蓄積情報からの特徴語抽出に基づく自動要約・提示システムに関する研究

○松林圭　山下晃弘（東京工業高等専門学校）野中秀俊　今野陽子（株式会社調和技研）

A Research on Document Summarization and Presentation System Based on Feature Word Extraction from Stored Information

* Kei Matsubayashi, Akihiro Yamashita (National Institute of Technology, Tokyo College),
Hidetoshi Nonaka, Yohko Konno (CHOWA GIKEN Corporation)

Abstract－ In recent years, Bulletin Board System is widely used in various companies and organizations. Huge amounts of documents are accumulated in the system, and some of them may be useful for future work. On the other hand, researches on how to discover useful information from huge data are active. In this study, I developed an effective method to detect valuable information for business activities from accumulated documents. As an approach, I used automatic summarization to obtain them from accumulated documents and evaluated the effectiveness by use of the actual accumulated data of a company's BBS.

Key Words: Natural Language Processing, Automatic Summarization, Text Mining

1. 緒言

近年，企業や組織で日報や議事録，社内 SNS 等様々な用途で電子掲示板が用いられている．電子掲示板には多くの文書が蓄積されており，現在このような組織の蓄積情報に着目し形式的な知識を可視化するための研究や，特定業務における成功要因やリスク管理などのナレッジマネジメントに活用するための研究が活発である．しかし，これらの内容が多岐に渡るとともに，ニュースなどの媒体と異なり書式が整っていないため，キーワード検索では目的の文書が見つからないことも多く，重要性の高い情報がどこに含まれているか把握することが困難であった．そこで本研究ではそのような組織で蓄積された大量の日本語のテキストデータを対象として組織内の知識として有効活用することを目的に，自動要約・提示システムで利用される自動要約の手法について検討する．具体的には，ユーザは質問文とカテゴリを自動要約・提示システムに入力し，システムは該当のカテゴリと，質問文に含まれる特徴語に基づいて必要な情報を抽出し，自動要約により重要な部分のみを取り出すことで質問への応答として提示する．自動要約では特徴語に関係の強い文を優先的に採用する．本研究では実際に企業内で用いられている電子掲示板のデータを評価データとして使用し，アルゴリズムの評価を行う．最終的には社内で実用可能な自動要約・提示システムを目指す．

2. システム構成

本研究において想定している自動要約・提示システム構成を Fig.1 に示す．システムの具体的な動作として，社員がインターフェースから求めている情報を質問文，情報カテゴリ，投稿時期に分けて入力し，システムが応答として求めている情報の要約文を返すことを想定している．自動要約・提示システム内では次のような動作で要約文を生成することを想定している．(1)入力された情報を受け取る，(2)入力情報から質問文，カテゴリ情報を用いて特徴語の抽出を行う，(3)特徴語，カテゴリ，投稿時期を基に該当するデータを検索する，(4)特徴語と該当する投稿データを用いて自動要約を行なって応答文を生成する．本研究で構築した自動要約・提示システムのプロトタイプを Fig.2 に示す．

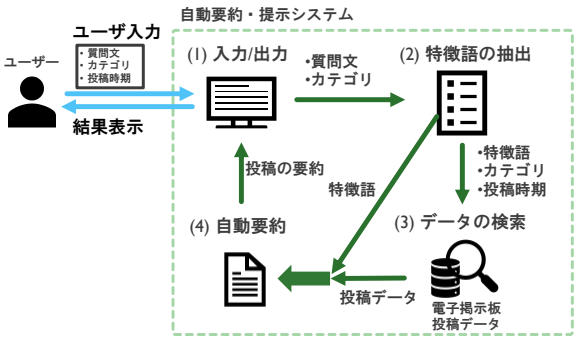

Fig.1 システム構成

Fig.2 入力画面(左)と結果画面(右)

ユーザはウェブインターフェースからデータを入力し，検索ボタンを押すと該当する文書の件数とその要約文がリストとして表示されるようになっている．

3. 要約手法の分類

自動要約の手法を検討するにあたって，自動要約についての調査論文[1]を元に，本研究で取り扱う自動要約について明確にする．

まず，自動要約タスクの目的とは一般的に，自動要約の目的として「指示的要約」と「報知的要約」に大別される．指示的要約は元の文書を読むべきかどうかを判断するための要約を指し，必要に応じて本文を読むことを前提としている．例としては新聞記事の見出しの一文が挙げられる．一方，放置的要約では，元の文書の概要を伝える要約で，元の文書を代替する要約

となっている．例としてはテレビニュースの字幕が挙げられる．

次に，要約を行う対象文書が単一文書か複数文書かに分けられる．単一文書としてはニュースサイトなどの一記事が挙げられる．複数文書の例としては掲示板の一つのトピックに対する投稿群の要約が挙げられる．本研究でも掲示板を対象に自動要約を行うが，対象としている企業内掲示板のデータは一つ一つが営業の事例としてまとまっており，100文字から1000文字程度の内容であるため，ニュースサイトなどと同様に単一文書要約として，個々の投稿に対して要約を行う．

また，要約を行う手法として，要約対象の文書の中から重要と思われる文を抽出する「抽出的要約」と文意を汲み取り抽象化した上で適切な要約を生成する「生成的要約」が存在する．後者の生成的要約に関しては意味表現と生成文の生合成に問題があり，良質な要約文生成が非常に困難であるため，本研究では抽出的要約を行う．

そのほか要約文の生成に関わってくるものとして，ユーザなどが指定したクエリを元に要約文を生成する「クエリ依存型要約」とクエリを用いない一般的な要約である「クエリ非依存型要約」があり，本研究ではユーザから与えられた質問文から特徴語を抽出し，それをクエリとして用いて自動要約を行うため，クエリ依存型要約を行う．

具体的な自動要約アルゴリズムについては次章にて説明する．

4. 自動要約アルゴリズムについて

本研究では文書分割を行うため，以下の 4 つの処理によって要約文の生成を行う．

4.1. 文書分割

文書分割では日本語の句点“。．”や改行で文分割を行う手法と，JUMAN[2]を用いて形態素解析を行いその結果を元に文の分割を行う手法の 2 通りを用いた．理由としては句点で分割する方が高速で実用的である点，実験の際に精度の比較対象とするアルゴリズムで JUMAN を使用する点から 2 つの分割方法を採用している．

4.2. 重要度計算

本研究で取り扱うアルゴリズムでは(a)形態素解析，(b)重要度計算，(c)クエリ考慮，(d)冗長性削減の手順で重要度の計算を行う．

まず，(a)で文の特徴量を計算するために MeCab[3]と新語辞書の Neologd[4]を用いて形態素解析を行い，その中で名詞の文のみを特徴量として用いる

次に(b)では分割された名詞を用いて重要度計算のアルゴリズムである Continuous LexRank[5]を用いて文ごとに重要度計算を行う．LexRank は初期のウェブ検索システムのアルゴリズムに使用されていた PageLank を元に作成されたアルゴリズムで，文に含まれる単語から TF-IDF によって文毎の類似度を計算し，ノードを文，類似度をエッジとしてグラフ構造を作成することで重要度計算を行う．

そして(c)ではクエリとして与えられる名詞群を正解文として考え，各文との COS 類似度を計算し，計算結果に報酬として与えることでクエリ考慮を行う．式(1)に計算式を示す．S_iは i 番目の文，Q はクエリ，w は重みを示しており，w を変更することによってクエリ考慮の度合いを調整する．

$$Score = LexRank(S_i) + wCosDist(Q, S_i) \quad (1)$$

最後に計算された文に対して，同じような意味の文を避け，文意の網羅性を上げるために(d)にて MMR(Maximal Marginal Relevance)[6]を適用する．MMR では既に重要文として選択された文と類似している重要文の候補に対してペナルティを与えることによって文意の網羅性向上させるものであり，式(2)のようにして計算される．S_iは i 番目の文，λは重みを示しており，本研究では 0.7 に固定して実験を行なう．

$$MMR = \arg\max_{D_i \in R \setminus S} \left[\lambda Sim_1(S_i) - (1-\lambda) \max_{D_j \in S} Sim_2(S_i, S_j) \right] \quad (2)$$

4.3. 要約文の選択

計算された重要度に対して，どの文を要約文として指定するか選択を行う必要がある．一般的には，閾値を設定しその値を上回る文を用いる場合と，特定の文字数に達するまで文を選択し続ける場合の 2 つが考えられる，本研究では実用性の点から後者を用いる．

4.4. 出力

最後に用いる要約文の集合を本来の文の並び順にソートして出力を行うことによって全体として意味の通りやすい文書にし，それを要約文として出力する．

5. 予備実験

本研究における自動要約アルゴリズムのクエリ考慮における特性を調査するため，いくつかの文書を対象として自動要約を実施した．対象文書は 2018 年の情報処理学会全国大会論文誌中から異なる分野の論文 7 編に対して人手で特定のトピックに関する正解文，クエリを論文の本文内から抽出したものを用いる．Table 3 に実際に用いた論文の ID，クエリ，正解文の一部を示す．クエリ考慮の特性に関しては，クエリの重み w をクエリ考慮を行わない 0.0 からクエリ考慮を行う 1.0 まで 0.1 刻みで変化させていくことで変化を測定する．また，評価に関しては自動要約タスクにおいて一般的に用いられている，単語単位での正解文との一致度を示す指標である ROUGE-1 および ROUGE-2 を用いた．

実験結果を Fig. 3 および Fig. 4 に示す．ROUGE-1，ROUGE-2 の結果では値が異なるものの，どちらも同様の傾向が現れた．対象とする文書や正解文とクエリのセットにより 6D-05 のように最初から対象文が抽出

Table 1 対象文書および正解文，クエリ

論文ID	クエリ	正解文
1ZC-04	通信	本研究では，通信環境が劣悪な地域において車載センサ情報を共有する通信システムを実現するため，複数の無線通信を組み合わせて通信状態に応じて情報共有に用いる無線通信を選択する…
1ZD-03	障害物	本研究では，HoloLensを応用した視覚障がい者向けナビシステムについて検討し，HoloLensを用いてリアルタイムに障害物を検出・回避できることを示す実験を行った．HoloLensによる環境スキャンは…
3ZD-02	危険	本研究では，OpenCVを用いて地図画像を分析することで，高齢者にとって危険なシチュエーションを検出する高齢者見守り支援システムを検討した．検証の結果，危険シチュエーションの検出精度におい…
5R-06	IoT	図に示されるように、エッジサーバ環境の場合の性能が非常に高く、書き込みでは単一サーバ、レプリケーション環境と比べてそれぞれ11倍と14倍であり、読み出しでは21倍と20倍であった。また、図3に示すように、書き込みでは単一サーバ環境の方がレプリケーション環境…
6D-05	サービス	グループ演習授業において，グループ別の進捗をモニタリングしファシリテーター・メンター間の連携を支援するサービス(PIMS)のプロトタイプ開発を行なった．実際にGoogleクラウドサービスとGAS，MESHに…
1F-02	人事	今回のプログラムは，オープンソースを用いたサーバにおける不正侵入の検知および防止に関する内容により，受講者のセキュリティ対策における技術的向上について，一定の成果があったと思われる．
5P-01	システム	与えられた雑多な文書から自動要約を行うシステムを構築するため，LexRankアルゴリズムを用いた．特徴抽出ではTF-IDFとFastTextによる特徴を用いて，MMRによる冗長性の排除やクエリと類似する文…

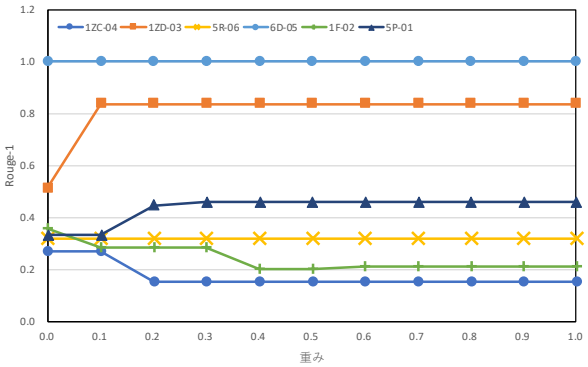

Fig. 3 対象文書における ROUGE-1

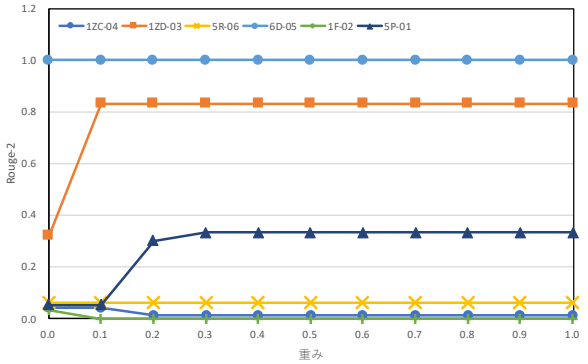

Fig. 4 対象文書における ROUGE-2

できている場合，5R-06 のようにクエリに依らず同じ要約結果になり続ける場合，1ZD-03 や 1F-02 のように重みを増やすことによって正解文に近づく場合と離れる場合が存在することがわかった．また，正解文に近づくもしくは離れる場合に重みがどれくらい関わってくるかは対象文書，正解文，クエリによって変わってくることから，実運用する際にはそれらを考慮し手動で重みの調整や，自動要約・提示システム内で動的に変化させるなどの工夫が必要なことが明らかになった．

6. 評価実験

自動要約の精度を検証するため，本研究で実装を行った LexRank とオープンソースソフトウェアの Shuca を本研究において適用したアルゴリズムに関して要約精度の比較実験を行った．

6.1. Shuca について

Shuca[7]は西川氏によって開発されたオープンソースの自動要約プログラムで，形態素解析エンジンの JUMAN の出力を KNP でパースしたデータを基に重要文の計算を行うプログラムである．Shuca 内部では自動要約のタスクをナップザック問題として捉える．ナップザック問題は NP 困難であり，それを高速で計算するために要約に含めるべき単語数を直接制御する冗長性制約付きナップザック問題に基づく要約モデルを実装している．また，ナップザック問題を解く際には予め KNP の出力結果から特定の単語や名詞に対して重み付けを行った上で計算を行っている．そのため Shuca は比較的ヒューリスティックであると言える．

6.2. 対象データ

本研究では，実際に企業内に蓄積されている 36,000 件のアルバイト求人情報に関する営業データから人手で 183 件のクエリと正解データのペアを作成し，そのデータの正規化を行なった上で実験に用いた．具体的には正解文の単語数が 2 単語から 92 単語まで大幅な開きがあったため，中央値から上下 5 割の単語数である 8.5 単語以上 25.5 単語未満の正解文のデータを実験に用いることにした．Table 2 に人手で抽出した要約例を示す．また，自動要約を行う際にはあらかじめ URL や色情報などのメタタグは全て取り除いた上で用いた．

6.3. 評価方法

自動要約タスクの一般的な評価指標として内容，読みやすさ，外的評価が挙げられる．そのため本システムの評価点するポイントとして，(1)要約内容が投稿原文に対して適当であるか，(2) ユーザの意図を汲み取れているか，(3)システム全体として合理的であるかが考えられる．以上を考慮し本研究では以下の 3 つについて評価を行う．

(A) 投稿原文に対する要約文の妥当性
(B) 特徴語に応じて要約内容が調整されているか
(C) 重要で有益な情報が提示できているか

上記を評価するため，(A)については本実験では人手で抽出した要約文が含まれているかどうかを正解数(正解率)として求め，(B)については実際に LexRank 内の重みを調整することで要約内容が変化するかの確認，(C)については自動要約によって求められた文が全体としてどれくらい正解文に近いかを ROUGE-N を用いて検証を行う．また，実験で用いる文分割アルゴリズムに関しては JUMAN のデータを用いることで LexRank と Shuca のアルゴリズムにおいて同一の入力による評価を行う．JUMAN の出力は実行時間の観点から予め処理を行い実験に用いる．その理由として，JUMAN の実行時間が非常に遅く，100 件以上のデータを形態素解析するのに現実的な時間で終了せず，最終的にシステムとして逐次計算することが困難であることが挙げられる．そして，実用性の観点から LexRank においては正規表現によって分割された文も利用できるため，(B)に関する実験の際には JUMAN および KNP における比較も同時に実施した．

Table 2 人手で抽出した要約例

文書ID	原文	キーワード	重要文(Aさん)
16341	<OPUS2関連部署のみなさまへ> 年末の休業が入るため、OPUS2の請求書出力を一部前倒しで行います。 12/27～31請求日分→12/27（木）に行います。 ※発送は年内にします。 12/28～31が請求日になるものは、 この段階ではまだ請求金額が未確定のものがある可能性も…	OPUS2	年末の休業が入るため、OPUS2の請求書出力を一部前倒しで行います。
27562	ここの所、読者さんから 履歴書の書き方について等の質問電話が数件入りました。 昨日は、けっこう年配風の男性より 写真付履歴書と、履歴書の違い　について聞かれました。 内心、"こんな事聞いてくるこの方って・・"と思いましたが どっちにしても写真は貼った方がいいと話すと、 採用にならなくても履歴書かえってくる事まずないし、…	必死	読者も必死になってるのが伝わってきます。
28414	リニューアルポイント ・メール会員登録がスタートします ・トップページデザインが変更されます ・P6以上の画面表示・リストからの導線が変更になります ・P18（3画面）商品が新設されます <メール会員登録> ユーザーは資格、職種・勤務地、フリーワードのいずれかから検索条件を設定し、合致した求人情報が月曜11：00にメールで届くシステムです。会員数と登録された情報は、営業ツールに使えるものとしてフィードバックします。	リニューアル	リニューアルポイント
…	…	…	…

Table 3 実験条件 1

条件	設定
対象文書数	107件
文字数	150文字以内
MMR	0.7
分割方法	JUMAN(KNP)・正規表現
クエリ考慮の重みw	0.0から1.0まで

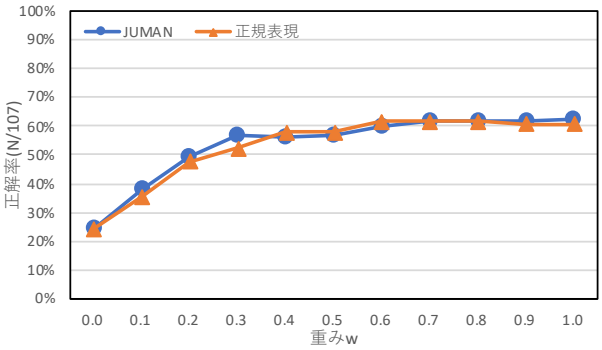

Fig. 5 重みによる正解率の変化について

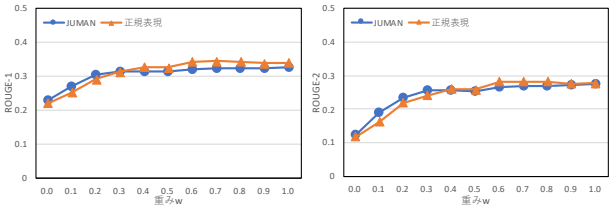

Fig. 6 重みによる ROUGE-1(左)と ROUGE-2(右)の変化について

7. 実験結果

7.1. 重みによる変化について

重みにおける自動要約結果の変化について，Table 3 のように条件を設定した上で実施した．実験結果を Fig.5 および Fig.6 に示す．

JUMAN 及び正規表現のどちらの場合においても重みと正解率に関して同様の傾向が見られた．また，文書分割の精度については 0.3 までは JUMAN で分割を行った方が僅かに良い結果が得られたが，0.4 から 1.0 までに関してはほとんど差が見られない結果となった．よって，どちらの分割方法を用いても実際の掲示板のデータにおいても重みを調整することで要約結果を変化させ，また，本研究で対象としている文書に関してはどちらの分割方法においても重みを増大させることで要約精度を向上させることができていることが確認された．

7.2. アルゴリズム比較結果

LexRank と Shuca に関して制限された文字数ごとに要約精度がどのように変化するのか調査するため，Table 4 のように条件を設定し，文字数を 10 文字から 200 文字まで 10 文字刻みで増加させ，実験を行なった．

実験結果を Fig. 7，Fig. 8，Fig. 9 に示す．なお，ROUGE-N の計算の際には結果の標準偏差をエラーバーとして示している．

正解率において Shuca は LexRank と比較していずれの文字数の要約文書に対しても Shuca の方が 1 割ほど良い結果となり，特に Shuca では 8 割近い正解率となった．また，ROUGE-N に関してはどちらも誤差が大きいものの，50 文字付近が一番良い結果となった．

ROUGE-N に関して Shuca も LexRank も同じような値が出力されたことから，LexRank は正解数が少ないものの，正解文に近い文は Shuca 同様に抽出できていることが確認できた．しかし，標準偏差においても 50 文字付近が最も大きい結果となったことから正常に正解が抽出できているケースと全くできていないケースで差が明確に出ることがわかった．

以上の結果よりどちらのアルゴリズムであっても ROUGE が最大でかつ正解数の増加率が減少する 70 文

Table 4 実験条件 2

条件	設定
対象文書	107件
文字数	10～200文字，10文字刻み
MMR	0.7
分割方法	JUMAN
クエリ考慮の重みw(LexRank)	1.0

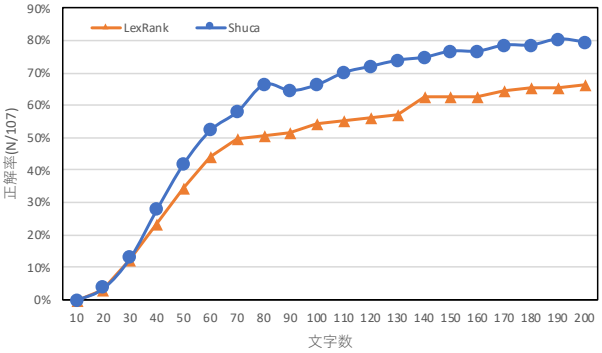

Fig. 7 要約文字数を変化させた時の正解率

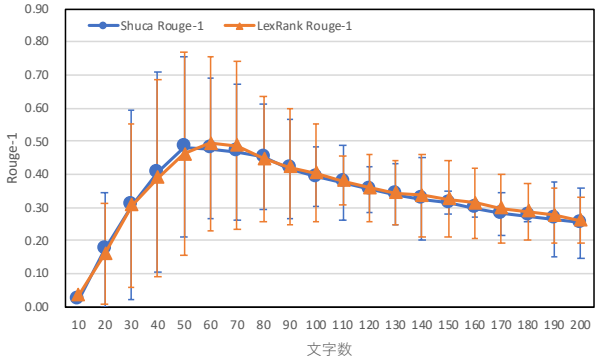

Fig. 8 要約文字数を変化させた時の ROUGE-1

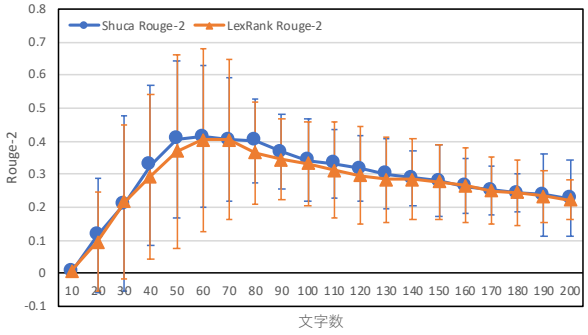

Fig. 9 要約文字数を変化させた時の ROUGE-2

字付近であると正解率も ROUGE-N も考慮された要約が生成できると言える．それと同時に文書によって要約精度が大きく異なることから，重みを動的に調整するなどの改良の余地があると言える．なお，自動要約における最適な要約文の文字数に関して，事前にデータ提供元の社員 10 人へのヒヤリングにより 100 文字程度が最も見やすいという結果に集約されており [8]，最も良い ROUGE-N スコアになる要約においても 100 文字を下回っている．また，100 文字で要約を行なった場合でも 6 割から 7 割程度の正解率が出せることから，実際の企業で応用する際にも十分実用に耐えうると考えられる．

また，Shuca と LexRank は実行時間と精度がトレードオフになっており，大量のデータを高速に処理する場合には正規表現のみで分割実行でき，高速である LexRank の方が優れているが，自動要約の精度を求める場合に関しては，速度の関係で事前処理などを行う必要があるが，Shuca の方が優れていると言える．

8. 結論

本研究では企業内で蓄積されてきた情報に対して知識の形式化などのナレッジマネジメントを行うため，自動要約・提示システムに関する研究を行なった．本研究ではその中でも自動要約アルゴリズムに関して扱い，LexRank を用いたクエリ依存型要約アルゴリズムを提案した．要約精度の評価に関してはオープンソースソフトウェアである Shuca を本研究に適用し，比較を行い，どちらの場合においても 6 割以上の精度が得られた．そのことから実際の企業の掲示板データにも適用できる可能性が十分にあることが明らかになった．

今後は実際に導入およびフィードバックを収集しアルゴリズムの改善を行なっていく必要があると言える．

参考文献

1) Horacio Saggion, Thierry Poibeau : Automatic Text Summarization: Past, Present and Future, Multilingual Information Extractionand Summarization(2012)
2) 黒橋・河原研究室:日本語形態素解析システム JUMAN, [オンライン]. Available: http://nlp.ist.i.kyoto-u.ac.jp/index.php?JUMAN. [アクセス日: 22 1 2019].
3) Taku Kudo, Kaoru Yamamoto, Yuji Matsumoto :Applying Conditional Random Fields to Japanese Morphological Analysis, Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP-2004)
4) 佐藤敏紀, 橋本泰一, 奥村学 : 単語分かち書き辞書 mecab-ipadic-NEologd の実装と情報検索における効果的な使用方法の検討, 言語処理学会 (2017)
5) Günes Erkan, Dragomir R. Radev : “LexRank: Graph-based Lexical Centrality As Salience in Text Summarization,” Journal of Artificial Intelligence Research 22 (2004)
6) Jaime Carbonell, Jade Goldstein : The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries, Porc. Of the 21th Annual International ACM SIGIR Conference on Research and Development in the Information Retrieval, pp.335-336 (1998)
7) Hitoshi, NISHIKAWA :Shuca, [オンライン]. Available: https://github.com/hitoshin/shuca. [アクセス日: 22 1 2019].
8) Yohko Konno, Nakamura Takuya, Masayuki Yoshida, Hidenori Kawamura : 営業活動の蓄積情報を利用した自動要約・知識提示に関する研究, The 32nd Annual Conference of the Japanese Society for Artifical Intelligence(2018)