一般語彙と専門語彙の Word Embedding を併用した特許文書検索

Patent document search using both general and field-dependent word embedding model

株式会社 NTT データ数理システム　データマイニング部　グループリーダー・主任研究員
岩本　圭介

1999 年株式会社数理システム（現：株式会社 NTT データ数理システム）入社。データマイニング及び深層学習関連パッケージ製品の開発リーダーを務める。現職はデータマイニング部グループリーダー・主任研究員。

iwamoto@msi.co.jp

株式会社 NTT データ数理システム　データマイニング部　主任研究員
柿沼　匡志

2012 年株式会社数理システム（現：株式会社 NTT データ数理システム）入社。自然言語処理・テキストマイニングに関わるツール・手法開発及び分析業務に従事。現職はデータマイニング部主任研究員。

kakinuma@msi.co.jp

1 はじめに

文書どうしがどれくらい似ているか、文書間の類似の度合いを評価することは、自然言語処理の主要な研究対象のひとつである。人間は文書群を読解し、「この文書とこの文書は同じようなことを言っている」「この文書とこの文書はあまり関係がない」といった認識を持つことができるが、これは文書間の類似を人間が認識し感じ取っているものといえる。このような働きをコンピュータ上で実現するために、そういった類似の度合いを「類似度」として数値化することは広く行われている。特許文書間に類似度を適切に定義することができれば、興味の対象である文書に対し類似度が高い他の文書を類似特許として提示する、もしくは互いに類似度が高い文書群を抽出しそれらをグループとして自動的にまとめ上げる、といった有益な応用が見込める。

文書間の類似度を判断するにあたっては、従来手法としては、両文書間で共通の単語がどれだけ用いられているか、といった単語のマッチングに基づいた手法が広く利用されてきたが、近年は、単語や文書そのものをベクトルとして表現しそれらの間の類似度を評価すること、またそのようなベクトル表現を学習によって獲得するための方法が数多く研究されている。単語のベクトル表現を求めるためのニューラルネットワーク手法 word2vec[1] の利用の広がりはまさにそれを象徴する出来事である。

学習によって得られた単語のベクトル表現には、オープンな情報として公開されているものもあり、技術のさらなる適用[2][3]に拍車をかけている。Wikipedia 等の大規模データソースをもとに学習を行ったものが知られており[4][5]、日常的な語彙を十分に含み利用に耐えるものである。

こういった大規模データに基づく単語のベクトル表現は、世間の一般的な単語の利用状況や出現パターンを反映したものである。しかし、専門性の高い特許文書や技術文書などでは、それとは異なった状況が出現している、もしくは極めて特殊な単語などは一般的な語彙ではカバーされないという可能性もある。

本稿では、大規模かつ一般的な語彙をもとにして学習して得られたベクトル表現に加えて、関心の対象である分野に絞った、専門語彙を含むデータから獲得したベクトル表現をあわせて用いることで、精度の高い類似度の算出を実現するための試みについて解説する。以降、2章で単語と文章のベクトル表現獲得のための基礎概念を解説し、3章では一般語彙と専門語彙の併用による類似文書検索手法を提案し、特許文書群に対しての適用例を示す。最後に、4章でまとめを行う。

2 単語ならびに文書の分散表現

2.1 分散表現とは

コンピュータが情報処理を行う際は、データを数値化して論じる必要があり、単語や文書といった情報が対象物であるときもこれは同様である。古典的な考え方では、まず語彙数と同じ次元数を持つベクトルを準備し、ベクトルの次元にそれぞれの単語を割り当てる。そして、ある単語に対応する次元のみが 1 で他が全て 0 であるベクトル（one-hot 表現）とこの単語そのものを同一視する。また、ある文章に含まれる単語の次元のみが 1 でその他の次元が全て 0 であるベクトル（Bag of Words）とその文章そのものを同一視する（図 1）。

a) one-hot表現

	分子量	モル質量	測定	計測
分子量	1	0	0	0
モル質量	0	1	0	0
測定	0	0	1	0
計測	0	0	0	1

b) Bag of Words

分子量を測定した。	{ 1 , 0 , … , 1 , 0 }
モル質量を計測した。	{ 0 , 1 , … , 0 , 1 }

図 1　one-hot 表現 と Bag of Words

この時点で 1 つの単語や文書 がベクトルとして表現できたことになるが、図 1 の Bag of Words をもとに文章間の類似を考えることは単語間の正確なマッチングによる判定を行っていることに外ならず、それぞれ「分子量」と「モル質量」、「測定」と「計測」は互いに類似した対象を表しているにも関わらず両文書間でマッチする要素はないため類似の内容であると判断されない。

分子量	= {	0.8 ,	0.7 , … ,	0.1 ,	0.2 }
モル質量	= {	0.9 ,	0.6 , … ,	0.2 ,	0.2 }
測定	= {	0.1 ,	0.3 , … ,	0.8 ,	0.7 }
計測	= {	0.2 ,	0.1 , … ,	0.8 ,	0.9 }

埋め込み次元数

図 2　単語の分散表現

一方、近年利用が広まったニューラルネットワークモデルを利用した単語のベクトル表現では、互いに類似した単語や文書が空間上の近い点になるよう（図 2）、学習によってこの表現を獲得する。図 1 と 図 2 はどちらもベクトルによる表現であるが、前者は 1 つの次元 がそのまま 1 つの単語 となるようその意味が固定されている（分散していない）。逆に、後者ではベクトルに埋め込まれた（embedded）数値パターンそのものが単語の表現であり、概念や意味がベクトル内に分散した形で表現されているといえる。このベクトルの次元数は埋め込み次元数と呼ばれ、学習の際に指定するパラメータである。一般に、図 1 の one-hot 表現や Bag of Words では次元数は数万～数百万のオーダーになるが、この埋め込み次元にはそれよりはるかに小さい値を指定するため（例として、100 や 200 など）単語の情報をコンパクトに表現できる利点もある。このような特性を持ったベクトル表現は、特に 分散表現（distributed representation）と呼ばれ、単語をこのようなベクトルに埋め込まれた形で表現する行為を Word Embedding と呼ぶ。

2.2 単語の分散表現

単語の分散表現を獲得する手法の一つ、CBOW（Continuous Bag-of-Words Model）[6] を紹介する。CBOW は、周囲の単語群が入力されたときに、その中心となる単語を予測するように学習を行うニューラルネットワークモデルである。単語を w とし、文書内に単語列 $\{\cdots, w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2}, \cdots\}$ があったとき、適当なウィンドウ幅 n を事前に決めておき、$\{w_{i-n}, \cdots, w_{i-1}, w_{i+1}, \cdots, w_{i+n}\}$ が入力されたときに w_i が出力されるように学習を行う。モデルの概略図を図 3 に示す。

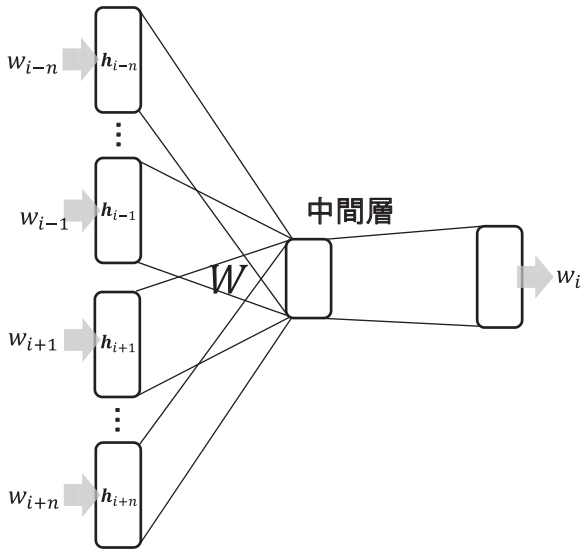


図 3　CBOW モデルの概略

中心単語を予測するモデルではあるが、予測を行うことがこのモデルを利用する際の主眼ではない。実際は、ネットワークには周辺単語 $X = \{w_{i-n}, \cdots, w_{i-1}, w_{i+1}, \cdots, w_{i+n}\}$ に対する one-hot 表現 $\{\boldsymbol{h}_{i-n}, \cdots, \boldsymbol{h}_{i-1}, \boldsymbol{h}_{i+1}, \cdots, \boldsymbol{h}_{i+n}\}$ を入力し、そのとき中間層の値は $\frac{1}{|X|}\Sigma_{j \in X} W\boldsymbol{h}_j$ とするが、周囲の語が似た単語 は 似た出力を導くようこの W が学習の過程で調整され、結果として単語 i に対応する分散表現 $W\boldsymbol{h}_i$ は周囲の語の状況、いわば文脈を考慮したベクトル表現になることが期待できる。

2.3　文書の分散表現

2.2 章では 各々の単語 に対しての分散表現を得る手法を解説したが、同様に 各々の文書そのもの を分散表現として表すことが考えられる。文書の分散表現を獲得するための手法には、主として以下のアプローチが存在する。

- 文書の分散表現 そのものを学習によって得る
- 文書を構成する単語それぞれの分散表現 から、文書そのものの分散表現を構築する

前者の代表的かつ著名なものに Paragraph Vector[7] がある。後者は、文書の構成単語の分散表現から文書そのものの分散表現を組み上げるものであり、シンプルだが検証において良好な成績を示すものとして知られてきた SWEM（Simple Word-Embedding-based Models）[8] を以下紹介する。

文書は一般に複数の単語から構成されるが、SWEM は構成単語の分散表現ベクトル群に対し、各次元の 最大値（SWEM-max）もしくは 平均値（SWEM-aver）を取ったベクトルを文書全体の分散表現として利用する方法である。

すなわち、文書 D の分散表現を $\boldsymbol{v}$ としたとき、それぞれ

- SWEM-max : $\boldsymbol{v} = MAX\text{-}pooling(\boldsymbol{v}^{w1}, \boldsymbol{v}^{w2}, \cdots, \boldsymbol{v}^{wn})$
- SWEM-aver : $\boldsymbol{v} = \frac{1}{n}\Sigma_{i=1}^{n} \boldsymbol{v}^{wi}$

と表せる。ただし、

- D の 構成単語を w とする。すなわち、D の構成単語数を n として $D=\{w_1, w_2, \cdots, w_n\}$
- さらに w_i の 分散表現を $\boldsymbol{v}^{wi} = \{v_1^{wi}, v_2^{wi}, \cdots, v_m^{wi}\}$ とする、ただし m は埋め込み次元数

である。埋め込み次元数 m は、Word Embedding の学習時に決めておくパラメータである。

3　一般語彙と専門語彙の併用

3.1　Word Embedding を利用した類似文書検索

続いて、2 章 で解説した分散表現を利用して 類似文書検索 を行う枠組みについて考える。

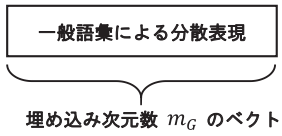
検索対象文書群 $D = \{D_i\}$ に対してクエリ文書 D_Q が与えられたとき、D_i の分散表現 $\boldsymbol{v}^i$ と D_Q の 分散表現 $\boldsymbol{v}^Q$ との類似度を コサイン類似度 similarity($\boldsymbol{v}^i$, $\boldsymbol{v}^Q$) = $\frac{\boldsymbol{v}^i \cdot \boldsymbol{v}^Q}{|\boldsymbol{v}^i||\boldsymbol{v}^Q|}$ によって与える。これにより D_i と D_Q との間の類似度を [-1,1] の範囲で決定することができ、similarity($\boldsymbol{v}^i$, $\boldsymbol{v}^Q$) の高い順に D_i をリストアップすることで類似度に応じて検索対象文書にランキングを付与することができる。

ここで、単語 w の分散表現に次の 2 通り を考える。手法 A, B の模式図を図 4 に示す。特に、手法 B が一般語彙と専門語彙それぞれによる Word Embedding

の結果を併用する、本稿での提案手法である。

- 手法 A) 一般語彙 で学習した 分散表現
  - $\boldsymbol{v}_G^w$（埋め込み次元数 m_G）
- 手法 B) 一般語彙 で学習した分散表現 と 専門語彙で学習した分散表現 を連結したもの
  - 専門語彙で構築した分散表現を $\boldsymbol{v}_F^w$（埋め込み次元数 m_F）として、$\boldsymbol{v}_G^w$ と $\boldsymbol{v}_F^w$ を連結した（$\boldsymbol{v}_G^w$, $\boldsymbol{v}_F^w$）を用いる（埋め込み次元数 m_G+m_F）

手法A）



手法B）

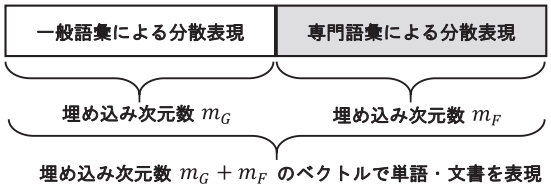


図 4　手法 A, B の模式図

3.2　特許文書検索への応用

本節では、3.1 章の手法により分散表現を利用した特許検索を試行し、手法 A)，B）それぞれの評価を行う。対象データや Word Embedding の実行条件を 表 1 に示す。

表 1　対象データと Word Embedding 実行条件

対象データ	原子力関連分野 の 国内特許公報 要約部分 460 件
一般語彙による分散表現	
配布先	参考文献 [4] を利用
埋め込み次元数 $\boldsymbol{v}_G^w$	50
専門語彙による分散表現	
学習データ	上記「対象データ」460 件
埋め込み次元数 $\boldsymbol{v}_F^w$	50
学習過程	ライブラリ gensim を利用

表 1 の条件によって得られた 一般語彙による分散表現 $\boldsymbol{v}_G^w$ と 専門語彙による分散表現 $\boldsymbol{v}_F^w$ を用いて、次の手順で 3.1 章 に示した類似文書検索を実施し評価を行う。

1．表 1「対象データ」460 件 から クエリ文書 D_Q を選出する。検索対象集合 D = $\{D_i\}$ は残りの 459 件とする。
2．手法 *A*, 手法 *B* 双方の手法を用いて、*D* からクエリ文書 D_Q との類似度が高い上位 *r* 文書を抽出する。
3．上位 *r* 文書の中に、D_Q の類似文書として妥当なものがどの程度含まれるか評価する。
4．別のクエリ文書 D_Q を選出し、すべての文書について上記の手順を繰り返す。

検索の妥当性は客観的に評価できる必要があるため、ここでは D_Q と類似度上位の *r* 文書との間で、筆頭 IPC のクラス・サブクラス・メイングループが一致する文書が幾つあったかという観点で確認を行った。

評価の結果を表 2 に示す。表の数値は、クエリ文書を変えて 460 回検索を行った結果の中で、クエリ文書と筆頭 IPC の クラス・サブクラス・メイングループ が一致したものの割合である。ただし、一回の検索当たりの上位抽出数は *r* = 3 とした。

表 2　手法 A, B による一致率

	評価対象	手法 A	手法 B
SWEM-max	クラス	51.23	53.84
	サブクラス	27.90	29.42
	メイングループ	14.93	17.39
SWEM-aver	クラス	65.72	**66.67**
	サブクラス	40.80	**44.06**
	メイングループ	28.04	**30.87**

SWEM-max と SWEM-aver を比較すると、文書の分散表現作成手法として SWEM-aver を用いた方が全体的に結果は良好であった。また、SWEM-max, SWEM-aver それぞれ同一条件の中では手法 A よりも提案手法である B の方が高い一致率を示していることがわかる。

検索の結果マッチした IPC の状況の例を表 3 に示す。例 1 では類似度上位 3 文書のうち手法 A では 1 文書のみマッチ、手法 B では 3 文書ともにマッチした例である。例 2 は、手法 A ではマッチせず、手法 B で 2 文書がマッチした例である。

表３　検索結果の筆頭 IPC（サブクラス）例

例１：クエリ D_Q の筆頭 IPC（サブクラス）B62D

類似度順位	手法 A		手法 B	
	類似度	サブクラス	類似度	サブクラス
1	0.99	B25J	0.99	**B62D**
2	0.99	B25J	0.98	**B62D**
3	0.99	**B62D**	0.98	**B62D**

例２：クエリ D_Q の筆頭 IPC（サブクラス）G21C

類似度順位	手法 A		手法 B	
	類似度	サブクラス	類似度	サブクラス
1	0.94	G21D	0.96	**G21C**
2	0.94	F03G	0.96	G21D
3	0.95	B34C	0.96	**G21C**

4 まとめ

一般語彙モデルと専門語彙モデルを併用することで、専門性の高い文書群においても、その分野に依存した情報を取り込み更に精度の良い文書検索が行える可能性を示した。

当社株式会社 NTT データ数理システム の取り組みとして、利用者自身が持つテキストデータに対して言語解析処理を行って単語の抽出を行い、またそれらに対して Word Embedding をはじめとする深層学習のアルゴリズムを適用させられるツールを開発・販売している。これらは共通の基盤プラットフォーム上で提供され、ツール間をシームレスに連係させて利用することができる。自分自身のデータを用いて、独自の Word Embedding モデルを構築することも可能である。図５に、テキストマイニングツール Text Mining Studio の解析結果に対して、深層学習ツール Deep Learner を適用させて Word Embedding モデルの学習を行っている様子を示した。これら分析ツールの組合せによって、多様化するデータに対しての様々な要請に応えることができると当社では考えている。

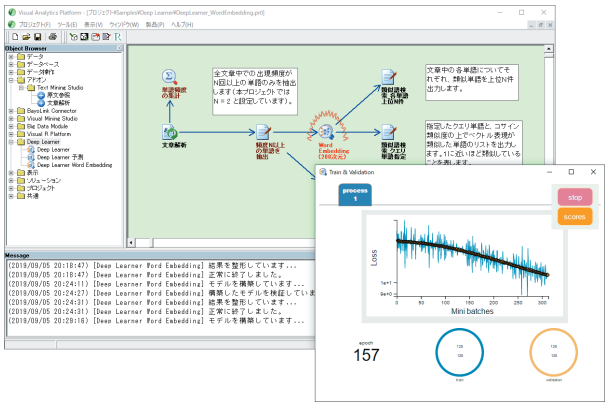


図５　Deep Learner を用いた学習の様子

参考文献

[1] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed Representations of Words and Phrases and their Compositionality,” in Advances in Neural Information Processing Systems 26, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 3111-3119.

[2] H. Aras, H. Sack, R. Turker, D. Geiss, and M. Milbradt, “Get Your Hands Dirty: Evaluating Word2Vec Models for Patent Data,” in Proceedings of the Posters and Demos Track of the 14th International Conference on Semantic Systems - SEMANTiCS2018, 2018.

[3] 樋口廉，川野邊誠，“ニューラルネットワークを用いた国際的なニューストピックスへの国民感情を踏まえた意見表明,” in エンタテインメントコンピューティングシンポジウム 2018 論文集, 2018, vol. 2018, pp. 226-229.

[4] “word2vec の学習済み日本語モデルを公開します,” カメリオ開発者ブログ. [Online]. Available: http://aial.shiroyagi.co.jp/2017/02/japanese-word2vec-model-builder/. [Accessed: 26-Aug-2019].

[5] “日本語 Wikipedia エンティティベクトル.” [Online]. Available: http://www.cl.ecei.tohoku.ac.jp/~m-suzuki/jawiki_vector/. [Accessed: 26-Aug-2019].

[6] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” Proceedings in the International Conference on Learning Representations (ICLR) Workshop, 2013.

[7] Q. Le and T. Mikolov, “Distributed Representations of Sentences and Documents,” in Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32, Beijing, China, 2014, pp. II-1188-II-1196.

[8] D. Shen et al., “Baseline Needs More Love: On Simple Word-Embedding-Based Models and Associated Pooling Mechanisms,” in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 2018, pp. 440-450.