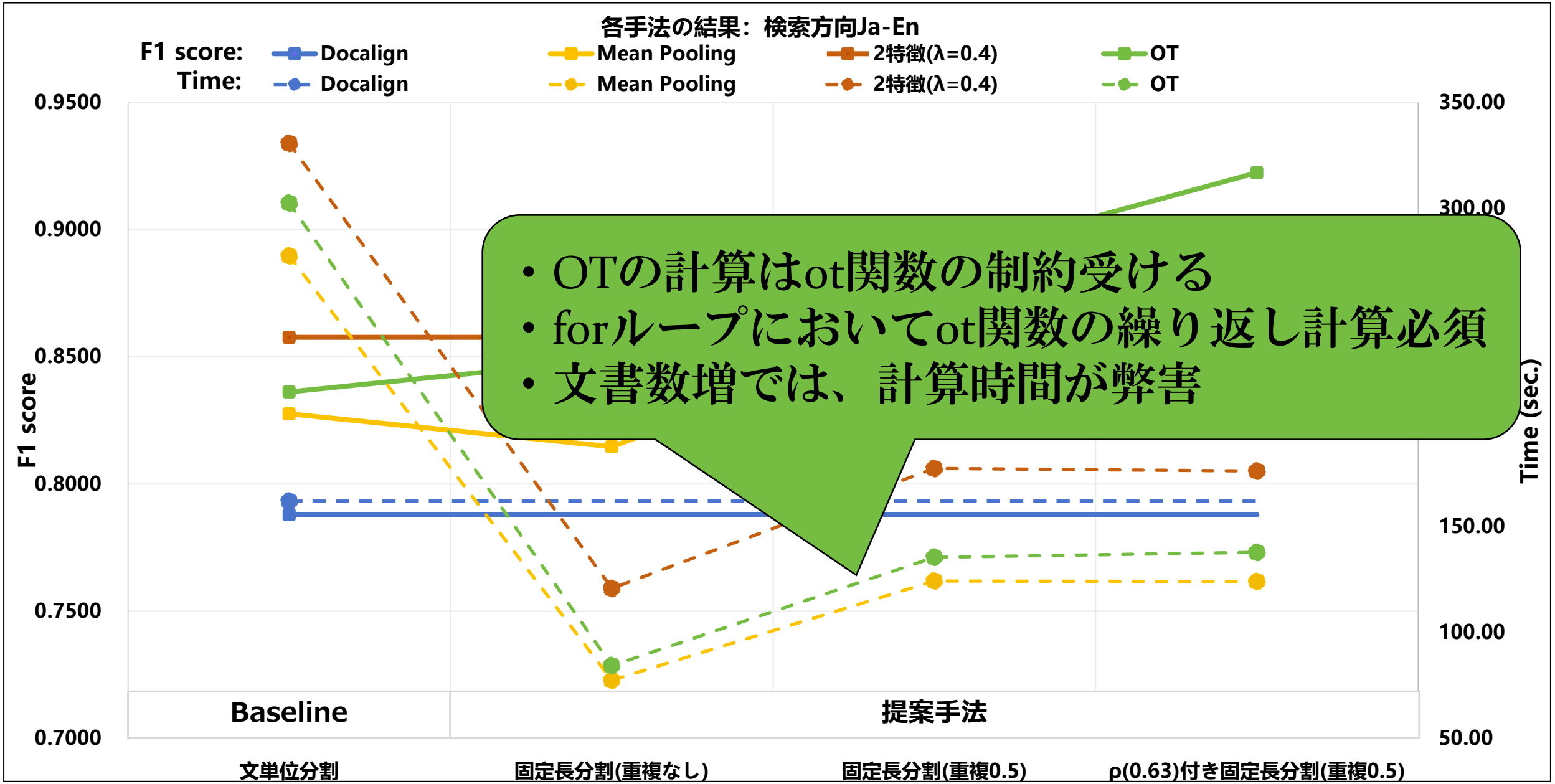
実験結果


各手法の結果：検索方向Ja-En
F1 score:
Time:
Docalign
Mean Pooling
2特徴(λ=0.4)
OT
0.9500
0.9000
0.8500
0.8000
0.7500
0.7000
F1 score
350.00
300.00
200.00
150.00
100.00
50.00
Time (se
・「２特徴ベクトル」は低速
・アルゴリズムにおいて速度面の改善の余地あり
Baseline
提案手法
文単位分割
固定長分割(重複なし)
固定長分割(重複0.5)
ρ(0.63)付き固定長分割(重複0.5)

実験結果


各手法の結果：検索方向Ja-En
F1 score:
Time:
Docalign
Mean Pooling
2特徴(λ=0.4)
OT
0.9500
0.9000
0.8500
0.8000
0.7500
0.7000
F1 score
350.00
300.00
250.00
200.00
150.00
100.00
50.00
Time (sec.)
• Mean Poolingが最も高速
• 候補対訳文書の生成方法として適切
Baseline
提案手法
文単位分割
固定長分割(重複なし)
固定長分割(重複0.5)
ρ(0.63)付き固定長分割(重複0.5)

実験結果


各手法の結果：検索方向Ja-En
F1 score:
Time:
Docalign
Mean Pooling
2特徴(λ=0.4)
OT
0.9500
0.9000
0.8500
0.8000
0.7500
0.7000
F1 score
350.00
300.00
150.00
100.00
50.00
Time (sec.)
Baseline
提案手法
文単位分割
固定長分割(重複なし)
固定長分割(重複0.5)
ρ(0.63)付き固定長分割(重複0.5)
・OTの計算はot関数の制約受ける
・forループにおいてot関数の繰り返し計算必須
・文書数増では、計算時間が弊害

目次

まとめ

◆まとめ

■提案

1. 「２特徴ベクトル」の文書対応手法：Mean Poolingベクトルだけではなく、文書の系列ベクトルの中の第１文ベクトルも使用
2. 分割方式：重複あり固定長分割 (OFLS)と言語間の文書長(トークン数)の比率ρ付き重複あり固定長分割 (LD-OFLS)

■実験結果

- 重複あり「固定長分割」は「文単位分割」の性能を上回る
- 「固定長分割」は「文単位分割」より計算速度が速い
- 重複あり「固定長分割」でSentence Embeddingに基づく文書対応手法は、Baseline (機械翻訳によるDocalign) の性能を上回る

◆今後の展望

- 既存の公開データセットを用いた評価