F ターム概念ベクトルを用いた特許検索システムの改良

目黒 光司 † 笹野 遼平 †† 榊原 隆文 † 菊池 悠太 †
高村 大也 †† 奥村 学 ††
東京工業大学 総合理工学研究科 †
東京工業大学 精密工学研究所 ††
{meguro, tsakaki, kikuchi}@lr.pi.titech.ac.jp
{sasano, takamura, oku}@pi.titech.ac.jp

1 はじめに

特許文献の検索に関する研究は，従来より数多く行われ，様々な検索システムが提案されている [1] [2]．特許検索は，調査の目的によって検索の性質が異なっている．例えば，審査段階においては，審査案件における請求項などの記載から，審査官や検索外注機関のサーチャーが検索ワードを抽出し，さらに特許分類などを用いて検索クエリを構成している．しかし，これらの作業は，対象分野に関する高度な知識や検索ノウハウが要求される．本研究は，検索クエリを必要とせずに，明細書を入力するとその内容に類似した特許文献を検索する類似文献検索システムを提案するものである．

特許検索には，特許特有の技術用語の表記揺れの影響を受けずに，同じような技術思想が開示されている先行技術文献を見つけ出すことが求められている．しかし，文献中の単語に基づいて文献間の類似度を計算する手法では，技術用語の表記揺れの影響を受ける．このため，同じような技術思想が先行技術文献中に開示されていても，文献間に出現する単語が互いに異なっていた場合，互いの類似度が低くなる可能性がある．また，出現する単語パターンが似ていると，技術思想が異なる場合であっても類似度が高くなってしまう可能性がある．

そこで，本研究では「F ターム」という日本の特許文献に人手で付与されている分類記号に着目し，特許文献における発明の「目的」,「手段」,「用途」といった概念を数値化し，特許文献に対応する 100 から 500 次元程度の概念ベクトルを生成し，文献間の類似度を計算する手法を提案する．

2 特許分類（F ターム）

F ターム [1] とは日本独自の特許分類であり，発明の内容を目的，材料，手段，用途など複数の観点で展開，細分化している．F タームは，アルファベットと数字の 5 文字で表されるテーマコード（e.g., 2H200）と，アルファベットと数字の 4 文字からなる観点（e.g., FA01）で構成されている．テーマコードは，全部で約 2000 コード存在し，各観点の数はテーマコードによって様々で，1 つのテーマコード内に概ね 100 から 500 個存在している．また，1 つの特許文献に対して F ターム観点は数十個程度付与されている．F タームの例を図 1 に示す．

2H200	電子写真における帯電・転写・分離
	G03G13/02;13/14-13/16;15/02-15/02,103;15/14-15/16,103

観点						
FA	FA00	FA01	FA02	FA03	FA04	
	目的	・環境変化への対応	・経時変化対策	・帯電メモリ対策	・転写ずれ対策	・・ス
		FA11	FA12	FA13	FA14	
		・安全対策	・メンテナンス、設定容易	・製造容易	・リサイクル容易、再利用	・・
GA	GA00	GA01	GA02	GA03	GA04	
	前提とする装置全体の構成	・転写材（除く中間転写体）に関する開示	・・転写材の長さに言及するもの	・・転写材の幅に言及するもの	・・転写材の厚さに言及するもの	・度
		GA11	GA12	GA13	GA14	

図 1: F タームリスト（PMGS より）

3 F タームに基づく特許文献間類似度の計算

本研究では，提案手法 1 として，各特許文献に対する F ターム観点の付与されやすさを数値化した F ターム概念ベクトルを作成し，特許文献間の類似度を計算

[1] F タームリストは，特許電子図書館（IPDL）の PMGS から入手できる．http://www5.ipdl.inpit.go.jp/pmgs1/pmgs1/pmgs

する．さらに，提案手法 2 では F ターム観点間の重みを調整し，重み付き F ターム概念ベクトルを作成し，特許文献間の類似度を計算する．

3.1 F ターム概念ベクトルの生成

特許文献には人手で F タームが付与されており，各 F ターム観点の付与の有無が 2 値で表されている．しかし，実際の特許文献を見てみると，ある F ターム観点が明らかに付与されるべきとすぐに判断できる場合と，付与すべきか悩むような微妙なケースが存在する．そこで，本研究では F ターム観点の付与を 2 値ではなく，連続的なものとして捉え，F ターム観点の付与されやすさを数値化し F ターム概念ベクトルを作成する．テーマコード内の F ターム観点を n 個選べば，特許文献は n 次元のベクトルで表されることになり，各次元の値は技術分野特有の「目的」,「手段」,「用途」等の概念を数値化したものとみなせる．

F ターム観点の付与されやすさを数値化する手法としては，様々な手法[2] が考えられるが，本研究では，あるテーマコードが付与されている特許文献を各 F ターム観点毎に分け，各観点毎に SVM 分類器を作成し用いる．SVM 分類器の学習には，F ターム観点が付与されている特許文献を学習データとし，それら文献中に出現する形態素 uni-gram を素性に用いる．

そして，特許文献 j の形態素 uni-gram で表された事例ベクトル x_j に対する各観点 i 毎の SVM 分類器の出力値 $f^i_{svm}(x_j)$ をシグモイド関数に渡し，出力値が -1 から 1 の範囲となるように係数 2 をかけて，文献 j の F ターム概念ベクトル $doc_j[i]$ を以下のように定義する:

$$doc_j[i] = 2 * \left(\frac{1}{1 + exp(-f^i_{svm}(x_j))} - 0.5 \right). \quad (1)$$

すなわち，文献 j におけるベクトル成分 i は，文献 j に対する F ターム観点 i の付与されやすさを表している．

3.2 重み付き F ターム概念ベクトルの生成

本研究ではさらに，提案手法 2 として F ターム観点の間に重み付けを行った．F ターム観点には，テーマコード内のほとんどの文献に付与されている出現頻度の高いものと，出現頻度の低い特徴的な分類が存在している．ここで，単純にどの文献にも付与されている F ターム観点の重みを一律に小さくしてしまうと，出現頻度の高い F ターム観点が付与されていないという特徴的な状況でも重みが小さくなってしまう．同様に，出現頻度が低い F ターム観点の重みを一律に大きくしてしまうと，出現頻度が低い F ターム観点が付与されていないという当たり前の状況でも重みを大きくしてしまう．そこで，提案手法 2 では、提案手法 1 において出現頻度が低い F ターム観点が「付与されやすい」と判断されている場合と，出現頻度の高い F ターム観点が「付与されにくい」と判断されている場合に，F ターム観点の重みを大きくする方向に調整し，逆に，出現頻度が高い観点が「付与されやすい」と判断されている場合と，出現頻度が低い観点が「付与されにくい」と判断されている場合に，F ターム観点の重みを小さくする方向に調整する．

本研究では，F ターム観点 i が付与されている文献数 m_i とテーマコード内の全文件数 N を用いて，F ターム観点の重み $w[i]$ を定義し，重み $w[i]$ を提案手法 1 の F ターム概念ベクトルの各成分に掛けあわせ，提案手法 2 における重み付き F ターム概念ベクトル $doc_j[i]'$ を以下のように定義する:

$$doc_j[i]' = w[i] * doc_j[i], \quad (2)$$

$doc_j[i] \geq 0$ の場合,

$$w[i] \quad = \quad log(N/m_i + 1), \quad (3)$$

$doc_j[i] < 0$ の場合,

$$w[i] \quad = \quad log(N/(N - m_i) + 1). \quad (4)$$

3.3 文献間の類似度の算出

特許文献 j_1，j_2 の類似度の算出は，以下で定義される余弦類似度により計算する:

$$score(j_1, j_2) = \frac{\sum_{i=1} doc_{j_1}[i] * doc_{j_2}[i]}{|doc_{j_1}[i]||doc_{j_2}[i]|}. \quad (5)$$

4 評価実験

4.1 特許データと評価手法

本研究で使用する特許データと評価手法は以下のとおりである．

1. 特許データ 実験で使用する特許データは，1994 年から 2013 年までの公開特許公報のうち，G03G15/16 が付与されているもの 22,465 件と，G03G15/20 が付与されているもの 23,895 件の 2 テーマ用いた．

[2] 例えば，ナイーブベイズ分類器，F タームを用いた半教師あり LDA などが考えられる．

G03G15/16 には，F タームテーマコード 2H200 が対応し，G03G15/20 には，テーマコード 2H033 が対応している．なお，今回の実験では，F ターム概念ベクトルの次元数を 2H200 では 317 次元，2H033 では 190 次元とした．

2. 評価セット　評価セットとして，特許庁の審査官が審査において新規性を否定する拒絶理由通知書を少なくとも 1 回は通知した審査案件と，その審査案件の拒絶理由通知書で引用されている引用文献を用いた．なお，引用文献には，審査案件に対して審査官が新規性を否定するために引用した文献以外に，進歩性を否定するために引用した文献や参考文献等が存在する場合があるが，それらの除去は行っていない．

実験では，G03G15/16 内の審査案件 462 件とその引用文献 1,657 件，G03G15/20 内の審査案件 616 件とその引用文献 2,331 件を用いた．

3. 評価手法　評価は，審査案件を入力とし，同一テーマが付与されている特許公報を類似度順にランキングした場合に，引用文献がどのくらい上位にランキングされるかによって評価する．特許検索では，通常数百件の文献を吟味するため，上位 10 位未満における順位の変動や，10,000 位から 5,000 位への順位の変動よりも，適合文献の順位を 1,000 位から 200 位以内に改善することに意義がある．そこで，本研究では，各審査案件に対して類似度を算出した際に，審査官が引用した引用文献が 50 位，100 位，200 位以内に入った件数により検索システムを評価する．

4.2　比較手法

比較手法として，一般的に使われている TF-IDF に基づく検索方式と，LDA [3] により教師無し学習でトピック分布ベクトルを作成し，トピック分布ベクトルの類似度の計算による検索方式を使用する．ただし．LDA を用いた検索方式は，トピック数を様々な値に変化させ最も精度が高いものを採用する．

4.3　実験結果

実験結果を表 1 に示す．表 1 は，各手法で審査案件毎に文献間類似度を計算し，その類似度上位 50 位，100 位，200 位までに入った審査官引用文献数を示している．なお，以下では，提案手法 1 を F-vec1，提案手法 2 を F-vec2 と表す．

	2H200			2H033		
	50	100	200	50	100	200
TF-IDF	229	329	464	316	466	681
LDA	177	264	387	257	381	549
F-vec1	167	250	363	207	286	411
F-vec2	231	334	469	278	439	624

表 1: 実験結果

表 1 に示すように，F-vec1 よりも F-vec2 のように F ターム観点間に重み付けを行った方が，審査官が引用した文献を高順位に出力する結果となった．また，F-vec2 と LDA では，F-vec2 の方が審査官が引用した文献を高順位に出力する結果となった．F-vec2 と LDA は，いずれも分類やトピックを考慮した素性を利用する点で類似すると考えられるが，F-vec2 は，専門家が人手で分類を付与した F タームデータを用いて学習しているため，このようなデータを用いない LDA より優れた結果になったものと考えられる．

ここで，個々の審査案件における審査官引用文献の出力順位を精査すると，次のような特徴がみられた．F-vec2 において，高いスコアの審査官引用文献は，F-vec1 においてもやや劣るが高いスコアになる傾向がある．つまり，F-vec1 と F-vec2 は同様の特性に基いて文献間類似度を計算していると考えられる．

一方，F-vec2，TF-IDF を比較すると，F-vec2 において，高い類似度の審査官引用文献であっても，TF-IDF では，類似度が低くなることがあり，逆に，TF-IDF において，高い類似度の審査官引用文献であっても，F-vec2 では，類似度が低くなることがあった．このことから，F-vec2 と TF-IDF に基づく検索手法は，一見，同程度の性能の検索システムに見えるが，異なる特性に基づいて文献間のスコアを算出していると考えられる．

そこで，互いの検索手法を補完し合うことで検索精度が向上すると考え，F-vec2 と TF-IDF，F-vec2 と LDA を組み合わせた手法を用いた実験も行った．また，比較のため，既存手法の TF-IDF と LDA で算出した類似度を組み合わせた実験も行った．なお，各手法のスコアは，スケールが異なっている可能性があるので，各手法の類似度に重み [3] を付けてから掛けあわせた．

表 2，および，図 2，図 3 に実験結果を示す．図 2，図 3 は，検索上限数 r を 1 件から 200 件まで変化さ

[3] 2 つの手法を組み合わせる場合は，以下の式によりパラメータ δ の値を調整する．また，F-vec2 は正のスコアのみ用いた:

$$score \quad = \quad {score_1}^{\delta} \times {score_2}^{1-\delta}.$$

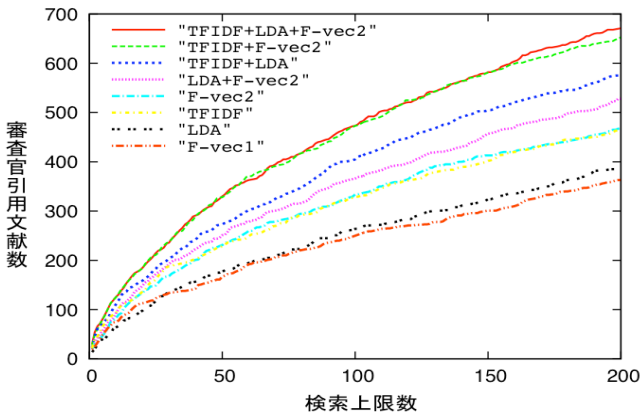

図 2: 適合文献の推移 2H200

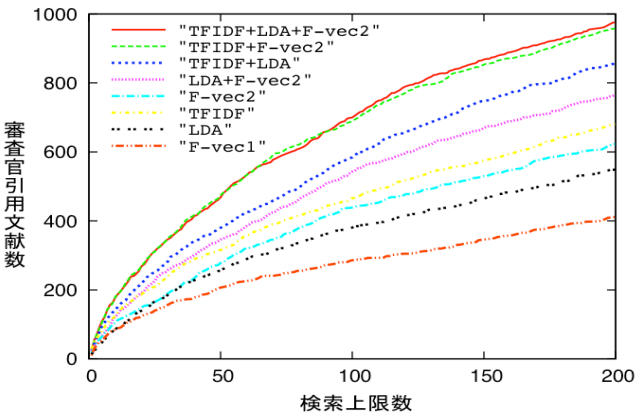

図 3: 適合文献の推移 2H033

	2H200			2H033		
	50	100	200	50	100	200
TF-IDF	229	329	464	316	466	681
LDA	177	264	387	257	381	549
$\text{TF-IDF}^{0.6}\text{LDA}^{0.4}$	273	405	576	380	584	857
F-vec2	231	334	469	278	439	624
$\text{LDA}^{0.2}\text{F-vec2}^{0.8}$	251	366	528	345	540	769
$\text{TF-IDF}^{0.4}\text{F-vec2}^{0.6}$	328	473	653	446	685	961
$\text{TF-IDF}^{0.3}\text{LDA}^{0.2}\text{F-vec2}^{0.5}$	330	473	671	468	700	976

表 2: 実験結果

せたときの，各手法における r 位までに入った審査官引用文献数の推移を表している．また，表 2 において各手法の右上に付いている数字は，重み δ を表し，$r = 200$ において最も審査官引用文献数が多くなるように調整し得られた値である．表 2 や図 2，図 3 に示すように，各手法を組み合わせることにより，ランキング 200 位以内に入る審査官引用文献数が向上した．

2 つの組み合わせの中では，TF-IDF と F-vec2 のペアが最も検索精度が高い結果となった．2 つの手法が同じような文献順位を出力している場合は，両手法を組み合わせても文献順位の入れ替わりが生じにくいことを考慮すると，TF-IDF と F-vec2 は異なる文献間の類似性を捉えていると考えられる．一方，LDA と F-vec2 のペアは，あまり検索精度が向上していない．これは，LDA と F-vec2 の作る文献間の類似性が TF-IDF に比べると近いためであると考えられる．特に，F-vec2 と TF-IDF の 2 つの組み合わせに対して，LDA を追加しても検索精度が向上していないことから，F-vec2 と TF-IDF の組み合わせに LDA が持つ情報が包含されていると考えられる．

5　まとめと今後の課題

本研究では，特許文献間の類似度計算において，F タームに基づく概念ベクトルを用いる手法を提案した．また，実際に，特許庁の審査官が，新規性を否定する拒絶理由書において引用した引用文献をどのくらい上位にランキングできるかという実験において，従来の単語に基づいた文献間の類似度と組み合わせることにより検索精度が向上することを示した．このことは，提案した F ターム概念ベクトルに基づく類似度が，単語に基づく手法では捉えられない類似性，すなわち，同じような概念が別の単語を用いて表現されている文献間の類似性を捉えているためであると考えられる．

また，提案する F ターム概念ベクトルは言語非依存なベクトルであり，異なる言語に対し同一の基準で F タームが付与されたデータがあれば，同様の性質を持つ概念ベクトルが生成可能である．このため，今後の課題として，異なる言語で出願された特許公報間の類似度計算への応用が考えられる．たとえば，日本語文献に対応する外国語のファミリー出願が多い技術分野では，日本の特許公報に付与されている F ターム等の分類を、外国語特許公報にも付与されるべき分類と見なし，F ターム付与の学習データとすることが可能である．このため，外国語文献に対しても F ターム概念ベクトルを生成し，日本語と中国語文献間や，英語と中国語文献間などの類似度を翻訳機を介さずに算出可能であると考えられる．

参考文献

[1] Mihai Lupu, Katja Mayer, John Tait, and Anthony J. Trippe. *Current challenges in patent information retrieval*. Springer, 2011.

[2] Atsushi Fujii, Makoto Iwayama, and Noriko Kando. Overview of the patent retrieval task at the ntcir-6 workshop. In *Proceedings of NTCIR-6 Workshop Meeting*, pages 15–18, 2007.

[3] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.