自然言語処理　138－8
（2000. 7. 19）

全文検索における英語接辞処理の評価

本間 咲子

（株）リコー ソフトウェア研究所
honma@src.ricoh.co.jp

英語文書の検索における接辞処理 (stemming) の検索精度への影響については、過去の研究において様々な評価実験が試みられている。TREC のようなテストコレクションを用いた評価実験では、接辞処理を行なわない場合との比較では、全体としてある程度の効果は得られるが、効果の度合はクエリーによってばらつきが大きく、著しく精度を下げる場合もあると指摘されている。索引語に対して接辞処理を適用する場合には、過度な正規化による悪影響が避けられる程度にすることが望ましい。我々は、索引語を正規化する際に適当な処理対象を検証するため、接辞処理の対象を以下の４段階に分けて、TREC-7 および TREC-8 の課題 (adhoc task) を用いて検索精度への影響を評価した。

1. 屈折形の関連付け (e.g. fertilized/fertilize)
2. 最小語幹を除く派生形の関連付け (e.g. fertilization/fertilize/*fertile)
3. 最小語幹を含む派生形の関連付け (e.g. fertilize/fertile/fertility)
4. 異表記の関連付け (e.g. fertilize/fertilise)

その結果、大幅な精度低下の殆どは最小語幹を含む派生形の関連付けの段階で生じ、その他の段階では、一部のクエリーを除いて大きな弊害は生じず、ほぼ一貫して効果が得られることを確認した。

An Evaluation of English Stemming Data in Full-Text Retrieval

HONMA Sakiko

Software Research Center, RICOH Co., Ltd.

Various studies have focused on the effect of stemming on IR tasks. Experiments using a test collection like TREC have shown that the overall improvement of stemming is not significant because its effects on independent queries are so inconsistent that the damage to some queries may cancel out the benefits to others.

When stemming indexing terms, we should avoid the risk of ill effects from overstemming. To understand the extent to which we should stem indexing terms, we conducted a set of experiments using TREC-7 and TREC-8 adhoc tasks. Targets for stemming are set in the following four steps:

1. Conflation of inflectionally related forms
2. Conflation of derivationally related forms excluding their minimal stem
3. Conflation of derivationally related forms including their minimal stem
4. Conflation of spelling variants

The result shows that most of the ill effects are caused by conflating derivational variants including their ultimate stems (step 3). It also shows that the other steps damage only a few queries and produce fairly consistent improvements.

1 はじめに

英語文書を対象とする全文検索において、同一語彙に対する表層語形の違い (e.g. fertilize / fertilizes) による検索洩れを防ぐための手法として、索引語および検索語を語幹 (stem) に相当する文字列も正規化する接辞処理 (stemming) が用いられる [1]。伝統的な手法としては、辞書的な知識を用いずに、文字列規則によって単語末尾の接辞相当文字列を削除する [2] [3] が知られている。[5] は、規則による不適切な接辞削除を辞書引きによって抑制する手法を提案している。[6] は、大規模な辞書を用いて表層語形を語幹に統一する手法をとっている。

接辞処理の検索精度への影響については、[4] が [2] [3] を対象に評価実験を行なっているが、幾つかのクエリーでは効果が見られるが、弊害も同程度に生じるため、全体としては殆ど影響がないと指摘している。一方、[6] は自身の手法を屈折形のみを対象にした場合と、派生形も対象に含めた場合に分け、更に [2] [3] とも比較して評価しているが、接辞処理を行なわない場合との比較では、いずれの手法も全体的な精度は上回るが、各手法間の優劣はクエリーによってばらつきがあり、全体として大きな差は見られないと結論している。

索引語が過剰に正規化された場合、検索時に調整することは困難である。例えば、"fertilize" と "fertility" がいずれも索引作成時に "fertile" に正規化された場合、非正規化語形による索引を別に持たない限りは、これらの3語は常に同一視されることになる。一方、索引語の正規化を行なわずに、検索時に関連語形による検索語展開を行なう場合は、統計処理による展開候補の絞り込み [7] [8] や、ユーザとの対話により不要な語を削除することが可能となるが、検索語数が増えることにより検索効率が悪化するという問題がある。つまり、あまり大きな悪影響が生じない程度に索引語を正規化し、それ以上の形態的な関連付けは検索語展開として行なうのが、現実的なアプローチであると言える。

我々は、このような前提に基づき、索引語と検索語に適用される正規化処理と、検索語にのみ適用される展開処理の2段階で形態的な関連付けを行なう接辞処理モジュールと、同モジュールで使用される辞書および規則データを開発した。各処理フェーズで用いられるデータの記述対象は、[5] [6] における実験結果の分析などを参考にし、以下のように設定した。

- 正規化データ：屈折形、および最小語幹[1]を除く派生形の関連付けを対象とする。
- 展開データ：最小語幹を含む派生形、および異表記の関連付けを対象とする。

上記の設定が妥当であるかを検証するため、TREC-7 および TREC-8 [2]の課題を用いて検索精度への影響を評価するための実験を行なった。以下、接辞処理データの概要、実験の手順と結果、影響の大きかったクエリーの分析の順に報告する。

2 接辞処理データの概要

2.1 正規化データ

正規化データは、約10万語の表層語形を見出しとする辞書と、辞書未登録語に対応するための規則（辞書見出し語形の末尾から抽出した約2000パターン）で構成される。辞書だけでなく規則を用いるのは、記述対象が生産的であり大規模文書を対象にした場合に未登録語が発生する可能性が高いこと、記述対象の規則化が比較的容易であることによる。記述内容は以下の通りである。

- 屈折語尾 -s, -ed, -ing の削除
- 接辞連続から先頭接辞への正規化

接辞連続は、生産性が高く、語幹の意味を特殊化する可能性が少ないと思われる約150パターン (e.g. -ization → -ize) を対象としている。

なお、記述対象であっても、以下に該当する場合は正規化されないよう、辞書に非変換記述（同形への変換）を設け、展開データで語幹との関連付けを行なうようにしている。

- 複数の語幹に対応する場合
  (e.g. attaches / attach,attache)
- 語幹とは独立した意味を持つ場合[3]
  (e.g. meeting / meet)

[1]「語幹＋接辞」に分析できない語。例えば、"captive" は接辞 "ive" を含むが、"capt" が語幹とは考えられないため、最小語幹扱いとする。

[2]米国の NIST(National Institute of Standards & Technology) が主催する検索システムの検索精度を競うコンテストで、1992年から毎年開催されている。公式サイトは http://trec.nist.gov/ である。

[3]正規化辞書のソースとして用いた約45,000語の単語辞書に見出しとして登録されている語形を対象としている。

2.2 展開データ

展開データは、辞書のみで構成され、約4万語の正規化語形に対して約8,000パターンの展開語形グループを記述している。派生形については、直接の派生関係にある語(e.g. advertiser / advertise)に加えて、これらの語を介して間接的に関係付けられる語も展開語形に加える(e.g. advertisement / advertise → advertiser / advertisement)。異表記展開も加えた場合の"advertise"に対する展開語形の記述は、例えば"advertise, advertisement, advertiser, advertize, advertizement, advertizer"のようになる。

3 実験手順

3.1 実験環境

TREC-7およびTREC-8[9] [10]の実験環境について、簡単に説明する。

- 検索対象文書:TREC-7,8共通で、新聞記事など分野の異なる4文書(約2GB)から構成される。
- 検索要求(トピック):検索すべき文書を自然言語で記述したもの。1トピックは、TREC-1からの通し番号であるトピックID、1フレーズ程度の短い記述であるtitle、1文程度の自然言語記述であるdescription、更に詳細な記述であるnarrativeで構成される。TREC-7はID 351-400, TREC-8はID 401-450の各50トピックで構成される。
- 評価指標:TRECからは幾つかの評価指標が提供されているが、本実験では最も一般的な平均適合率[4]を用いる。

トピックからのクエリー生成および検索には、[11]のシステムを利用した。

3.2 ベースラインクエリーの作成

ベースラインとなる非正規化クエリーは以下の手順で作成される。

1. トピックのtitleおよびdescriptionフィールドを単語単位に分割し、句読点を削除する。
2. 1で切り出された検索語から、ストップワードを削除する。ストップリストは[12]を使用しているが、description用に約60語を追加している。
3. 2で残った検索語のうち、titleフィールドに由来するものだけで構成されるクエリー(tile only)と、title, description両方に由来する検索語で構成されるクエリー(tile + desc)の2種類のクエリーを作成する。検索語が複数あるクエリーには#OR演算子を用いる。

トピックからのベースラインクエリー作成の例を以下に示す。

```
<num> Number: 377
<title> cigar smoking
<desc> Description:
Identify documents that discuss the renewed
popularity of cigar smoking.

title only:
  #or(cigar,smoking)
title+desc:
  #or(cigar,smoking,popularity,renewed)
```

3.3 段階別展開データの作成

屈折語尾正規化と接辞連続正規化の影響を独立して調べるには、本来であれば複数のインデックスを作成する必要がある。しかし、インデックスの作成および保存には、時間的、資源的コストがかかる上、クエリーの中のどの検索語が精度に影響しているのかを特定するのが困難という問題がある。このため、検索対象文書から索引語として登録される文字列を切り出し、これを元に、インデックスを正規化した場合と同一内容となる以下のような展開データを段階別に作成した。

Step1: 屈折語尾正規化
  e.g. fertilize,fertilizes,fertilized,fertilizing
Step2: Step1 + 接辞連続正規化
  e.g. Step1 + fertilization,fertilizations
Step3: Step2 + 派生形展開
  e.g. Step2 + fertile,fertility,fertilities
Step4: Step3 + 異表記展開
  e.g. Step3 + fertilise,fertilises, etc.

上記の展開データでベースラインとなる非正規化クエリーを展開し、非正規化インデックスを検索するこ

[4]再現率10ポイント刻み毎に測定した適合率の平均値。

とにより、正規化したインデックスとクエリーを用いた場合と同等の実験を行なった。なお、検索語展開には#SYN 演算子[5]を用いた。以下はStep1 の展開データによる展開の例である。

```
#or(#syn(cigar,cigars),#syn(smoking,smokings))
```

表1は、展開による検索語数の変化を段階毎に示したものである。U (=unstemmed) はベースラインクエリーを示す。

	TREC-7		TREC-8	
	平均	最大	平均	最大
title only				
U	2.44	6	2.42	4
Step1	5.98 (+145.0)	11	5.82 (+140.4)	11
Step2	6.52 (+9.03)	15	6.16 (+5.84)	11
Step3	15.0 (+130.6)	39	16.9 (+174.6)	46
Step4	15.0 (+0.26)	39	17.0 (+0.59)	46
title + desc				
U	5.90	16	6.36	16
Step1	14.4 (+145.0)	38	15.7 (+146.8)	39
Step2	16.2 (+12.3)	51	16.8 (+7.00)	40
Step3	44.9 (+176.6)	145	49.2 (+193.3)	134
Step4	44.9 (+0.13)	145	49.4 (+0.36)	136

表 1: 展開による検索語数の変化
＊( ) は上段の数値に対する増加率 (%)

Step1 と Step3 においてかなり多くの語が関連付けられており、Step4 では関連付けられる語が非常に少ないことがわかる。

なお、クエリーを正規化した上で Step3 以降の展開を行なった場合、Step4 に相当するクエリーの平均検索語数は表 2のようになる。正規化しない場合と比較して、検索語数は約 40%に抑えられる。

title only		title + desc	
TREC-7	TREC-8	TREC-7	TREC-8
6.56	7.09	18.5	20.3

表 2: 正規化した場合の Step4 の検索語数

4 結果

表 3は、各クエリーセット (TREC-7/TREC-8, title only/title + desc) における段階別の平均適合率、表 4は、各段階において 10%以上平均適合率が向上または低下したクエリーの数で、( ) はそのうち 50%以上向上または低下した数である。

	TREC-7	TREC-8
title only		
U	0.1888	0.2157
Step1	0.1941 (+2.80)	0.2504 (+16.0)
Step2	0.2011 (+3.60)	0.2527 (+0.91)
Step3	0.1944 (−3.33)	0.2504 (−0.91)
Step4	0.1948 (+0.20)	0.2525 (+0.83)
title + desc		
U	0.1821	0.1988
Step1	0.2031 (+11.5)	0.2356 (+18.5)
Step2	0.2101 (+3.44)	0.2343 (−0.55)
Step3	0.2102 (+0.04)	0.2372 (+1.23)
Step4	0.2107 (+0.23)	0.2402 (+1.26)

表 3: 段階別の平均適合率
＊( ) は上段の数値に対する改善率 (%)

	TREC-7		TREC-8	
	向上	低下	向上	低下
title only				
Step1	12 (5)	6 (1)	25 (11)	2 (0)
Step2	3 (2)	0 (0)	3 (2)	1 (0)
Step3	8 (3)	15 (3)	7 (5)	15 (3)
Step4	1 (1)	0 (0)	1 (1)	0 (0)
title + desc				
Step1	20 (8)	10 (1)	27 (15)	4 (0)
Step2	5 (1)	2 (0)	1 (0)	2 (0)
Step3	16 (8)	10 (2)	13 (6)	14 (5)
Step4	1 (1)	0 (0)	1 (1)	0 (0)

表 4: 平均適合率が 10%以上向上/低下するクエリー数
＊( ) はうち 50%以上の向上/低下

Step1 (屈折語尾正規化) による改善率は、TREC-7 の title only を除き、いずれも 10%を超えている。影響を受けるクエリーの数が多い割には、悪影響を受けるクエリーは少なく、一貫して全体的な精度アップに貢献していると言える。

Step2 (接辞連続正規化) の影響はクエリーセットによりばらついており、TREC-7 では 3%程度精度が向上しているが、TREC-8 ではあまり効果がない。影響を受けるクエリー数は Step1 に比べるとかなり少ないが、50%以上の精度低下は生じていない。

Step3 (派生形展開) は Step1 と同様に多くのクエリーに影響しているが、精度が向上するクエリー数、低下するクエリー数ともに多いため、全体としてはあま

[5]#SYN 演算子については [13] を参照されたい。

り効果がない。特に title only では悪影響が強く出ており、派生形に対する接辞処理はクエリーが短い場合に効果が高いという [6] の指摘に矛盾する結果となっている。検索語数が多い場合には、良い影響を受ける語と悪影響を受ける語が 1 クエリー中に混在し、結果として若干良い影響の方に傾いているのではないかと思われる。

Step4（異表記展開）の影響を受けるクエリーは非常に少なく、いずれも精度向上に働いているが、全体としての効果は小さい。

表 5は、Porter ステマー [2] [6] との比較である。Porter は異表記を対象にしていないため、Step3 を比較対象としている。TREC-7 では Step3 が、TREC-8 では Porter が全体としては良い結果を出している。これについては、5.4で分析する。

	TREC-7	TREC-8
title only		
Step3	0.1944	0.2504
Porter	0.1903	0.2553
title + desc		
Step3	0.2102	0.2372
Porter	0.2024	0.2439

表 5: Porter ステマーとの比較

表 6、表 7は、再現率 10 ポイント刻み毎の適合率の推移を、Step2（屈折語尾正規化&接辞連続正規化）と Step3（派生形展開）について示したものである。

いずれのクエリーセットにも共通した傾向として、Step2 は再現率が低い段階から効果が現れるが、Step3 は再現率が低い状態ではむしろマイナスに働き、再現率が高くなるにつれて効果が生じている。つまり、屈折語尾正規化と接辞連続正規化については、どのような検索をする場合にも効果が期待できるが、派生形展開については、検索洩れがないように詳細に検索する場合には、ある程度の効果が期待できるが、少ない文書で効率的に情報を得ようとする場合には、むしろ逆効果であるといえよう。

[6] [1] の実装例を電子化したものを以下のサイトから入手した。
ftp://ftp.uu.se/pub/unix/networking/wais/ir-book-sources/

Recall	U	Step2	Step3
title only			
0.00	0.7138	0.7107 (−0.43)	0.6780 (−4.60)
0.10	0.4443	0.4386 (−1.28)	0.4275 (−2.53)
0.20	0.3386	0.3414 (+0.82)	0.3226 (−5.50)
0.30	0.2765	0.2836 (+2.56)	0.2659 (−6.24)
0.40	0.2125	0.2261 (+6.40)	0.2100 (−7.12)
0.50	0.1392	0.1672 (+20.1)	0.1668 (−0.23)
0.60	0.0831	0.1098 (+32.1)	0.1092 (−0.54)
0.70	0.0590	0.0796 (+34.9)	0.0813 (+2.13)
0.80	0.0442	0.0457 (+3.39)	0.0559 (+22.3)
0.90	0.0362	0.0320 (−11.6)	0.0427 (+33.4)
1.00	0.0097	0.0080 (−17.5)	0.0075 (−6.25)
Ave.	0.1888	0.2011 (+6.51)	0.1944 (−3.33)
title + desc			
0.00	0.7501	0.7590 (+1.18)	0.7338 (−3.32)
0.10	0.4446	0.4868 (+9.49)	0.4868 (+0.00)
0.20	0.3267	0.3526 (+7.92)	0.3534 (+0.22)
0.30	0.2514	0.2758 (+9.70)	0.2702 (−2.03)
0.40	0.1934	0.2290 (+18.4)	0.2182 (−4.71)
0.50	0.1378	0.1788 (+29.7)	0.1746 (−2.34)
0.60	0.0862	0.1183 (+37.2)	0.1254 (+6.00)
0.70	0.0600	0.0902 (+50.3)	0.0986 (+9.31)
0.80	0.0383	0.0527 (+37.5)	0.0602 (+14.2)
0.90	0.0262	0.0231 (−11.8)	0.0325 (+40.6)
1.00	0.0067	0.0066 (−1.49)	0.0080 (+21.2)
Ave.	0.1821	0.2101 (+15.3)	0.2102 (+0.04)

表 6: 再現率毎の適合率 (TREC-7)
* ( ) は左側の数値に対する改善率 (%)

Recall	U	Step2	Step3
title only			
0.00	0.7323	0.7397 (+1.01)	0.7472 (+1.01)
0.10	0.4872	0.5238 (+7.51)	0.5064 (−3.32)
0.20	0.3511	0.3923 (+11.7)	0.3746 (−4.51)
0.30	0.2784	0.3299 (+18.4)	0.3247 (−1.57)
0.40	0.2027	0.2588 (+27.6)	0.2572 (−0.61)
0.50	0.1780	0.2178 (+22.3)	0.2229 (+2.34)
0.60	0.1472	0.1861 (+26.4)	0.1824 (−1.98)
0.70	0.1024	0.1442 (+40.8)	0.1432 (−0.69)
0.80	0.0787	0.1144 (+45.3)	0.1125 (−1.66)
0.90	0.0551	0.0852 (+54.6)	0.0906 (+6.33)
1.00	0.0150	0.0265 (+76.6)	0.0354 (+33.5)
Ave.	0.2157	0.2527 (+17.1)	0.2504 (−0.91)
title + desc			
0.00	0.7164	0.7069 (−1.32)	0.7136 (+0.94)
0.10	0.4214	0.4744 (+12.5)	0.4542 (−4.25)
0.20	0.3464	0.3842 (+10.9)	0.3633 (−5.43)
0.30	0.2723	0.3276 (+20.3)	0.3128 (−4.51)
0.40	0.2027	0.2633 (+29.8)	0.2659 (+0.98)
0.50	0.1674	0.2198 (+31.3)	0.2308 (+5.00)
0.60	0.1329	0.1732 (+30.3)	0.1796 (+3.69)
0.70	0.0953	0.1305 (+36.9)	0.1442 (+10.4)
0.80	0.0620	0.0860 (+38.7)	0.1088 (+26.5)
0.90	0.0276	0.0456 (+65.2)	0.0512 (+12.2)
1.00	0.0090	0.0187 (+107.7)	0.0209 (+11.7)
Ave.	0.1988	0.2343 (+17.8)	0.2372 (+1.23)

表 7: 再現率毎の適合率 (TREC-8)
* ( ) は左側の数値に対する改善率 (%)

5 分析

Step1 から Step3 の各段階[7]で特に影響が大きかったクエリーについて、また、Porter ステマーとの比較で大きな差が生じたクエリーについて分析する。取り上げるクエリーは、title only, title + desc の両方で影響が大きかったものを対象にしているが、参照している平均適合率は title only のものである。

5.1 屈折語尾正規化による影響

トピック ID	U	Step1
424	0.0071	0.1445 (+1935.2%)
379	0.3242	0.0275 (−91.5%)

424 では “suicides” に “suicide” を関連付けることにより大幅に精度が向上している。

379 では “mainstreaming” に “mainstream” を関連付けることにより精度が低下している。トピックにおける “mainstream(ing)” の意味は「障害児を普通学級に入れる」だが、名詞でもある “mainstream” を加えることで、より一般的な「主流」の意味で用いられている文書を上位にランキングしてしまうことが原因ではないかと思われる。[8]

正規化において 379 のような悪影響を避けるためには、正規化辞書における例外的な非変換記述 (2.1) を充実させることが必要であると言える。

5.2 接辞連続正規化による影響

トピック ID	Step1	Step2
358	0.0834	0.1916 (+129.7%)
417	0.3445	0.3067 (−10.9%)

358 では “fatal” に “fatally” を関連付けることにより精度が向上している。417 では “creativity” に “creative” を関連付けることにより精度が低下している。しかしながら、該当する正規化パターン (-ally → -ly, -ivity → -ity) に一般的に言えることかどうかは、今回の実験だけでは判断できない。[9]

[7] Step4 については、影響を受けるクエリーが非常に少なく、悪影響は生じなかったため省略する。

[8] [14] は、”public education” のような語句の追加が有効であると述べているが、トピック中でこれに類する語は narrative の “school” だけであり、今回のクエリー作成対象フィールドには含まれていない。

[9] 402 では、“-ly” 副詞との関係付けが、358 とは逆に精度低下の原因となっている。

5.3 派生形展開による影響

トピックID	Step2	Step3
431	0.2614	0.5701 (+118.0%)
360	0.5442	0.1291 (−76.2%)

431 では "robotic" に "robot" を関連付けることにより精度が向上している[10]。360 では "legalization" に "legal" を関連付けることにより精度が低下している。

この他に精度向上の原因となった関連付けとしては "impairment / impair"(379), "disabled / disability"(386), "production / produce"(413), "poaching / poacher"(407) などが、精度低下の原因となった関連付けとしては "territorial / territory"(357), "currency / current"(378), "tourism, tourist / tour"(438,446), "motorist / motor"(432) などが挙げられる。

"-ism, -ist" による派生形と語幹との関連付けによる悪影響が TREC-8 では目立つが、これが一般的に言えるかどうかは、今回の実験だけでは断言できない。

また、表1で示したように、派生形展開によって多くの語形が関連付けられるが、精度に大きく影響するのは、一部の展開語形に限られ、多くの展開語形はプラスにもマイナスにも殆ど影響していないことがわかった。

5.4 Porter ステマーとの比較

トピックID	Step3	Porter
374	0.4876	0.3379
417	0.0751	0.1809

374 では精度向上の原因となる "winner / win" が Porter では関連付けられない。417 では精度低下の原因となる "creative / create" は Porter では関連付けられない。また、Porter では "-ism, -ist" による派生形と語幹との関連付けは行なっていないため、TREC-8 では全体として良い結果が出ていることになる。

6 まとめ

接辞処理の対象を以下の 4 段階に分け、各段階における検索精度へ影響を TREC-7, TREC-8 の課題を用いて評価した。

Step1: 屈折語尾正規化
Step2: Step1 + 接辞連続正規化
Step3: Step2 + 派生形展開
Step4: Step3 + 異表記展開

[10] 同様に 392 では "robotics" と "robot" の関連付けにより精度が向上している。

その結果、派生形展開（最小語幹を含む派生形の関連付け）を除いて、ほぼ一貫して効果が得られることを確認した。

我々が開発した接辞処理データでは、Step2 までを正規化で、Step3 以降を検索語展開で扱っているが、これは大旨妥当であることが確認できた。更に精度を上げるためには、例外的記述の充実などのデータ整備が必要であるが、今回の実験だけでは一般的なデータ整備方針を導き出すことはできず、更に対象を広げて評価実験を行なうことが、今後の課題として挙げられる。

派生形展開については、効果が得られる場合と弊害が生じる場合のばらつきが大きく、また、再現率が低い段階では効果が得られにくい傾向があることがわかった。データ整備によって、精度をある程度上げることは可能であろうが、根本的な解決は難しいと思われる。派生形展開を有効に利用するためには、ユーザが展開を実施するか否かを指定したり、必要な展開語形だけを選択する枠組が必要であろう。また、処理効率の面からは、悪影響を及ぼす展開語形だけでなく、精度的な影響が期待できない語形はできるだけ除外することが望ましい。[7] は統計的な展開語形の絞り込みによる効率面への貢献についても言及しているが、統計処理による精度および性能向上も今後検討すべき課題である。

参考文献

[1] W.B. Frakes. Stemming algorithms. In *Information Retrieval: Data Structures and Algorithms*, eds. by W.B. Frakes and R. Baeza-Yates, pp. 131-160, 1992.

[2] M.F. Porter. An algorithm for suffix stripping. In *Program*, 14(3), pp. 130-37, 1980.

[3] J. B. Lovins. Development of a stemming algorithm. In *Mechanical Translation and Computational Linguistics*, 11, pp. 22-31, 1968.

[4] D. Harman. How effective is suffixing? In *Journal of the American Society for Information Science*, 42(1), pp. 7-15, 1991.

[5] R. Krovetz. Viewing morphology as an inference process. In *Proc. of 16th ACM SIGIR Conf.*, pp. 191-203, 1993.

[6] D.A. Hull. Stemming algorithms: a case study for detailed evaluation. In *Journal of the American Society for Information Science*, 47(1), pp. 70-84, 1996.

[7] J. Xu and B.W. Croft. Corpus-based stemming using cooccurrence of word variants. In *ACM Transactions on Information Systems*, 16(1), pp. 61-81, 1998.

[8] H. Jing. Information retrieval based on context distance and morphology. In *Proc. of 22th ACM SIGIR Conf.*, pp. 90-96, 1999.

[9] E. Voorhees and D. Harman. Overview of the Seventh Text REtrieval Conference (TREC-7). In *Proc. of TREC-7*, 1999.

[10] E. Voorhees and D. Harman. Overview of the Eighth Text REtrieval Conference (TREC-8). In *Proc. of TREC-8*, 2000.

[11] Y. Ogawa, et.al. Structuring and expanding queries in the probabilistic model In *Proc. of TREC-8*, 2000.

[12] C. Fox. A stop list for general text. In *SIGIR Forum*, 24(1-2), pp. 19-35, 1990.

[13] E.W. Brown. Fast evaluation of structured queries for information retrieval. In *Proc. of 18th ACM SIGIR Conf.*, pp. 30-38, 1995.

[14] W. Liggett. Topic by topic performance of information retrieval systems. In *Proc. of TREC-7*, 1999.