LDA が抽出する topics は本来意味なものではない？

幾つかの言語の語形の (L-)LDA を使った自己教師型分類の性能評価に基づいて

黒田　航 (杏林大学医学部)

1　はじめに[1)]

Latent Dirichlet Allocation (LDA) [1, 2, 3] は，文書集合 $D = \{d_1, d_2, \ldots, d_n\}$ が与えられた時に，D とそれらの構成要素 (e.g., D で使われている用語 (terms) (e.g., 単語) $T = \{t_1, t_2, \ldots, t_m\}$ の両方を生成する生成源を，教師なしで抽出する手法だと理解されている．生成源はトピック (topics) と呼ばれる．トピックは概念に対応すると理解されているが，これは「仮定により真」だと見なされているだけであり，理論的な根拠が十分にある訳ではない．実際，生成源がトピックと呼ばれるのは歴史的な経緯 [1, 2, 3] のためである．LDA の結果を見ると，トピック毎に帰属が決まる用語の集合は確かに概念に緩く対応しているように見える．だとしても，これはトピックが概念に対応している事の証拠にはならない．質的に異なる二つの変量に強い相関があり，それらの外延が重なっているだけなのかも知れない．

研究 [4, 5] は幾つかの言語，あるいは幾つかの分野の文字列 (単語の綴りや発音記号列) を文書とし，それらの文字 n-grams を用語とする設定で LDA を実行し，文字列を教師なしでクラスタリングできる事を示した．この結果は，LDA が抽出するトピックが概念的である必要がない事を示唆する．本研究は更に一歩進んで，**LDA が抽出するトピックは概念的な意味に対応する必要がない**という再解釈を提案し，その解釈の下で LDA が利用できる条件を再検討する．具体的には，研究 [4] のような教師なしクラスタリング[2)]に簡単な設定を追加するだけで，分類性能の自動評価が可能である事を示す．具体的には，複数の言語の綴りの集合を LDA 基盤でエンコードし，その結果に教師なしクラスタリングの結果を，元の言語ラベルを正解とする分類課題と見なして評価する事で，性能を測れる事を示す．

なお，この研究の目的は，LDA の動作原理とそれで得られる結果のより経験的に妥当な理解であって，任意の言語に属する語形の自動分類を高精度に実現する事ではない．それは LDA より強力な機械学習の手法を使えば実現できる可能性がある (本研究が採用したのと同じ自己教師あり条件で別の機械学習を実装すれば，より高精度の結果が得られる可能性は大きい)．

[1)] 研究の目的と意義の明確化に関して，匿名の査読者二名から頂いた評言が参考になった．この場を借りて謝意を表する．

[2)] 本研究で Hierarchical Dirichlet Process (HDP) [6] を使わなかったのは，そうすると document エンコードの精度が落ちるとわかったからである．`FastText` を使わなかったのはエンコード自体がうまく行かないと事前調査で判明していたからである．

2　方法

複数言語の単語の綴りの，LDA [1, 2, 3] を使った教師なしクラスタリングの性能評価を，次のような自己教師式 (self/auto-supervised) に実行する．1) 語形 D を文字 (skippy) n-gram (n = 1, 2, 3) でエンコードする．結果を E とする．2) E に LDA を実行し，その結果から語形を topic の分布としてエンコードする．結果を E' とする．3) E' を t-SNE で変換する[3)]．結果を F とする．4) F を DBSCAN でクラスタリングする[4)]．結果を C とする．5) C の精度を，クラスター毎の言語の dominance d と言語毎の purity p の F 値 $= 2 \times d \times p/(d + p)$ で評価する (d, p の定義は後述)．d, p はクラスターに幾つかの言語の語が混在しうる状態だから意味を持つ．

英語の綴りは，CMU Pronunciation dict [5)] からランダムに選んだ．他の言語の綴りは 1000 most common words[6)] のデータを利用した．

LDA の実装は `gensim` [7)] を利用した[8)]．Jupyter Notebook 上の実行コード (Python 3.10 以上での利用を想定) は GitHub site [9)] 公開してある．

LDA では通常，term として単語 token を与え，document を bag-of-words としてエンコードする．この方法を綴りに対応させると，“document : term = 単語 (の文字表記) : 文字” になるが，これでは term の document 中の出現位置がエンコードされない．これでは綴りに固有の表現を与えられない可能性が大きい．文字の (skippy) n-gram を使い，この難点を克服する．

文字の n-gram は包括的で，n-gram は再帰的に $(n-1)$-gram を含む (n の最小値は 2)．

評価が自己教師式だと言うのは，語形の教師なしクラスタリングに言語名という形で “正解” があり，それと別に正解を用意する必要がない．自己教師式クラスタリングの理想的な結果は次である: 得られたクラスターが $C = \{c_1, c_2, \ldots, c_n\}$ だとすると，A) どのクラスターの要素もただ一つのカテゴリー値 (e.g., 言語種) が割り当てられ，B) カテゴリーのどの値 v についても，v の要素

[3)] 本調査は次元圧縮に t-SNE を使ったが，他の次元圧縮法でも構わない (が，距離指標をうまく選ぶ必要がある)．PCA, MDS, PLS は力不足だと判明している．

[4)] クラスタリングはクラスターの個数を固定しない=最適化するタイプなら何でも良い．この調査では DBSCAN を使ったが，X-means でも原理的には問題ない (パラメータは手法ごとに最適化が必要)．

[5)] `http://www.speech.cs.cmu.edu/cgi-bin/cmudict`

[6)] `https://1000mostcommonwords.com/`

[7)] `https://radimrehurek.com/gensim/`

[8)] `gensim.models.LdaModel` か `gensim.models.LdaMulticore` のいずれかが選べる．Multicore 条件では後者の方が実行が早い．

[9)] `https://github.com/kow-k/self-supervised-word-classification`

を (一定以上) 含むクラスターの個数の，v を要素に含むクラスター全体数に対する割合が 1.0 である．本調査では具体的に，一定の大きさをもつクラスターについて，i) 最大の要素数を持つ言語 l を特定し，l の要素数のクラスターの要素数に対する割合を dominance d とし，これを A の指標とした．言語 l について，d が適当に設定した閾値 (0.33[10]) を越えるクラスターの個数の，l を要素に持つクラスターの個数に対する割合割合を purity p とした．d, p は相反する指標であり，d, p の F 値 = $2 \times d \times p/(d + p)$ をクラスタリングの良さの評価指標とした ($0 \leq \mathrm{F} \leq 1.0$).

2.1 探索範囲

探索したパラメーターは L, S, T, A, N の 4 つである:

(1) L: 言語の異なり: Arabic, English, French, German, Russian の 5 種類[11]

S: 言語毎の初期サンプル数: `sample_n_per_language` = [200, 800] (後処理で 1 割 ~3 割目減り)

T: LDA 用の term の異なり: [1-gram, 2-gram, 3-gram, skippy 2-gram, skippy 3gram]

A: LDA の term の濫用指数 a [0.10, 0.67, 0.90][12]

N: LDA の topic 数 [2, 3, 4, ..., 7, 10, 15]

この他に，次のパラメーターが結果に影響するが，今回の調査ではそれぞれ次の値に固定した．1. `max_doc_size` = 11 文字; `min_doc_size` = 5 文字; 2. LDA の元になる DTM で term を選択するための最低限の doc の異なり `min_freq` = 2; 3. t-SNE の perplexity = 3 `n_topics`; 4. DBSCAN の `min_samples_rate` =0.005; outliers の個数の許容度 0.03; 5. `purity_min_population_rate`= `0.007`; `purity_threshold` = 0.33.

3 解析結果

先に言語毎の初期サンプル数が 200 の場合の結果示し，初期サンプル数が 800 の場合の結果を続ける．また，$a = 0.90, a = 0.67, a = 0.10$ の，a が大きい順に得られた結果を示す．

3.1 初期サンプル数 200 の場合

濫用指数 $a = 0.90$ の場合の，$L \times T \times N$ の F 値を図 1 に示す．言語ごとに F 値が 1 位，2 位の組み合わせをオレンジ枠で囲って区別した (これは以下同様).

[10] 0.50 とすると purity を計算するクラスター候補が減り過ぎるので，より穏健な値の 0.33 を選んだ.

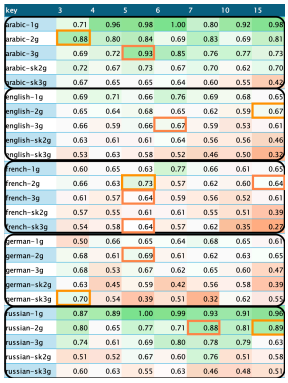
[11] 英語と似ている 2 言語 (e.g., フランス語とドイツ語) と文字種が異なる 2 言語 (e.g., アラビア語とロシア語) の条件で選んだ.

[12] gensim の LDA には token が docs 全体の a に現われている場合，濫用と見なして term から除外する前処理が実装されている (デフォールト値は 0.5). A = [0.10, 0.33, 0.5, 0.67, 0.90, 0.99] を試しているが，結果が煩雑にならないよう，この 3 値に限定した.

key	3	4	5	6	7	10	15
arabic-1g	0.84	0.74	0.95	1.00	0.88	0.87	0.90
arabic-2g	0.67	0.72	0.78	0.91	0.95	0.82	0.88
arabic-3g	0.67	0.77	0.73	0.87	0.68	0.71	0.71
arabic-sk2g	0.74	0.69	0.73	0.64	0.52	0.80	0.41
arabic-sk3g	0.73	0.69	0.76	0.69	0.62	0.60	0.49
english-1g	0.68	0.69	0.66	0.64	0.65	0.66	0.62
english-2g	0.56	0.64	0.57	0.63	0.64	0.68	0.60
english-3g	0.66	0.66	0.65	0.61	0.58	0.66	0.56
english-sk2g	0.55	0.52	0.59	0.53	0.60	0.69	0.38
english-sk3g	0.52	0.66	0.61	0.51	0.40	0.40	0.39
french-1g	0.69	0.67	0.59	0.66	0.65	0.64	0.63
french-2g	0.63	0.66	0.63	0.61	0.55	0.56	0.66
french-3g	0.60	0.61	0.64	0.68	0.66	0.60	0.70
french-sk2g	0.61	0.62	0.61	0.55	0.62	0.57	0.00
french-sk3g	0.48	0.54	0.52	0.48	0.33	0.00	0.49
german-1g	0.64	0.60	0.67	0.60	0.67	0.64	0.60
german-2g	0.69	0.68	0.62	0.62	0.60	0.56	0.55
german-3g	0.58	0.61	0.67	0.63	0.57	0.61	0.61
german-sk2g	0.50	0.60	0.54	0.58	0.62	0.59	0.33
german-sk3g	0.63	0.62	0.51	0.57	0.50	0.49	0.41
russian-1g	0.88	0.93	1.00	0.98	0.99	0.86	0.95
russian-2g	0.73	0.77	0.74	0.70	0.79	0.92	0.84
russian-3g	0.81	0.68	0.79	0.68	0.65	0.86	0.69
russian-sk2g	0.58	0.73	0.76	0.69	0.77	0.64	0.57
russian-sk3g	0.51	0.70	0.66	0.56	0.45	0.61	0.52

図 1: $s = 200, a = 0.90$ の場合の F(d, p) の表

濫用指数 $a = 0.67$ の場合の，$L \times T \times N$ の F 値を図 2 に示す．$a = 0.90$ の場合に比べて性能が落ちている．



key	3	4	5	6	7	10	15
arabic-1g	0.71	0.96	0.98	1.00	0.80	0.92	0.98
arabic-2g	0.88	0.80	0.84	0.69	0.83	0.69	0.81
arabic-3g	0.69	0.72	0.93	0.85	0.76	0.77	0.73
arabic-sk2g	0.72	0.67	0.73	0.67	0.70	0.62	0.70
arabic-sk3g	0.67	0.65	0.65	0.64	0.60	0.55	0.42
english-1g	0.69	0.71	0.66	0.76	0.69	0.68	0.65
english-2g	0.65	0.64	0.68	0.65	0.62	0.59	0.67
english-3g	0.66	0.59	0.66	0.67	0.59	0.53	0.61
english-sk2g	0.63	0.61	0.61	0.64	0.56	0.56	0.46
english-sk3g	0.53	0.63	0.58	0.52	0.46	0.50	0.32
french-1g	0.60	0.65	0.63	0.77	0.66	0.61	0.65
french-2g	0.66	0.63	0.73	0.57	0.62	0.60	0.64
french-3g	0.61	0.57	0.64	0.59	0.56	0.52	0.61
french-sk2g	0.57	0.55	0.61	0.61	0.55	0.51	0.39
french-sk3g	0.54	0.58	0.64	0.57	0.62	0.35	0.22
german-1g	0.50	0.66	0.65	0.64	0.68	0.65	0.61
german-2g	0.68	0.61	0.69	0.61	0.62	0.63	0.65
german-3g	0.68	0.53	0.67	0.62	0.65	0.60	0.47
german-sk2g	0.63	0.45	0.59	0.42	0.56	0.58	0.39
german-sk3g	0.70	0.54	0.39	0.51	0.32	0.62	0.55
russian-1g	0.87	0.89	1.00	0.99	0.93	0.91	0.96
russian-2g	0.80	0.65	0.77	0.71	0.88	0.81	0.89
russian-3g	0.74	0.61	0.69	0.80	0.78	0.79	0.63
russian-sk2g	0.51	0.52	0.67	0.60	0.76	0.51	0.58
russian-sk3g	0.60	0.63	0.55	0.63	0.46	0.48	0.51

図 2: $s = 200, a = 0.67$ の場合の F(d, p) の表

濫用指数 $a = 0.10$ の場合の，$L \times T \times N$ の F 値を図 3 に示す．$a = 0.67$ の場合に比べて更に性能が落ちている．

全体に性能が良い $a = 0.90, n = 10, t$ = 3-gram の条件の，t-SNE 配置 (D1, D2 平面) は図 4 の通り．大局的には，言語ごとの類似性と非類似性がそれなりにうまく捉えられているのがわかる．

全体に性能が良い $a = 0.90, n = 10, t$ = 3-gram の条件の階層クラスタリングの結果は図 5 の通り．ドイツ語の単語があちこちのクラスターに分散している．

3.2 考察 1

アラビア語とロシア語の結果と他の 3 言語の結果は大きく異なっている．これは，文字種が他の言語と異っているからだろう．同じ文字種を使う別の言語 (例えばペルシャ語やウクライナ語) を追加すれば，分類性能は下がるだろう．

他の 3 言語は，多かれ少なかれ文字集合を共有し，混同率が上がる．この意味では，英語とフランス語とドイツ語の分類性能の比較が本質的である．これらの言語では $a = [0.90, 0.67]$ で，0.65 ~ 0.7 の F 値が出ている．

key	3	4	5	6	7	10	15
arabic-1g	0.90	0.97	0.89	0.93	0.92	0.74	0.88
arabic-2g	0.61	0.56	0.60	0.64	0.58	0.68	0.81
arabic-3g	0.60	0.62	0.65	0.61	0.62	0.56	0.73
arabic-sk2g	0.68	0.63	0.66	0.66	0.68	0.35	0.70
arabic-sk3g	0.58	0.64	0.55	0.43	0.58	0.51	0.42
english-1g	0.67	0.63	0.61	0.63	0.61	0.64	0.63
english-2g	0.54	0.59	0.54	0.49	0.58	0.61	0.67
english-3g	0.62	0.60	0.65	0.51	0.50	0.08	0.61
english-sk2g	0.63	0.60	0.66	0.60	0.53	0.58	0.46
english-sk3g	0.45	0.51	0.46	0.39	0.49	0.51	0.32
french-1g	0.66	0.67	0.70	0.65	0.68	0.66	0.65
french-2g	0.62	0.56	0.61	0.60	0.62	0.55	0.64
french-3g	0.55	0.55	0.66	0.59	0.59	0.51	0.61
french-sk2g	0.60	0.49	0.53	0.46	0.61	0.64	0.39
french-sk3g	0.54	0.52	0.60	0.40	0.62	0.42	0.27
german-1g	0.64	0.72	0.68	0.70	0.65	0.68	0.61
german-2g	0.68	0.55	0.57	0.64	0.63	0.59	0.65
german-3g	0.61	0.57	0.60	0.56	0.61	0.47	0.47
german-sk2g	0.55	0.62	0.52	0.51	0.49	0.55	0.39
german-sk3g	0.49	0.49	0.42	0.53	0.36	0.67	0.53
russian-1g	0.97	1.00	0.95	0.94	0.87	0.76	0.90
russian-2g	0.94	0.81	0.80	0.81	0.92	0.64	0.89
russian-3g	0.63	0.78	0.70	0.63	0.62	0.70	0.63
russian-sk2g	0.72	0.65	0.66	0.59	0.62	0.74	0.58
russian-sk3g	0.61	0.40	0.52	0.57	0.53	0.38	0.53

図 3: $s = 200, a = 0.10$ の場合の F(d, p) の表

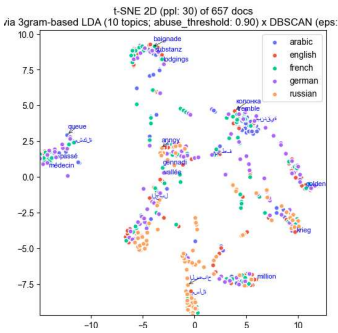


図 4: t-SNE 配置 (D1, D2) で言語毎に色分け

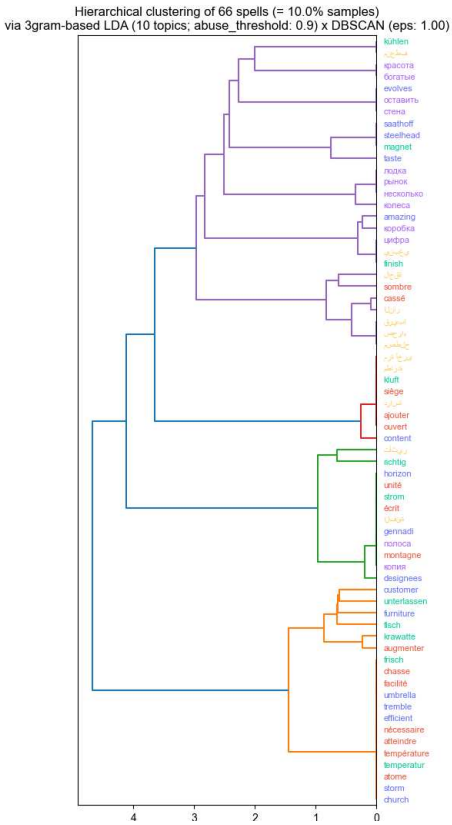


図 5: $s = 200, a = 0.90$, topic 数=10, 3-gram [言語の配色は図 4 と同じ]

図 1 の結果から判断する限り，topic 数がある程度大きい方 (6, 7, 10) が性能が良いが，大きくなり過ぎる (15) と性能が下がる．ただ，$a = 0.67$ と $a = 0.10$ の結果を見れば判るように，a が小さくなると逆の事が成立する．一般に a の値が小さいほど，性能が下がる．a の値が小さいと言う事は，それだけ多くの term が濫用に該当し，エンコードに利用されない事を意味する．a の値が小さくなると性能が下がるのは自然である[13]．語形クラスタリングが目的であるならば，term の濫用指数 a は大きいほど良い結果を与える．

LDA のトピック数は大き過ぎても小さ過ぎてもいけない．未確認な場合があるが，最適値は 8 前後だろう．

最適な topic 数が言語で異なる可能性が示唆される．これは頑健な傾向である．英語とロシア語については，topic 数が 10 の時に最良の F 値が得られているが，アラビア語では最適な topic 数がもう少し小さく，フランス語とドイツ語では更に最適な topic 数が小さい．具体的な最適値はサンプル規模に依存する一方で，それは言語毎に異なるらしい．

単純な 2-, 3-gram の性能が良く，skippy 3-gram は性能が悪い．これは a の値に拠らない頑健な傾向で，skippy n-gram の性能の高さを示唆する結果 [7] と突き合わせると，意外な結果だった[14]．理由として考えられるのは，初期サンプル数が 1 言語当たり 200 というのは訓練データの規模として小さ過ぎた事である．この可能性に気付いて初期サンプル数を 500 や 800 に増やしたが，結果は改善されなかった．

後に実施した様々に異なる設定[15]で得られた結果から奇妙な特性が確認された．どうやら，異なる文字種の単語がクラスターに混在するが LDA では避けられない (一部は図 5 でも確認できる)．異なる言語が同一クラスターに混在する事は不思議ではないが，異なる文字種の単語が同一クラスターに混在するのは不思議である．設定を色々変えてみたが，LDA の範囲ではこれを防ぐ方法は見つけられなかった．これが不可避なら，提案手法には自動分類アルゴリズムとしては致命的な弱点がある．

3.3 Labeled LDA の解析結果

この問題の解決を見込んで，Labeled LDA (L-LDA) [8] を利用した．L-LDA は (半) 教師あり条件でトピックを構築する．この実験では，語 (document に相当) の label としてその語の言語名を与えた．実装には TomotoPy[16] を使った．L-LDA では上の GenSim を使った実験と違って，a の違いは反映する結果は得られない．

表 6 の結果が示すように L-LDA により分類精度が劇的に改善された[17]．英語，フランス語，ドイツ語は文字

[13] a が大きい程性能が良くなる訳でもない．0.99 のように極端に大きくすると副作用が出る．詳細は紙面の都合で割愛する．

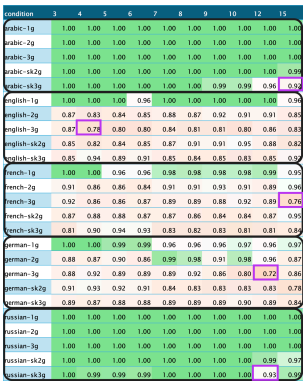
[14] 最大 skip 数を document の最大長 11 の 0.33 の 4 文字という控え目な値にしているのに，そうだった

[15] 異なるデータ量 × 異なる次元圧縮法 (t-SNE, UMAP, LLE, Isomap, PLS, ICA, PCA, MDS) を試した．

[16] `https://bab2min.github.io/tomotopy/`

[17] 交差検証で十分な精度の分類の実現が確認できたが，詳細は割愛．

種を共有するので，言語名を綴りから一意に確定するのは不可能である (それでもドイツ語は 2-gram で高性能)．なお，サンプル数を増やすと，性能は更に向上する．



condition	3	4	5	6	7	8	9	10	12	15
arabic-1g	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
arabic-2g	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
arabic-3g	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
arabic-sk2g	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99
arabic-sk3g	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99	0.96	0.92
english-1g	1.00	1.00	1.00	0.96	1.00	1.00	1.00	1.00	1.00	0.96
english-2g	0.87	0.83	0.84	0.85	0.88	0.87	0.92	0.91	0.91	0.85
english-3g	0.87	0.78	0.80	0.80	0.84	0.81	0.81	0.80	0.86	0.83
english-sk2g	0.85	0.82	0.84	0.85	0.87	0.91	0.91	0.95	0.88	0.82
english-sk3g	0.85	0.94	0.89	0.91	0.85	0.84	0.85	0.83	0.85	0.92
french-1g	1.00	1.00	0.96	0.96	0.98	0.98	0.98	0.98	0.99	0.95
french-2g	0.91	0.86	0.86	0.84	0.91	0.91	0.93	0.91	0.89	0.96
french-3g	0.92	0.86	0.86	0.87	0.89	0.89	0.88	0.92	0.89	0.76
french-sk2g	0.87	0.88	0.88	0.87	0.87	0.86	0.84	0.84	0.87	0.95
french-sk3g	0.81	0.90	0.94	0.93	0.83	0.82	0.83	0.81	0.81	0.84
german-1g	1.00	1.00	0.99	0.99	0.96	0.96	0.96	0.97	0.96	0.97
german-2g	0.88	0.87	0.90	0.86	0.99	0.98	0.91	0.98	0.96	0.87
german-3g	0.88	0.92	0.89	0.89	0.89	0.92	0.86	0.80	0.72	0.86
german-sk2g	0.91	0.93	0.92	0.91	0.84	0.83	0.83	0.83	0.83	0.78
german-sk3g	0.89	0.87	0.88	0.88	0.89	0.89	0.89	0.90	0.89	0.84
russian-1g	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
russian-2g	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
russian-3g	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
russian-sk2g	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.97
russian-sk3g	1.00	0.99	0.99	0.99	1.00	1.00	1.00	1.00	0.93	0.97

図 6: $s = 200, a = n.a.$ の場合の $F(d, p)$ の表 [言語毎の最低値を紫で囲った]

3.4 考察 2

教師ありモデルの L-LDA と教師なしモデルの LDA の結果の違いは歴然としている．L-LDA の性能は他の強力な機械学習の手法に見劣りしない．だが，ヒトの言語知識のモデルとして考えた場合は，どうなのか？ L-LDA と LDA の違いは，幾つかの異なる言語を知っていてそれらに属する語形を分類する状態と，それを知らないで分類する状態に対応するように思える．その意味では L-LDA は不自然な設定ではない．重要なのは，L-LDA の設定でも，英語，フランス語，ドイツ語の分類性能が 1.0 に及ばない事である．これは語形の自動分類が課題として原理的に難しい事を意味する．

DBSCAN は結果のクラスタリングで重要な役割を演じている．他のアルゴリズムを使った場合，異なる結果が得られるかも知れない．比較して確かめる必要がある．

T について言うと，skippy n-gram は n が大きいと有害だと示唆される．これは意外だった．

4 議論

単語の綴りにトピックがあるという想定は妥当なのか？ 妥当なら，綴りのトピックは何なのか？ なぜその最適値は 7〜10 辺りなのか？ などの疑問が自然に生じる．今の時点で最後の二つに答えを出す事はできないが，最初の問には答えがある．**トピックが意味的なものでなければならないなら，LDA の結果から単語の綴りを高精度に分類できる事が説明できない**．だから，単語の綴りにトピックがあるという想定は妥当である．

(L-)LDA は topic models [1, 2, 3, 6] であり，文書分類の手法として発展して来た．トピックが何らかの意味に対応しているという想定は，この背景から「仮定により真」なだけなのではないか？ アルゴリズムとして topic models がやっている事は (LSI であれ，pLSI であれ (L-)LDA であれ)，**全体となる集合 (例えば文書) とその構成部分 (例えば単語) との共起を説明する潜在因子**の抽出である．これらは意味よりも抽象的な部分/全体関係を表現する因子である．これらが概念的な意味に対応するのは，文書を bag-of-words としてエンコードして (L-)LDA 適用した時にのみ発生する偶発的副産物であると考えるべきである．また，この点は (L-)LDA に限定されず，topic models 一般に言えるはずである．

これが正しいなら，非言語データにも topic models が適応できると予想できる．たとえば，画像データに LDA が適用し，対象を抽出できる可能性がある．より言語に近い対象であれば，音譜を対象に LDA を適用し，主題のようなものを抽出できる可能性がある．これらは term の定義が自明ではないとは言え，将来的に確かめるに値する可能性である．

5 結論

本研究は，LDA が抽出するトピックは意味的なものである必要はないという再解釈の妥当性を確かめるため，LDA を使って幾つかの言語の語形を教師なしでクラスタリングし，その結果を自己教師型分類として評価すると，それなりの精度が得られる事を示した．L-LDA を使った分類は分類手法としても高性能であった．これらの結果は「LDA が抽出するトピックは意味的なものである必要はない」という再解釈の妥当性の証拠になる．

参考文献

[1] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet allocation. *J. of Machine Learning Research*, Vol. 3, pp. 993–1022, 2003.

[2] 岩田具治. トピックモデル. 講談社, 2015.

[3] 佐藤一誠. トピックモデルによる統計的潜在意味分析. コロナ社, 2015.

[4] Kow Kuroda. Finding structure in spelling and pronunciation using Latent Dirichlet Allocation. In *Proc. of the 30th Annual Meeting of the NLP Association*, 2024.

[5] 黒田航, 相良かおる, 東条佳奈, 麻子軒, 西嶋佑太郎, 山崎誠. LDA を使った専門用語の教師なしクラスタリング. 言語処理学会 30 回年次大会発表論文集, pp. 2858–63, 2024.

[6] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical Dirichlet processes. *J. of the American Statistical Association*, Vol. 101, No. 476, pp. 1566–1581, 2006.

[7] Kow Kuroda. In search of efficient, parsing-free encodings of word structure: Efficacy comparison among n-grams, skippy n-grams and extended skippy n-grams on noun classification tasks. In *Proc. of the 32th Annual Meeting of the NLP Association*, pp. 1504–09, 2025.

[8] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proc. of the 2009 Conference on Empirical Methods in NLP: Vol. 1*, Vol. 1, pp. 248–256, 2009.