6K-09

開発文書間におけるトレーサビリティ抽出方法の提案と評価

浜田尭雅† 中島毅†
芝浦工業大学†
髙橋加寿子‡ 徳本修一‡
三菱電機株式会社 情報技術総合研究所‡

1. 研究背景

ソフトウェア開発において,各開発工程における成果物間の関連性のことをトレーサビリティ[1],特定の成果物間の関連を追跡するための情報のことをトレーサビリティリンクという.トレーサビリティは要求事項の成果物間での対応漏れを確認することや,仕様変更の際の影響分析にも利用できるので,特に大規模ソフトウェア開発の品質確保のために重要である.トレーサビリティリンクの確立・維持は,開発者にとって作業負担が多い.そこで,トレーサビリティの維持管理をせずに要求事項の実装や変更影響範囲の確認を支援できることが求められている.

2. 従来研究と課題

トレーサビリティリンクについての従来研究にはTIAN氏らの研究[2]がある.同研究ではword2vec[3]を用いて単語の分散表現を生成し,文章に存在する単語の分散表現の総和の平均を文章の特徴量として利用することや,出現頻度の低い単語に着目してトレーサビリティリンクの候補を生成している.この従来研究では単語に焦点を当てているため,文章の順序構造については考慮できていない点が課題として挙げられている.

この課題を解決するために文章単位で分散表現の生成を行う doc2vec[4]を用いることが考えられるが,この研究では,doc2vec をただ使用するだけでは有効性が低いと結論づけている.そこで本研究ではトレーサビリティ候補抽出に doc2vec を有効に使用できるような方法を提案する.

3. 研究目的

本研究は,要求仕様書と方式設計書における文章の分散表現を用いてトレーサビリティリンクが存在しそうな箇所の候補を抽出する方法を提案する.これによりトレーサビリティリンクの確立を行う際に確認する成果物の箇所を少なくし,作業負担を軽減させることを目的とする.

4. 研究内容

4. 1 提案概要

提案方法の概要を図1に示す.

まず開発文書を一文ごとに分け比較文を生成する.次に doc2vec を用いて入力文と各比較文の分散表現を得る.そして入力文の分散表現と各比較文との分散表現から入力文と比較文の類似度を算出し,類似度が高い文章をトレーサビリティの候補として出力する.

なお,doc2vec のモデルには日本語wikipediaのコーパスを学習したものを用いた.

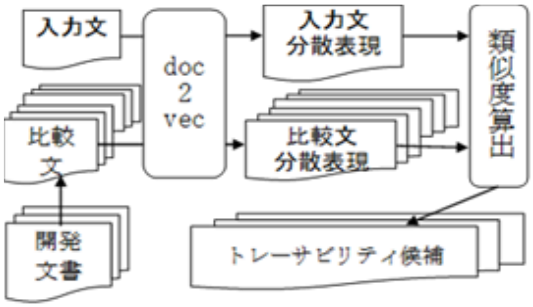

図1 実験の流れ

4. 2 予備実験

doc2vec の単純な適用により,入力文に対してどのような文が高い類似度を示すのかを調べるために,実際の開発文書を用いて予備実験を行った.類似度として,入力文の分散表現に対する対象文のコサイン類似度を用いた(doc2vec が算出する分散表現に基づくコサイン類似度を doc2vec 類似度と呼ぶこととする).

予備実験の結果,doc2vec 類似度は文の構造が近いものが高くでる傾向にあることがわかった.

この傾向と分析を表1に示す例文によって詳しく説明する.表1の入力文と比較文を見た際,比較文Bの方が類似度が高い印象を受ける.しかし,実際の doc2vec 類似度では,助詞や接続詞のような出現頻度の高い単語が文に強い影響を与え,文の構造が似ている比較文Aの方が類似度が高くなったと推測できる.

表1 予備実験に用いた入力文と比較文例

入力文	画面点灯時にパワーボタンが入力された場合、画面を消灯する。
比較文A	文字入力待ち時にキーボードが入力された場合、文字を出力する。
比較文B	入力のない時が1分続いた場合、画面を消灯する。

4. 3 提案方法

予備実験の結果から doc2vec 類似度の上位には,文の構造が似ているものが現れる.関連性の高い文章は,文の構造に加え,出現する単語が似ている傾向があるのではないかという仮説を立てた.本研究で

「Proposal and Evaluation of Traceability Extraction Method between Development Documents」
†「Takumi Hamada・Shibaura Institute of Technology」
†「Tsuyoshi Nakajima・Shibaura Institute of Technology」
‡「Kazuko Takahashi・Mitsubishi Electric Corporation」
‡「Shuichi Tokumoto・Mitsubishi Electric Corporation」

は，この仮説に基づき，doc2vec 類似度と出現する単語による類似度の重みづけ加算により，文章の関連度を求めることとした．具体的には入力文と比較文で同一の名詞がある場合と動詞がある場合に doc2vec 類似度に重みづけを行い，次式のように関連度$Simf$を求める．

$$Sim_f = Sim_d \cdot \mathrm{n} \cdot \mathrm{m}$$

ここで，Sim_dは doc2vec 類似度．nは入力文と比較文とで同一の名詞，もしくは動詞の数．mは重みである．

評価実験

4．3．1 対象データ

対象にしたデータは組込みシステムについての開発文書で，対象としたドキュメントは要求仕様書と方式設計書である．開発文書内には変更要求が生じた際に影響があった箇所が識別できるようになっており，その箇所にある文章同士を「トレーサビリティのある文章」の正答とした．

4．3．2 評価方法について

本実験では，トレーサビリティ候補の箇所の抽出により，作業負担の軽減を目的としているため，入力文に対して，比較文を関連度Sim_fの高い順に並べた際に，比較文全体の上位何%までにトレーサビリティがある文章がすべて含まれるかの割合を評価値として用いる．評価値はFとし，以下の式で算出する．

$$F = \frac{(\sum_i^N Rank_i)}{N_{input} \cdot N_{data}}$$

ここで，$Rank$は入力文に対して算出した関連度$Simf$の高い順に文章を並べた際にトレーサビリティのある文章がすべて含まれている順位（最低値）．N_{input}は入力文の総数，N_{data}は比較文の総数であり，本実験ではそれぞれ 52 文，21412 文であった．

また，評価値は，ある文章に対してトレーサビリティのある文章を見つける際に見るべき文章の割合となる．そのため，評価値が低いほど評価は高くなる．

4．3．3 実験結果と考察

図２は，名詞と動詞に対して重みづけを行った結果を示す．図の x 軸が重みの値mで，y 軸が評価値Fである．また，表２は各方法における重み値と評価値の比較を行った結果である．

入力文と比較文両方に存在する名詞に関して，重みづけを行い，類似度を算出した場合，重みの値を大きくするほど評価が高くなっている．これは関連度の高い文章は，出現する単語が似ているという仮説が名詞においては検証されたものと考える．具体的な値を見ると，重み値が 0.28 の時，重みなしの評価値の半分となり，これは重みなし時に比べると，トレーサビリティのある文章を見つける際の比較文章を半分に減らせるということである．

しかし，重みの値を大きくし過ぎると入力文と比較文両方に存在している名詞の数のみで判断しているのと相違なくなってしまい，文章の順序構造を無視してしまうことになる．その為重みの値は大きすぎず，小さすぎない値でないとならないと言える．

動詞に関して重みづけを行った場合に関しては，評価値に関するグラフは特に変化が見られなかった．これは入力文 52 文に対して，動詞を含む入力文が 8 文であり，動詞の重みづけが関連度に与える影響が少なかったためと考える．

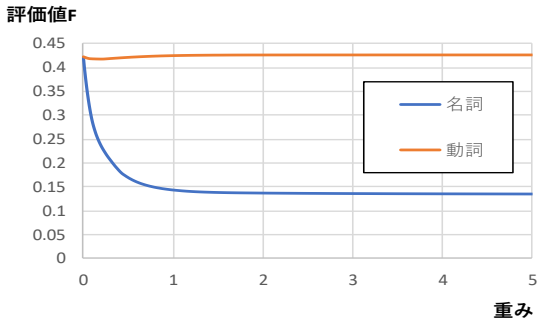

図２ 名詞に対して重みづけを行った結果

表２ 各方法における重み値と評価値の比較

	重みなし	名詞に重み		動詞に重み
評価値	0.421	0.210	0.133	0.416
重み	0.0	0.28	10.0（最大値）	0.2

5． まとめと今後の課題

本研究では，文章の分散表現を比較する文章同士の同一の単語に着目しながら利用することにより，開発文書間におけるトレーサビリティ抽出を行う方法を提案し評価を行った．その結果，名詞に重みづけを行う方法に関しては，文章の分散表現のみでトレーサビリティの抽出を行うケースよりも本研究の方法を用いることの有用性を確認できた．

今後の課題としては，名詞に対して重みづけを行う際の重みの値の最適化や，異なる開発文書に対しても本研究の方法が同様の傾向と有用性を示すことができるかどうか．また，文章の構造以外の開発文書特有の性質の利用の検討が挙げられる．

参考文献

[1] 宇田川佳久, et al. 情報システム開発におけるトレーサビリティの事例と今後. 情報処理, 2010, 51.2: 150-158.
[2] TIAN, Qingsong; CAO, Qinghua; SUN, Qing. Adapting Word Embeddings to Traceability Recovery. In: 2018 International Conference on Information Systems and Computer Aided Education (ICISCAE). IEEE, 2018. p. 255-261.
[3] MIKOLOV, Tomas, et al. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781, 2013.
[4] LE, Quoc; MIKOLOV, Tomas. Distributed representations of sentences and documents. In: International conference on machine learning.