6Q-08

ウェブ上の文書を要約表示する検索エンジンの開発

鷲見　有貴，長谷川　明生(中京大学)

On a web search engine with summarization

Yuki Sumi, Akiumi Hasegawa (Chukyo University)

1．まえがき

自分用のウェブサイトを作ることが容易になったことにより，インターネット上には多くのウェブサイトが存在する．膨大に存在するウェブサイトの中から知りたい情報が書かれているウェブサイトを短時間で探すことは Google のような高機能の検索エンジンを利用しても難しい．各ウェブサイトに要約が存在すれば，ユーザは要約を見ることでそのウェブサイトに書かれている内容を理解することができる．

本システムでは，ウェブサイトの内容の要約を自動的に作成し，検索結果と共に表示することで，ウェブサイトの内容の概略が分かり，必要とする情報により短時間で到達可能とした．

本システムではウェブサイトの収集に Google Custom Search API を使用し，ウェブサイトのリンクを取得する．各サイトへのリンクから HTML 文書を取得し，本文を抽出した後，要約を行う．要約過程では不要な句の削除や文への点数付けを行い，重要文を抽出することで要約を作成する．

2. サイトへのリンクから本文の抽出

ローカル上でユーザに検索ワードをテキスト入力欄に入力してもらう．しかし，HTML 文書のテキスト入力欄にユーザが入力した検索ワードを Python は取得することができない．そのため，Python のウェブフレームワークである Bottle を使用し，ローカル環境で通信を行い，検索ワードを Python で取得できるようにした．

検索結果の URL の取得には Google Custom Search API を使用している．ユーザが入力したワードを Google Custom Search API に渡すことで検索結果の URL を得ることができる．それらの URL を使用して各ウェブサイトにアクセスし，HTML 文書を取得している．

HTML 文書は自己サイトへのリンクなどの要約を行う上で不必要な情報が含まれるため，HTML 文書から文章の抽出を行う必要がある．Python のパッケージの readability-lxml を使用することで，HTML 文書から文章の抽出を行った．

しかし、readability-lxml は YouTube や画像主体のサイトでは文章の抽出を行うことができないことがあるが，本研究の目的には差し支えない．

3. 本文の要約.

要約には重要文抽出の手法を使用した．重要文抽出は原文中の個々の文に対して点数付けを行い，点数の高い文章から N 個の文を取り出すことで要約を作成する手法である．重要文を取り出すだけでなく文章中に存在する不要な単語や句の削除を行うことで，より短い要約を作成することができる．

3.1　文短縮

括弧で囲まれた文章は補足情報である場合が多く，重要であるとは言えないため，削除する．しかし，鉤括弧で囲まれた文章は強調するために使われる場合もあり，重要な単語を含む可能性があるため，削除を行わない．

3.2　表記のゆれの統一

文短縮をした後，MeCab を使い形態素解析を行った．また，形態素解析と同時に表記の揺れを抑えるために単語の統一を行った．例えば，”子供”，”子ども”，”こども”をすべて”こども”に統一するという具合である．また，動詞についても「動き」や「動け」などの活用形を「動く」という原型に修正した．この活用形を原型に修正することはのちに行う単語の重要度の計算に用いるための修正であり，ユーザに表示する要約については修正をしていない文を表示する．

3.3　単語の重要度の計算

単語の統一や，活用形を原形に変更した文書について，単語の重要度の計算を行った．単語の重要度の計算には単語の出現頻度を単語の重要度と考える TF 法を用いた．

文章中には「とても」や「すごく」といった副詞などの動詞，形容詞，名詞以外の品詞の単語は文章の特徴を表していない単語が多く出現する．これらの動詞，形容詞，名詞以外の単語については重要度の計算を行わない．しかし，動詞の”する”や、”ある”は文章の内容の特徴を表していないが，出現回数が多くなり重要度が高くなる．これらの文章の特徴を表していないが，出現頻度が多いため重要度が高くなる単語の削除をするため，それぞれの単語の出現回数の計算後，平均の 2 倍以上出現している単語は，重要度を 0 とした．

3.4 文の重要度の計算

重要度の高い単語を使用し各文に対して点数付けをし，重要文の抽出を行う．ここでの文への点数付けの方法として，重要度の高い単語がある範囲内にどれだけ含まれているかで評価した．このシステムでは，文中に登場する重要単語と，その次の重要単語の距離が，5 未満であれば，それらをまとまりとして扱い，またその次の単語との距離が 5 未満であればまとまりに組み込みことを繰り返す．重要単語とその次の重要単語の距離が 5 以上であれば，そのまとまりに新たな単語を組み込むことをやめ，また新たなまとまりを作成する．

すべての文に対して上記の手法を行ったの後，まとまりが大きい文ほど重要であると判断する．

本システムでは Luhn[1]の手法を採用し，まとまりを作る重要単語の距離を 5 とした．

4. ユーザが知りたい情報の要約

3 章では，自動要約の手法について説明したが，ここで作成される要約はウェブサイトの要約であり，ユーザの知りたい情報が入っているとは限らない．ウェブサイトの要約にはユーザにとって不必要な情報も含まれる．そこで，ユーザの知りたい情報を要約に取り込む操作を行う．

4.1 単語の重要度の計算

本システムにおいてユーザの知りたい情報は検索ワードの単語と判断する．例えば検索ワードが「Python 自然言語処理」であれば，ユーザは「Python」と「自然言語処理」について知りたいということである．

原文中からこれらの単語を含む文章を抜き出す．文章を抜き出す作業は HTML 文書の P タグ内の文章に入力された単語が含まれていれば，その文章を抜き出す．この作業を行った後に 3 章で説明した要約の手法で要約を作成する．

5. 実験結果

「自然言語処理」を検索ワードとして実行した結果の一部を図 1 に示す．

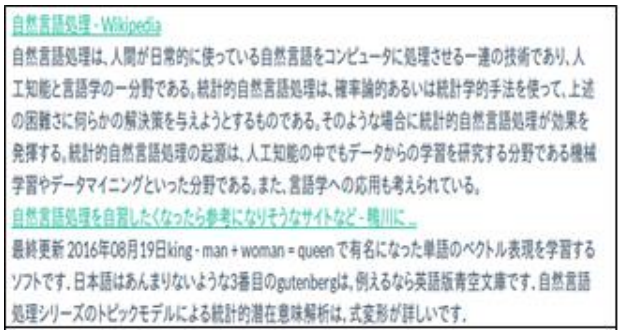
自然言語処理 - Wikipedia
自然言語処理は、人間が日常的に使っている自然言語をコンピュータに処理させる一連の技術であり、人工知能と言語学の一分野である。統計的自然言語処理は、確率論的あるいは統計学的手法を使って、上述の困難さに何らかの解決策を与えようとするものである。そのような場合に統計的自然言語処理が効果を発揮する。統計的自然言語処理の起源は、人工知能の中でもデータからの学習を研究する分野である機械学習やデータマイニングといった分野である。また、言語学への応用も考えられている。
自然言語処理を自習したくなったら参考になりそうなサイトなど - 鵜川に...
最終更新 2016年08月19日king - man + woman = queenで有名になった単語のベクトル表現を学習するソフトです。日本語はあんまりないような3番目のgutenbergは、例えるなら英語版青空文庫です。自然言語処理シリーズのトピックモデルによる統計的潜在意味解析は、式変形が詳しいです。

図 1 「自然言語処理」での検索結果

実線が引かれている文章はリンク先のタイトル，タイトルの下の文章がリンク先のウェブサイトの要約となっている．原文の内容を要約できている部分もあるが，一部のウェブサイトの要約では不必要な情報が要約に存在する．

本システムでは要約の長さを原文の文章の長さの 10%としたが，ウェブサイトによって文章の量がさまざまであり，長い要約のウェブサイトと，短い要約のウェブサイトが存在する．ウェブサイトによって要約の量の差が大きくなってしまい，ユーザの得られる情報に差が出てしまうこともある．

6. 今後の展望

重要である文を取り出してまとめるという要約の手法を用いているため，要約の文章が何について書かれている文章なのかわからない要約も含まれている．図 1 では「king - woman = queen で有名に・・・」という文章は word2vec についての説明であるが，word2vec を知らない人にとっては何についての文章なのか理解することは難しい．重要な文を要約として取り出す際にはその段落の重要文以前の文章を取り出すなどの方法をとることにより，改善することができる．

7. 参考文献

[1] Luhn,H.(1958)：″The automatic creation of literature abstracts.″ IBM Journal of Research and Development, 2 (2), pp.159-165.