

Unsupervised Text Classification

Iqbal Pahlevi Amin *
Faculty of Computer Science
Universitas Indonesia
Depok, Indonesia
Email: *iqbal.pahlevi@ui.ac.id

Abstract—Pada eksperimen ini, penulis melakukan klasifikasi dokumen tanpa label berdasarkan label yang telah didefinisikan di awal melalui pendekatan *unsupervised learning*. Pendekatan ini memerlukan label beserta deskripsi yang telah didefinisikan di awal serta dataset tanpa label yang akan digunakan. Metode yang digunakan bekerja dengan cara melakukan *embedding* vektor pada dokumen dan deskripsi masing-masing label. Kemudian, masing-masing vektor *embedding* dokumen akan dihitung *similarity*-nya dengan vektor *embedding* semua deskripsi label. Penulis mendapatkan F1-score terbaik sebesar 0.823 pada data *training* dan 0.817 pada data *testing*.

Index Terms—Unsupervised learning, vektor *embedding*

I. PENDAHULUAN

Paper ini membahas tentang cara melakukan klasifikasi teks atau dokumen tanpa label melalui pendekatan *unsupervised learning* dengan menggunakan Lbl2Vec. Klasifikasi teks atau dokumen umumnya dilakukan melalui pendekatan konvensional, yaitu *supervised learning*. Namun, pendekatan ini membutuhkan jumlah *labeled data* yang sangat besar untuk *training* model. Padahal pada kenyataannya, *labeled data* dalam bentuk dokumen itu sering tidak tersedia. Di sisi lain, apabila ingin melakukan labeling data secara manual, itu akan memakan banyak waktu dan membutuhkan *resource* yang banyak. Oleh karena itu, pendekatan *unsupervised learning* mulai digunakan karena lebih efisien dan murah. Klasifikasi teks secara *unsupervised* disebut juga sebagai *zero-shot text classification*.

pada eksperimen ini penulis mereproduksi eksperimen pada *paper*[1]. Tidak seperti pada *paper* acuan, pada eksperimen ini penulis melakukan eksperimen dengan hanya menggunakan dataset dari AG's corpus tanpa data dari 20News-groups. Penulis melakukan menjalankan tiga skenario dalam melakukan *training* model. Skenario ini berbeda dalam jumlah data *training* yang akan digunakan. Skenario pertama, menggunakan 10% data *training*. Skenario kedua menggunakan 25% data *training*. Dan skenario ketiga menggunakan 100% data *training*. Dari ketiga skenario tersebut didapatkan bahwa semakin banyak data *training* yang digunakan, model dapat menghasilkan nilai F1-score yang lebih bagus. Pada penggunaan 100% data *training*, penulis mendapatkan F1-score yang tidak jauh berbeda dari hasil eksperimen pada *paper* acuan[1]. Selain itu, penulis juga melakukan eksperimen dengan melakukan empat percobaan dengan inisiasi model menggunakan *hyperparameter* yang berbeda.

Selanjutnya pada bab alur kerja program akan dijelaskan lebih detail terkait alur program berjalan, dari proses tokenisasi

hingga proses evaluasi pada data *testing*. Kemudian, pada bab percobaan dan analisis hasil akan dijelaskan terkait setup eksperimen dan penjelasan terkait hasil yang didapatkan. Terakhir, pada bab kesimpulan dan saran akan dijelaskan kesimpulan yang didapat penulis setelah melakukan eksperimen ini serta beberapa saran yang dapat dilakukan pada penelitian selanjutnya.

II. CARA KERJA PROGRAM

Secara garis besar, program ini berjalan melalui empat tahap penting. Pertama, melakukan *data preprocessing*. Kedua, melakukan *embedding* vektor untuk label dan dokumen *training-testing*. Ketiga, mengategorikan masing-masing dokumen *training* ke suatu label berdasarkan hasil perhitungan *similarity*. Dan terakhir, mengategorikan masing-masing dokumen *testing* ke suatu label berdasarkan hasil perhitungan *similarity*.

```
# Subsetting and tag documents contained title + description for Lbl2Vec training
ag_full_corpus["tagged_docs"] = ag_full_corpus.apply(lambda row: TaggedDocument(tokenize(row["title"] + " " + row["description"]), [str(row.name)]), axis=1)
%ls
✓ 31
```

	class	title	description	data set type	tagged docs
0	3	Wall St. Bears Claw Back Into the Black (Reuters)	Reuters - Short-sellers, Wall Street's dwell...	train	[wall st bears claw back into the bla...
1	3	Carlyle Looks Toward Commercial Aerospace (Reuters)	Reuters - Private investment firm Carlyle Grou...	train	[carlyle looks toward commercial aersp...
2	3	Oil and Economy Could Shake' Oilfield (Reuters)	Reuters - Slumping crude prices plus worries ab...	train	[oil and economy could shake oilfield c...
3	3	Iraq Halts Oil Exports from Main Southern Pipe	Reuters - Authorities have halted oil export...	train	[iraq halts oil exports from main south...
4	3	Oil prices soar to all-time record, posting new...	AFP - Turbulent world of prices, trading rest...	train	[oil prices soar to all time record po...
1271	4	AGU Properties Sign Grants for Thailand State...	AGU Properties Sign Grants for Thailand State...	test	[agu properties sign grants for thailand...
1272	4	Casino Spins Into Little Saturn Moon (AP)	AP - NASA's Cassini spacecraft has spun two ti...	test	[casino spins into little saturn moon...
1273	1	On front line of AIDS in Russia	An industrial city northeast of Moscow struggl...	test	[on front line of aids in russia see l...
1274	4	Nobel Laureate Decides Stem Cell Limits (AP)	AP - A Nobel laureate in medicine said Monday...	test	[nobel laureate decides stem cell limits...
1275	2	Iran Can Hear of Kobe Accuser's Sex Life (AP)	AP - Prosecutors suffered another setback Mond...	test	[iran can hear of kobe accuser sex li...

1276 rows x 6 columns

Fig. 1. Dokumen *train* dan *test* beserta hasil tokenisasinya.

Gambar 1 menampilkan cuplikan data *training* dan *testing* beserta hasil tokenisasinya. Proses tokenisasi data memanfaatkan fungsi `simple_preprocess` yang berasal dari *library* `gensim`. Hasil tokenisasi kedua jenis data disimpan dalam bentuk *list of words*.

```
# Create words
lbl2vec_words = FreqList
%ls
✓ 16
```

	class	title	description	data set type	tagged docs
0	3	Wall St. Bears Claw Back Into the Black (Reuters)	Reuters - Short-sellers, Wall Street's dwell...	train	[wall st bears claw back into the bla...
1	3	Carlyle Looks Toward Commercial Aerospace (Reuters)	Reuters - Private investment firm Carlyle Grou...	train	[carlyle looks toward commercial aersp...
2	3	Oil and Economy Could Shake' Oilfield (Reuters)	Reuters - Slumping crude prices plus worries ab...	train	[oil and economy could shake oilfield c...
3	3	Iraq Halts Oil Exports from Main Southern Pipe	Reuters - Authorities have halted oil export...	train	[iraq halts oil exports from main south...
4	3	Oil prices soar to all-time record, posting new...	AFP - Turbulent world of prices, trading rest...	train	[oil prices soar to all time record po...
1271	4	AGU Properties Sign Grants for Thailand State...	AGU Properties Sign Grants for Thailand State...	test	[agu properties sign grants for thailand...
1272	4	Casino Spins Into Little Saturn Moon (AP)	AP - NASA's Cassini spacecraft has spun two ti...	test	[casino spins into little saturn moon...
1273	1	On front line of AIDS in Russia	An industrial city northeast of Moscow struggl...	test	[on front line of aids in russia see l...
1274	4	Nobel Laureate Decides Stem Cell Limits (AP)	AP - A Nobel laureate in medicine said Monday...	test	[nobel laureate decides stem cell limits...
1275	2	Iran Can Hear of Kobe Accuser's Sex Life (AP)	AP - Prosecutors suffered another setback Mond...	test	[iran can hear of kobe accuser sex li...

1276 rows x 6 columns

Fig. 2. Label beserta *embedding* vektor dan dokumen terkait dari hasil *training* model Lbl2Vec.

Pemanggilan *function* `.fit()` pada gambar 2 digunakan untuk melakukan *training* model yang akan digunakan dalam

eksperimen ini. Pada proses *training*, terjadi beberapa hal penting sebagai berikut.

- 1) Melakukan embedding vektor untuk label berdasarkan keyword deskripsi masing-masing label.
- 2) Melakukan embedding vektor untuk data *training*.
- 3) Menghitung *similarity* dari hasil vektor embedding setiap dokumen *training* dengan hasil vektor embedding label.
- 4) Mengategorikan setiap dokumen ke dalam label dengan *similarity* tertinggi.

Gambar 2 menampilkan dua hal, pertama nama label beserta keyword deskripsi dan vektor embedding label dan kedua nama label beserta index data yang termasuk ke dikategorikan sebagai label tersebut.

```
# predict similarity scores (cosine similarities, range: [-1,1])
model_docs_lbl_similarities = lbl2vec_model.predict_model_docs()
```

```
2023-11-24 14:23:13,796 - lbl2vec - INFO - Get document embeddings from model
2023-11-24 14:23:13,798 - lbl2vec - INFO - Calculate document<->label similarities
```

PERHITUNGAN SIMILARITAS DOKUMEN DENGAN LABEL

doc_key	most_similar_label	highest_similarity_score	World	\
0	0	World	0.000814	0.000814
1	1	Sports	0.043607	0.037535
2	2	Business	0.167956	0.044195
3	3	Business	0.185562	0.064150
4	4	World	0.010808	0.010808
...
1195	1195	Business	0.049616	-0.075422
1196	1196	World	-0.026890	-0.026890
1197	1197	Business	0.035581	0.009555
1198	1198	Sports	0.146370	0.052269
1199	1199	Science/Technology	0.162072	0.043237

	Sports	Business	Science/Technology
0	-0.046638	-0.067967	-0.003685
1	0.043607	0.032575	0.016319
2	-0.033512	0.167956	-0.051153
3	0.140987	0.185562	0.044856
4	-0.021755	-0.001197	0.003367
...
1195	-0.013300	0.049616	0.048875
1196	-0.080881	-0.094129	-0.105980
1197	0.011518	0.035581	0.015231
1198	0.146370	0.035285	0.066323
1199	0.143853	0.078076	0.162072

Fig. 3. Hasil perhitungan *similarity* antara dokumen *training* dengan label.

Gambar 3 menampilkan hasil pengkategorian label untuk setiap dokumen *training* beserta skor *similarity* dengan semua label yang ada. Hasil ini didapatkan dari proses perhitungan *cosine similarity* dari vektor embedding dokumen dengan vektor embedding label.

```
# predict similarity scores of new test documents (they were not used during training)
new_docs_lbl_similarities = lbl2vec_model.predict_new_docs(tagged_docs=ag_full_corpus['tagged_docs'],[ag_full_corpus['data_set_type']=='test'])
```

```
2023-11-24 14:01:30,206 - lbl2vec - INFO - Calculate document embeddings
2023-11-24 14:01:30,221 - lbl2vec - INFO - Calculate document<->label similarities
```

PERHITUNGAN SIMILARITAS DOKUMEN TEST DENGAN LABEL

doc_key	most_similar_label	highest_similarity_score	World	Sports	\
0	1200	Sports	0.021133	0.011549	0.021133
1	1201	Business	0.094095	0.056599	0.014640
2	1202	Sports	0.094550	0.062710	0.004928
3	1203	Science/Technology	0.131308	0.065518	0.112505
4	1204	World	0.091723	0.091723	0.013812
...
71	1271	Business	0.148031	0.025283	0.052244
72	1272	Science/Technology	0.012080	0.005796	-0.004461
73	1273	World	0.070052	0.070052	-0.131862
74	1274	Sports	0.010224	0.005275	0.010224
75	1275	Business	0.070051	-0.000636	-0.097134

	Business	Science/Technology
0	0.012156	0.002344
1	0.094095	-0.004021
2	0.084572	-0.042385
3	0.129105	0.131308
4	-0.017071	-0.000908
...
71	0.148031	0.078802
72	-0.004340	0.012080
73	-0.014663	0.000134
74	-0.056178	0.010303
75	0.070051	-0.000752

[76 rows x 7 columns]

Fig. 4. Hasil perhitungan *similarity* antara dokumen *testing* dengan label.

Gambar 4 menampilkan hasil pengkategorian label untuk setiap dokumen *testing* beserta skor *similarity* dengan semua label yang ada. Hasil ini didapatkan melalui proses yang sama dengan proses untuk mendapatkan hasil pada gambar 3, yaitu dengan menggunakan *cosine similarity*.

III. HASIL UJI COBA DAN ANALISIS

A. Metrik Evaluasi

Metrik evaluasi yang digunakan pada eksperimen ini adalah F1-score. F1-score dihitung dengan cara mempertimbangkan precision dan recall secara imbang[2].

$$F1score = \frac{2 \times precision \times recall}{precision + recall} \quad (1)$$

B. Dataset

Dataset yang digunakan dalam eksperimen ini adalah dataset **AG's Corpus**¹ dengan detail seperti pada pada tabel I. Dataset tersebut menggunakan bahasa Inggris. Pada eksperimen ini, dataset disimpan dalam bentuk csv. *File train.csv* yang berisi corpus untuk *training* dan *test.csv* yang berisi corpus untuk *testing*.

TABLE I
DETAIL DATASET[1]

Dataset	#Training documents	#Test documents	#Classes
AG's Corpus	120000	7600	4

C. Experiment Settings

Eksperimen ini tidak membutuhkan spesifikasi *hardware* atau *software* khusus dalam menjalankan program. Saya menggunakan beberapa model dengan *hyperparameters* yang berbeda dalam melakukan eksperimen ini yang dijelaskan pada tabel II.

TABLE II
Experiments settings

Model	Hyperparameters
A	<i>similarity_threshold</i> = 0.30 <i>min_num_docs</i> = 100 <i>epochs</i> = 10
B	<i>similarity_threshold</i> = 0.50 <i>min_num_docs</i> = 100 <i>epochs</i> = 10
C	<i>similarity_threshold</i> = 0.30 <i>min_num_docs</i> = 200 <i>epochs</i> = 10
D	<i>similarity_threshold</i> = 0.30 <i>min_num_docs</i> = 100 <i>epochs</i> = 15

¹https://github.com/mhjabreel/CharCnn_Keras/tree/master/data/ag_news_csv

D. Hasil

Hasil skor eksperimen untuk masing-masing model dapat dilihat pada tabel III. Model A (*baseline model*) memberikan hasil F1-score yang cukup baik, yaitu 0.8219 saat *training* dan 0.8151 saat *testing*. Kemudian, peningkatan nilai *similarity_threshold* pada model B berdampak pada turunnya F1-score sebesar 0.0104 pada *training* dan 0.0116 pada *testing*. Sedangkan, peningkatan nilai *min_num_docs* pada model C memberikan hasil F1-score yang paling tinggi, baik pada saat *training* maupun *testing* dengan skor sebesar 0.8233 dan 0.8171. Terakhir, peningkatan jumlah *epochs* pada model D berdampak pada penurunan F1-score *training* sebesar 0.002, namun peningkatan F1-score *testing* sebesar 0.0012.

TABLE III
HASIL EKSPERIMEN

Model	F1-score	
	Training	Testing
A	0.8219	0.8151
B	0.8115	0.8035
C	0.8233	0.8171
D	0.8199	0.8163

E. Diskusi

Pada skenario model B, peningkatan *similarity_threshold* menyebabkan penurunan F1-score untuk kedua fase *training* dan *testing*. *Hyperparameter* ini digunakan sebagai *threshold*/batas minimum untuk mengategorikan suatu dokumen menjadi suatu label. Semakin tinggi nilai *hyperparameter* ini, maka semakin sulit untuk mengategorikan suatu dokumen. Oleh karena itu terjadi penurunan F1-score pada tahap *training* dan *testing*.

Pada skenario model C, peningkatan *min_num_docs* memberikan hasil F1-score tertinggi untuk kedua fase *training* dan *testing*. *Hyperparameter* ini merupakan jumlah minimal dokumen yang akan dipilih untuk menghitung *label embeddings*. Semakin tinggi nilai *hyperparameter* ini, maka model akan menghasilkan dokumen dengan informasi yang lebih signifikan dan kontekstual. Oleh karena itu, model akan lebih mudah dalam mengategorikan suatu dokumen dan menghasilkan F1-score yang lebih tinggi pada *training* dan *testing*.

Pada skenario model D, peningkatan jumlah *epochs* menyebabkan penurunan F1-score pada fase *training*, namun peningkatan pada fase *testing*. *Epochs* adalah *hyperparameter* yang digunakan untuk melatih seberapa lama model akan dilatih. Semakin tinggi nilai *epochs*, maka model akan semakin lama dilatih. Oleh karena itu, dengan meningkatnya jumlah *epochs* model akan lebih memahami pola dataset yang diberikan dan akan memberikan hasil evaluasi *testing* yang lebih bagus. Namun, perlu diwaspadai bahwa semakin tinggi nilai *epochs*, model akan semakin rawan menjadi *overfitting*.

Kelemahan model Lbl2Vec adalah model tidak dapat belajar untuk menemukan *hyperparameter* sendiri yang bagus seperti pada *neural networks*. Melainkan, peneliti harus

menentukan *hyperparameter* sendiri yang didefinisikan di awal. Agar mendapatkan model dengan performa yang bagus, peneliti harus mencoba-coba berbagai kombinasi *hyperparameter* secara manual. Selain itu, kelemahan lain dari model ini adalah jika pada label terdapat kata/frasa yang belum pernah dipelajari/didapat dari dokumen *training*, maka kata/frasa tersebut akan di-*exclude* dan menyebabkan turunnya hasil F1-score.

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Pada eksperimen ini, penulis melakukan klasifikasi dokumen melalui pendekatan *unsupervised learning* dengan memanfaatkan model Lbl2Vec. Model ini bekerja berdasarkan vektor *embedding* dari dokumen sebagai *unlabeled data* dan label beserta deskripsi yang telah didefinisikan di awal. Pada eksperimen ini model C memberikan hasil F1-score terbaik, 0.8233 pada *training* serta 0.8171 pada *testing*. Model C menggunakan *hyperparameter* *similarity_threshold*=0.30, *min_num_docs*=200, dan *epochs*=10. Hal ini menunjukkan bahwa model Lbl2Vec memberikan hasil F1-score yang cukup baik dalam melakukan klasifikasi teks secara *unsupervised*.

B. Saran

Eksperimen yang telah dilakukan oleh penulis masih sangat terbatas. Eksperimen ini hanya menggunakan 120.000 data *training* dan 7600 data *testing* yang berbahasa inggris. Oleh karena itu, perlu dilakukan eksperimen lebih lanjut pada dokumen berbahasa selain inggris. Hal ini dilakukan untuk mengetahui performa model Lbl2Vec pada dokumen yang berbahasa selain bahasa inggris. Selain itu, penulis menyarankan untuk melakukan eksperimen pada dokumen multibahasa.

ACKNOWLEDGMENT

Eksperimen ini didukung oleh Fakultas Ilmu Komputer, Universitas Indonesia. Penulis mengucapkan terima kasih banyak kepada Bu Ika Alfina selaku pengampu mata kuliah Pengolahan Bahasa Manusia pada Semester Gasal 2023/2024. Serta kepada Luthfi Balaka selaku asisten dosen yang membantu dalam proses penyelesaian eksperimen ini.

REFERENCES

- [1] T. Schopf, D. Braun, and F. Matthes, "Lb12vec: An embedding-based approach for unsupervised document retrieval on predefined topics," in *Proceedings of the 17th International Conference on Web Information Systems and Technologies*. SCITEPRESS - Science and Technology Publications, 2021. [Online]. Available: <http://dx.doi.org/10.5220/0010710300003058>
- [2] M. Sokolova, N. Japkowicz, and S. Szpakowicz, "Beyond accuracy, f-score and roc: A family of discriminant measures for performance evaluation," vol. Vol. 4304, 01 2006, pp. 1015–1021.