分散表現とクラス分類に基づく潜在する報告書推薦手法の検討

赤木 里騎† 徐 海燕†

†福岡工業大学大学工学研究科 〒811-0295 福岡県福岡市東区和白東 3-30-1

E-mail: †mfm17101@bene.fit.ac.jp, xu@fit.ac.jp

あらまし 近年，自然言語の分野において単語をベクトル化した分散表現を情報推薦へ応用する研究が盛んになっている．本研究では，本学が運営している内定報告書を web 上で閲覧するシステムにおける内定報告書の推薦に分散表現とクラス分類を用いる．検索時の単語と，内定報告書内の後輩へのアドバイスに記述された文章の単語の分散表現を用いることでコサイン類似度による検索と報告書の類似性を計算する．また，過去実施した報告書に対する評価実験の結果を活用し,クラス分類手法によって評価の高い内定報告書と,そうでなかったものを分類した．検索時の単語と報告書の類似性を測るために分散表現を，報告書自体の質を判断するためにクラス分類手法を活用することで推薦時の重みを調整する 3 つの推薦のパターンを構築した．これらの推薦を比較するために評価実験を実施し，t 検定を行ったところ p=0.02（p<.05）となり，類似性と質に着目した推薦手法の有効性を確認した．

キーワード 分散表現，クラス分類，情報推薦

1. はじめに

近年，機械学習や深層学習が盛り上がりを見せ，自然言語処理の分野では Mikolov ら[1]が提案した word2vec を活用して，分散表現を用いた文書分類に関する研究が増えている[2][3].

本学では，就職支援ツールとして内定報告書を電子化して，職種や企業名などによって検索する報告書閲覧システム Sugoole（以降，Sugoole と呼ぶ）が本研究室で開発され 2008 年度より運用されている．現在約 3,300 件の報告書が登録されている．就職課が本学の就職活動を終えた学生に対して行ったアンケート調査では Sugoole を利用したことがある学生が毎年約 3 割存在していることが分かっているが，Sugoole への要望・改善点として，学生や就職課から「検索時に入力した文字が企業名と一文字でも異なると何も表示されないので類似の報告書を検索結果として表示させてほしい」や「参考にならないアドバイスが多かった」といった意見が出てきている．Sugoole に限らず，検索時の入力文字で完全一致や部分一致を採用する検索システムでは同じような状況に置かれる．既存システムでは報告書を検索する際に企業名での部分一致検索手法が使用されていたために，就職活動が本格的に始まる前の学生にとって企業名を知らない場合には，業種や職種，学科などの大きな枠での検索となっていた．また，今後報告書が Sugoole に蓄積されればされるほど埋もれてしまう報告書が出てきて，役立つ情報が書かれた報告書を検索結果として表示することが困難になってしまうことが考えられる．そのため，自由な入力による検索推薦機能を実装した[4]．推薦には報告書の内容と検索時の入力の分散表現を用いた類似性と，入力と報告書が持つトピックの類似性に着目して報告書毎に推薦度を計算した．しかし，[4]の研究の結果，検索時に使われる入力のほとんどが単語であったため，入力が持つトピックと報告書の中身が持つトピックの類似性を測ることは困難であり，推薦された報告書の中には内容が類似していないもの，類似していても内容が役立つものではないことも多かった．

そこで，本研究ではトピックによる入力と報告書の類似性を測る指標を外して,分散表現に任せた.また，類似しているだけで，報告書自体の質が伴っていない報告書が，過去の評価実験において低い点数をつけられることが多数あった．そこで，評価の点数を正解データとして，評価が低くなる報告書をクラス分類手法により予測して推薦の重みを下げる.そうすることで，類似している報告書の中でも，質の高い報告書を推薦することができると考えた．

検索時のイメージと近く，内容が充実した報告書を推薦することで利用者の就職活動に有用な報告書を推薦する機能を構築する．

2. 分散表現とクラス分類

検索時の入力と報告書の類似性に分散表現を，報告書自体の質をクラス分類手法によって数値化する．本章では分散表現とクラス分類手法について説明する．

分散表現は単語を高次元のベクトルで表しているため応用がしやすく様々な研究がある．分散表現を用いて単語の類似度を求める事で文書の類似度を学習する研究[5]や，単語の意味を単語の周辺単語から学習する分散表現の弱点である語義曖昧性を解消するための研究[6]，web 上で得られた観光地の情報から分散表現を学習させて観光地の推薦をする研究[7]など，分散表現を応用する研究は近年増えている．

機械学習には大別して教師あり学習,教師なし学習,強化学習の 3 つの学習方法がある．その中でも教師あ

表 1　分散表現学習に用いたコーパスの詳細

	後輩へのアドバイス	キャリアポートフォリオ
単語数	12 万 5677 個	37 万 2789 個
語彙数	7134 個	1 万 1598 個
文書数	3324 個	3 万 2673 個
平均文字数	130 文字	39 文字

り学習にクラス分類は含まれ，学習対象となるデータに正解データを付与して学習する方法である．本研究では，scikit-learn ライブラリ[1]のクラス分類手法として代表的な SVM（SVC）[8]，RandomForestClassifier[9]，MLPClassifier[10]によって報告書の評価を予測する．

本研究で学習データに付与する正解データは過去の評価データとする．ただし，個人の主観で大きく点数が異なってしまう．高評価か低評価のどちらかの点数をつけがちな利用者や，高い点か低い点の極端な点数をつける利用者が存在する．また，同じ報告書でも利用者や，調べたいことによって評価は異なるので，高い精度で報告書の評価が高い，低いことを予測することは困難である．そこで本研究では評価の低い報告書の予測精度に着目し，評価の低くなりやすい報告書の推薦時の重みを低くすることによって評価が低くなりやすい文章の報告書を推薦しないようにする．予測精度だけでなく適合率や再現率，F 値[11]にも着目することでクラス分類手法の選定を行う．選定については第 4 章で説明する．

本研究では検索時の入力単語の分散表現と報告書の後輩へのアドバイスが持つ単語の分散表現による類似性に加えて，報告書が持つ情報を特徴量としてクラス分類をして質を考慮する事で，報告書の類似性と質に着目して推薦の精度向上の手法を提案する．

3. 報告書推薦システム

報告書を推薦するために検索時の入力と報告書の類似性と報告書自体の質を推薦のための 2 つの軸とした．本章では，分散表現学習のためのテキストに対する前処理と，推薦に用いた 2 つの軸について説明する．

3.1 テキストの前処理

分散表現の学習においてテキストへの前処理によって分散表現モデルの精度は変わる．そのため，本研究では以下のようにして分散表現を学習する前に前処理を施した．

1. クリーニング処理
   1.1 多くの報告書に出現するテンプレートワードの削除
   1.2 表記揺れへの対応
   1.3 HTML タグの削除
2. 単語の正規化
   1.1 英語を半角小文字へ変換
   1.1 カタカナを全角へ変換
3. 助詞と助動詞，記号，数字を削除する分かち書き
4. ストップワードの除去（SlothLibrary の活用[2]）

表記揺れへの対応は，就職活動の報告書ということもあって，「es」，「gd」のような略称が存在していたので「エントリーシート」，「グループディスカッション」のように置換処理を行った．また，形態素をスペース区切りで分ける分かち書き処理をする時に，日本語におけるつなぎ言葉である助詞や助動詞などは記号や数字と同様に曖昧な意味を持つため削除した．

3.2 検索時の入力と報告書の類似性

検索時の入力と報告書の類似性を計算するために分散表現を用いているが，分散表現の学習には Python の gensim[3] ライブラリの word2vec を使用した．word2vec には多くのパラメータを設定することができ，パラメータによって学習結果は変わる．本研究で設定したパラメータは以下の通りである．

- ■学習モデル：skip-gram
- ■次元数：100
- ■出現回数が n 回未満の単語を削除：1
- ■ウィンドウサイズ：10
- ■エポック：50
- ■ネガティブサンプリングサイズ：20

分散表現の学習にはパラメータ以外にも word2vec の入力となる文書の集合群（以下，コーパスと呼ぶ）が重要になる．word2vec に与えるコーパスは大きけれ

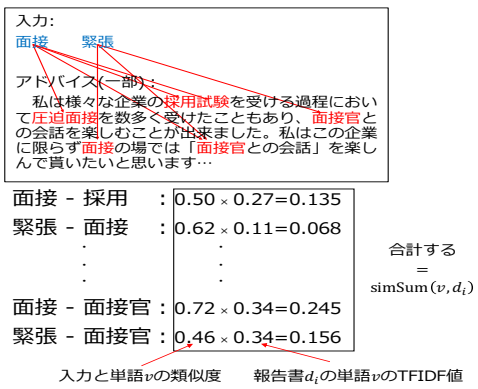

図 1　*simSum*の算出

[1] https://scikit-learn.org/stable/

[2] http://svn.sourceforge.jp/svnroot/slothlib/CSharp/Version1/SlothLib/NLP/Filter/StopWord/word/Japanese.txt

[3] https://github.com/RaRe-Technologies/gensim/blob/develop/gensim/models/word2vec.py

表 2　後輩へのアドバイスとキャリアポートフォリオコーパスの分散表現を用いた上位類似単語

	「面接」を入力とした場合		「自己 pr」を入力とした場合	
類似度順位	類似語	コサイン類似度	類似語	コサイン類似度
1	筆記試験	0.73	志望動機	0.92
2	spi	0.72	履歴書	0.66
3	おく	0.70	学情ナビ	0.63
4	面接官	0.70	エントリーシート	0.62
5	受け答え	0.69	するどい	0.61

ば大きいほど良いとされているが，推薦システムに用いる分散表現は必ずしもその限りではない．それは対象となるシステムの内容に合わせて，検索時の入力の類似単語がシステムに適していなければならないからである．そこで本研究では，作成するコーパスの内容を内定報告書閲覧システムに蓄積された報告書の中身の後輩へのアドバイスと本学の学生が自身のキャリアについて記述したキャリアポートフォリオの 2 つを用いた．通常 Wikipedia やスクレイピングによって web 上から取得したデータを使用するが，先行研究[4]によってキャリアポートフォリオを活用した場合の分類精度が良かったことと，コーパスが全体で 5MB（前処理後）であるため学習時間が短いことを考慮してキャリアポートフォリオを使用した．分散表現の学習のために用いたコーパスについては，**表 1** にまとめた．後輩へのアドバイスに比べるとキャリポートフォリオデータは量も語彙数も多いが，1 つの文書あたりの文字数は少ない傾向がある．

また，後輩へのアドバイスとキャリポートフォリオデータをまとめたコーパスから分散表現を学習させた結果を**表 2** に示す．過去の評価実験にて，入力される頻度の高かった単語の「面接」,「自己 pr」の分散表現とコサイン類似度が高い単語を示す.「面接」の類似語として筆記試験や spi,「自己 pr」には志望動機や履歴書が上位類似語として出現している.自己 pr と志望動機のコサイン類似度が他の単語の組み合わせよりも高い数値を示しているが，これはこの 2 つの単語がセットで使われることが特に多いためである．

学習した分散表現モデルを利用して，検索時の入力と報告書の類似度を計算する．本研究ではコサイン類似度と TFIDF という指標を用いて検索時の入力と報告書の類似性を数値化する．コサイン類似度はベクトル同士の成す角度の近さを表現する類似度計算手法で，式 (1)のようにして計算する．

$$\mathrm{cosSim}(w_q, w_d) = \frac{w_q w_d}{|w_q||w_d|} \quad (1)$$

w_q，w_dをそれぞれ入力時のクエリの分散表現と報告書の後輩へのアドバイスの分散表現とする．

TFIDF は文書中の単語の特徴を重みづけする手法である．本研究で TF は報告書d内の単語vの出現頻度を示す（式 (2)）.

$$TF(v, d_i) = \frac{n_{v,d}}{\sum_{e \in d} n_{e,d}} \quad (2)$$

$n_{v,d}$はある報告書d内の単語vの出現回数を示し，$\sum_{e \in d} n_{e,d}$は報告書d内の全単語数を示す．IDF はある単語が出現する報告書頻度を逆数とした値で示す．TFIDF は式 (2)に IDF の値を掛け合わせた値である(式 (3))．

$$TFIDF(v, d_i) = TF(v, d_i) \times \log \frac{N}{DF(v)} + 1 \quad (3)$$

$DF(v)$はある単語vが出現する報告書数で，Nは全報告書数である．入力と報告書の類似度を計算するために，入力に対して出力される類似単語の類似度を活用する．ただし，同じ単語が複数ある場合には重複を削除した．

$$simSum(v, d_i) = \sum_{e \in d} cosSim(W_v, W_{d_i}) \times TFIDF(v, d_i) \quad (4)$$

ここで，W_vは単語vの分散表現である．vが入力単語自身の時にはコサイン類似度を 1 とする．式 (4)によって同じような文章量の場合でも TFIDF によってよくある文章の場合には$simSum$は小さくなる．$simSum$の式について実際の報告書を例にして**図 1** に示す．

3.3 報告書の質

分散表現によって入力のイメージと近い単語を含む報告書を取得することができるが，その報告書の中には内容が似ているだけでありがちな文章であることがある．利用者にとっては推薦される報告書は入力と似ているだけでなく報告書の中身が伴うことで本当に役立つ情報を推薦することができると考える．本研究では報告書自体の質の高低を予測するためにクラス分類手法を活用する．クラス分類のためには特徴量と呼ばれる，予測したい項目を説明するための変数を作成する必要がある．本研究では Sugoole に蓄積された報告書の内容を参考に作成した．

Sugoole に蓄積されている報告書の内容を**表 3** に示

表 3　報告書の内容

項目名	概要
報告書登録日	報告書を作成した日付
会社名	報告書登録者が受験した会社名
本社所在地	受験した会社の本社所在地
職種	〃　の職種
業種	〃　の業種
後輩へのアドバイス	後輩に対するアドバイス
内定獲得日	内定を獲得した日付
各種選考日程	説明会や，筆記試験，面接などの日付と内容を記述
その他応募した企業	報告書の登録者がエントリーや選考に進んだ他の企業
筆記試験内容	その企業での筆記試験の内容
面接質問事項	各種面接を通しての全ての質問項目
内定獲得有無	内定の獲得の有無

表 4　説明変数と概要

説明変数	概要
Count_selection	選考の回数
Report_created_datetime	報告書を登録した日付
Today_created_diff_days	〃　から経過した日数
First_final_diff_days	選考にかけた日数
Word_length	報告書の後輩へのアドバイスの文章長
TFIDF_sum	報告書の後輩へのアドバイスに記述された文章の単語の TFIDF の合計
Count_ident_word	1 人称単語の出現回数
Count_NE_word	固有名詞の出現回数
Count_word	前処理後の報告書の後輩へのアドバイスの単語数

表 5　各種クラス分類手法の予測・分類精度

	低い評価の報告書			全体
	適合率	再現率	F 値	予測精度
svm(SVC)	0.66	0.86	0.74	0.67
RandomForest	0.68	0.85	0.75	0.68
MLPClassifier	0.66	0.84	0.74	0.67

す．企業に関する基礎的な項目に加えて，後輩に対してのアドバイスや選考に関する情報が記述されている．

また報告書を記述した学生が受験した他の企業や筆記試験の内容，面接時の質問内容などが記述されている．これらの項目から，自然言語処理や日付の差分を取得することで**表 4** の説明変数を作成した．

表 4 の各説明変数の作成手順については以下の通りである．

■Count_selection…各種選考日程に記述された説明会を含める試験の回数をカウント

■Report_created_datetime…報告書登録日を年月までの 6 桁に変更（例：201801）

■Today_created_diff_days…前処理にて特徴量作成時の日付から報告書登録日の差分の日数を取得

■First_final_diff_days…各種選考の最後の選考の日付から最初の選考の日付の差分の日数を取得

■Word_length…報告書の後輩へのアドバイスの前処理前の文章の長さをカウント

■TFIDF_sum…報告書の後輩へのアドバイスの前処理後の分かち書きされた単語に付与された TFIDF の値の合計を取得

■ Count_ident_word…3.1 節で説明したコーパスによく含まれる 1 人称の単語の,「わたし, 私, 僕, ぼく ，俺，おれ，自分，じぶん」が報告書の後輩へのアドバイスに含まれている回数

■Count_NE_word…報告書の後輩へのアドバイスの固有名詞の単語数

■Count_word…分かち書きあとの報告書の後輩へのアドバイスの単語数

過去取得した評価データを見たところ，評価の高くなる報告書は自身の体験談を具体的に書いている傾向があった．そこで，1 人称の単語が多く使われる文章は自身の体験談を記述するとして,「わたし, 私，僕，ぼく，俺，おれ，自分，じぶん」をカウントし，Count_ident_word とした．また，形態素解析ツールの mecab[4]と，新語や未知語に対応するための辞書である mecab-ipadic-NEologd を組み合わせることで，固有名詞をカウントして Count_NE_word とした．以上のようにして作成した特徴量を入力としてクラス分類手法へと適用した．クラス分類手法による分類精度や評価実験で使用するクラス分類手法については 4 章にて説明する．

4. クラス分類手法の選定

本研究では報告書の質を予測するために，2018 年 10 月 30 日までの評価データ 456 件を利用した．評価されたデータの点数の上位 4 割を 1，下位 6 割を 0 として 2 値の正解データを付与した．ただし，利用者毎に評価点数の傾向が異なるため，利用者毎に z-score 法を用いる事で平均を 0，標準偏差を 1 とするように標準化を行なった．これにより評価を低くつけがちな利用者と高くつけがちな利用者の評価点数を調整した．テストデータは 3 割とした．これにより訓練データは 327 件，テストデータは 141 件として，訓練データを SVM（SVC），RandomForestClassifier，MLPClassifier の 3 つの分類手法で学習させた．また，訓練データの学習時にはモデルのパラメータ探索方法であるグリッ

[4] http://taku910.github.io/mecab/

表 6　報告書の類似性と質に着目した評価結果の違い

	*simSum*の評価		*simAndQuolity*の評価		t 値	有意確率
	平均	標準偏差	平均	標準偏差		
報告書の類似性	2.31	1.08	2.51	1.08	-2.25	0.02

ドサーチをした．グリッドサーチは複数のパラメータのすべての組み合わせを試す方法である．グリッドサーチによって，最も予測精度の高いパラメータの組み合わせによるモデルを分類手法毎に構築した．検証したパラメータは以下の通りである．

【SVM】
'C': 0.001, 0.01, 0.1, 1, 10, 100
'γ': 0.001, 0.01, 0.1, 1, 10, 100
【RandomForestClassifier】
'max_depth': 2, 3, 4, 5, 6, 7, 8, 9
'n_estimators': 10, 100, 1000
【MLPClassifier】
'hidden_layer_size': (100),(200),(100,100),(200,200)
'learning_rate': 'invscaling', 'adaptive', 'constant'
' activation': 'logistic', 'identity', 'relu', 'tanh'

テストデータに対する予測精度と，適合率，再現率，F 値による分類精度を**表 5** に示す．**表 5** は 10 回分の学習結果の平均を取得している．また，適合率，再現率，F 値については，評価が低いと付けられた下位 6 割の報告書の分類精度の数値である．

表 5 より，各分類手法間で大きな差は出なかった．しかし，適合率と F 値，予測精度においてRandomForestClassifierの学習結果が他手法に比べて微小ながらも高い数値となった．また，RandomForestClassifierはパラメータのチューニングがほとんど必要ないことも，今後，報告書推薦システムを運用していく中での利点としてあげられたため，本研究ではクラス分類手法としてRandomForestClassifierを使用することとした．ここで，ある報告書d_iを評価の高い報告書であると予測する割合$probability(d_i)$を示す．（式 5）．

$$probability(d_i) = \frac{\sum_{n=1}^{t} \frac{c_n}{a_n}}{t} \quad (5)$$

c_n，a_nは決定木によって最終的な分類先の予測クラスのサンプル数と全サンプル数を示す．tはRandomForestClassifier で作成する決定木の数である．

ただし，*probability*は 0.5 未満の場合には 0 とした．*probability*の閾値 0.5 は**図 2** を参考に，著者が報告書を読み，決定した．**図 2** は Sugoole の全報告書のprobabilityを示した．縦軸は報告書数で横軸はprobabilityである．

図 2の分布は大きく分けて3つの分布になっている．

表 7　推薦式ごとの説明変数の中央値

説明変数	simSum	simAndQuality
Count_selection	3	4
Report_created_datetime	201403	201505
Today_created_diff_days	1769	1330
First_final_diff_days	24	39
Word_length	135	379
TFIDF_sum	5.1	7.7
Count ident word	0	2
Count_NE_word	4	12
Count_word	41	106

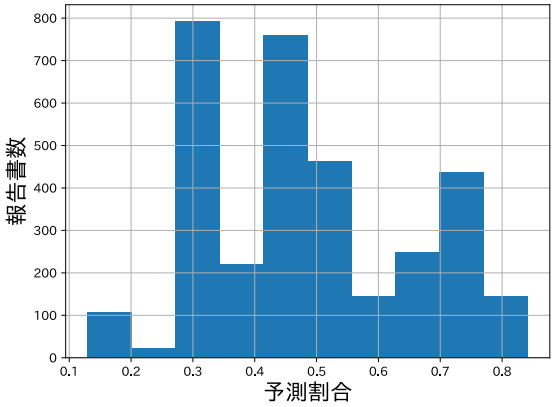

図 2　報告書の予測割合

①　0.4 未満
②　0.4 以上 0.5 未満
③　0.5 以上

①は報告書の中でも誰もが記述しそうな文章，③は報告書の中でも特に参考になる報告書，②については良いかどうかが曖昧な中間の報告書であると仮定した．著者がそれぞれの分布から 50 件ずつランダムに報告書を抽出し実際に読んだ．入力がどんな場合であったとしても，就職活動に参考になるかどうかについて着目した．①と③については仮定した通りで Sugoole に登録された報告書の中でも記述量や内容が明らかに異なる報告書であった．例えば，①の分布の報告書は未記入や「後悔ないように頑張ってください」，「就職活動は意欲が大事」のようなアドバイスとなっていた．③は自分の経験談を語ったり，面接についてより詳細に記述していたりと後輩に読んでほしい報告書であった．しかし，②については①と同等の報告書と，③の分布の報告書と比べて，参考になる報告書が少なかった．そのため，本論文では以上のことを踏まえて，①の報告書と，良いかどうかが曖昧な報告書である②の

分布の報告書については推薦時の重みを 0 とし，③の報告書は，そのままの予測割合を推薦の重みとすることで，質が高いと判断することが難しい報告書を推薦から外すようにした．

5. 評価実験

5.1 推薦式の推敲

評価実験には報告書の類似性と質に着目して比較するために，本研究では 2 つの推薦パターンを構築した．1 つ目は報告書の類似性のみを考慮する式 4 の $simSum$を用いた式である．2 つ目は類似性と質を考慮する式として，クラス分類手法によって各報告書の評価が高くなると予測された割合と $simSum$との和をとった（式 6）．

$$simAndQuolity = simSum(v, d_i) + probability(d_i) \quad (6)$$

式 6 に使用される$simSum$は$probability$のスケールと合わせるために，全ての報告書で計算された$simSum$の値を min_max 法の利用によって最小値を 0，最大値を 1 とする正規化を行った．

5.2 実験手順

式 4，6 を使用して評価実験を実施した．報告書毎に付与する評価の高いと予測される確率についてはランダムフォレストを使用する．Sugoole の利用者の多くが就職活動を控えた学生や就職活動中であることを考慮して，就職活動前の学生 5 人，就職活動を終えた学生 1 人の合計 6 人を対象とした．実際に就職活動を終えた学生にしか気付けない部分もあると考え対象に含んでいる．

評価は以下の手順で実施した．

1. どちらか 1 つの推薦式を選択したら就職活動に関するキーワードを入力する
2. 推薦された報告書に対して，ランダムに指定された順位の報告書を読んで評価を 10 件分つける（この時，指定された番号がすでに評価されている場合は順位が近い報告書を読んで評価をする）
3. キーワード 3 つ分の評価をするまで 1 に戻る
4. もう片方の推薦式で 1～3 を実行する

被験者がどんな式で推薦されているか分からないように毎回ランダムに数値を付与して推薦の式を選択してもらった．報告書を読んだ後に評価する点数は 0.1~5.0 までの 0.1 刻みでつけてもらった．また，評価する際に個人で点数にばらつきが出過ぎてしまうことを抑えるために評価の基準を設けた．

0.1~2.0 までは検索時の入力と報告書が類似しておらず，参考にもならなかった場合とした．2.1 以上は参考になった報告書とした．2.1~3.0 までは検索時の入力と報告書が類似していない，3.1~4.0 までは検索時の入力と報告書が類似している，4.1 以上は検索時の入力と報告書が類似していて，他の学生にも読んでほしいぐらいにお勧めする事ができる報告書として評価してもらった．

以上の手順でキーワード 3 つ分を 1 セットとして，1 人の被験者が 1 もしくは 2 セット分の評価をした．1 セットで 60 件の評価を得る事ができる．

5.3 実験結果

被験者 6 人による評価実験を実施して，欠損データや評価の途中で検索のキーワードを変更したデータの削除を行なった後，全部で 480 件の評価データを取得した．また，評価基準を設けてはいるが，それでも被験者ごとに点数のばらつきがあるため，z-score 法による標準化を行なった．

推薦の式 4，6 の間で統計的な有意差が見られるかを検証するために，式 4，6 の評価点数を被験者毎に標準化した点数を用いて t 検定を実施した．2 つの式の評価については**表 6** にまとめた．t 検定の結果が$p = 0.02$（$p < .05$）で，式 4，6 において有意差が認められた．この結果，検索時の入力と報告書の類似性と，報告書自体の質を考慮した式 6 の結果が，式 4 の類似性を考慮した推薦よりも評価が高くなる事がわかった．

6. 考察

t 検定によって式 4, 6 の間に有意差が認められたが式の違いで推薦される報告書の特徴量にどのような変化があったかに着目しつつ考察する．式 4，6 の推薦を機械学習の説明変数に着目して，各説明変数の中央値を**表 7** にまとめた．

検索時の入力と報告書の類似性と報告書の質に着

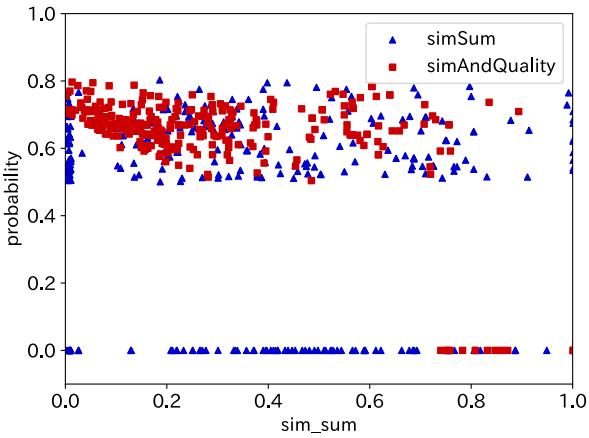

図 3 $simSum$と$simAndQuolity$の予測確率の分布

目して推薦を行なった$simAndQuolity$の場合には，Word_length や Count_word などの文章量に関する数値が大きくなっている．これに Count_NE_word や TFIDF_sum の数値も大きくなっている．過去の評価データを学習に用いているが，文章量が多いものは内容が濃い報告書である場合が多いため，式 6 の$probability(d_i)$の値が高くなっていると考えられる．また，Count_selection や First_final_diff_days の値も$simAndQuolity$の方が大きくなっていることから，面接の多い企業ほど内定獲得が難しくなる事が予測でき，報告書の内容も内定獲得のための方法や経験談が語られる事が考えられる．

式 4 の場合には$simSum$のみで推薦を行なっているが，式 4 で推薦された報告書の予測確率がどうなっているのかを調査するために，縦軸に$probability$，横軸に$simSum$を取ったグラフを**図 3** に示した．三角の青色でプロットされているデータが式 4 の場合のデータで，四角の赤色でプロットされているデータが式 6 の場合のデータである．三角のプロットがなされた式 4 で推薦・評価された報告書の高いと予測する割合が 0.5 未満となり 0 となっている報告書が多いことがわかる．つまり，式 6 で推薦をした場合には$probability$が 0 となって推薦の重みが低くなった報告書が多数あることがわかる．本研究において最も検索のキーワードとして多かった「面接」の場合の推薦について考える．式 4 にて推薦順位が 3 位の報告書 A が持つパラメータは以下の通りであった．

- Count_selection : 3
- Report_created_datetime: 200706
- Today_created_diff_days: 4214
- First_final_diff_days: 22
- Word_length: 77
- TFIDF_sum: 4.0
- Count_ident_word: 0
- Count_NE_word: 5
- Count_word: 21
- 後輩へのアドバイス:

  ・決め手は，面接で相手の質問にきちんと答えられたこと

  ・ポイントは早めに就活の準備を！！

  SPI，Web テスト，自己分析，履歴書の書き方，面接で注意すること．

報告書 A は$probability$の値が 0 であったため，式 6 の推薦の場合には 100 位以内にすら入らなかった．

対照的に式 4 にて 34 位に推薦された報告書 B が，式 6 において推薦順位が上がって 13 位となる場合があった．報告書 B が持つパラメータは以下の通りであった．

- Count_selection : 3
- Report_created_datetime: 201609
- Today_created_diff_days: 833
- First_final_diff_days: 28
- Word_length: 228
- TFIDF_sum: 5.6
- Count_ident_word: 3
- Count_NE_word: 7
- Count_word: 74
- 後輩へのアドバイス:

  説明会に参加し，自分がこの会社で何をしたいのか明確にすることが大切．選考ステップの中でＳＰＩや会社独自の筆記試験等に自分はとても苦労したので，甘く見ていると痛い目に合うので，事前の対策をしっかりしておくことが大切です．また，自分の性格から人前で話すことが苦手なので，面接対策を事前にしておくことで，本番では落ち着いて臨むことができた．面接対策を事前にしておくことで，本番では落ち着いて臨むことができた．面接対策で自分自身の性格をよく理解しておくことが大切．

報告書 B は$probability$の値が 0.63 であった．報告書 B の Count_ident_word と Count_NE_word が報告書 A よりも多く，報告書 B の登録者自身が体験した内容が記述されている．RandomForest で学習したモデルが持つ決定木をいくつか確認したところ，Count_ident_word や Count_NE_word が一定数より下回った時や一定数より上回った時に分類が変わる条件が見受けられた．ただし，文章量が多くなることによって Count_ident_word や Count_NE_word などのカウントは増えることも考えられるので注意が必要である．

7. まとめ

本研究により，分散表現を用いた検索時の入力と報告書の類似性と，クラス分類による報告書の質を考慮する推薦の有意性を示した．

今後の課題としては，クラス分類手法の予測・分類精度向上が挙げられる．本研究では報告書内の後輩へのアドバイスから，固有名詞や 1 人称単語をカウントして数値化したが，固有表現や，誰でも書ける内容では無いことが判定できる特徴量を作成すると予測・分類精度向上を見込めるだろう．また，報告書に記述される項目の中で，その他応募した企業，筆記試験内容，面接質問事項などは特徴量として，本研究では含めていない．報告書を登録した学生が応募している企業群の報告書の評価や，面接質問が珍しい企業，Sugoole に未登録の企業，推薦での内定かどうかなどによって特徴量を増やすことができる可能性がある．

本研究では検索時の入力単語は自由に任せているが，そもそも Sugoole に蓄積されている報告書の中に，調べたい事柄を記述している報告書が少ない，もしくは無い場合がある．こういった場合に，他の単語での検索へと誘導したり，利用者がどのような情報を欲しがっているのか前もって調査したりしておくことなどが考えられる．

本推薦システムは 2019 年 1 月より本番環境にて運用を開始している．

参 考 文 献

[1] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean, “Distributed Representations of Words and Phrases and their Compositionality”, In NIPS, pp.3111-3119, 2013.

[2] Jey Han Lau, Timothy Baldwin, “An empirical evaluation of doc2vec with practical insights into document embedding generation”, arXiv preprint arXiv:1607.05368, 2016.

[3] J. Lilleberg, Y. Zhu, Y. Zhang, “Support vector machines and word2vec for text classification with semantic features”, In Proc. of IEEE ICCI*CC, pp.136–140, 2015.

[4] 赤木里騎，徐海燕，“大学が有する潜在文書を活用した報告書推薦システムの構築”，情報処理学会研究報告，Vol.2018，IFAT-132，No.26，2018.

[5] 齋藤祐樹, 田頭 幸浩, 小野 真吾, 田島 玲,“検索における分散表現を用いた類似度定量化”，DEIM Forum2016，C1-6，2016.

[6] 鈴木類，古宮嘉那子，浅原正幸，佐々木稔，新納浩幸，“「分類語彙表」の類義語と分散表現を利用した all-words 語義曖昧性解消”，言語資源活用ワークショップ発表論文集，pp.258-264，2017.

[7] 開地亮太，檜垣泰彦，“観光地推薦システムへの単語分散表現の適用”, 信学技報, Vol.116, No.488, LOIS2016-85，pp.129-134，2017.

[8] V.N. Vapnik, “An Overview of Statistical Learning Theory”, IEEE Transactions on Neural Networks, Vol.10，pp.988-999，1999.

[9] L. Breiman, “Random forests”, Machine Learning, Vol.45，pp.5-32，2001.

[10] M. W.Gardner, S. R Dorling, “Artificial neural networks (The multilayer perceptron) - A review of applications in the atmospheric sciences.”, Atmospheric Environment, Vol.32，pp.2627-2636，1998.

[11] 長谷川新，相沢明子，浜本隆之，“自然情報を用いたユーザプロファイル獲得と文書推薦”，情報処理学会自然言語処理研究会研報, Vol.194, No.8, pp.1-6，2009.