

---

## PENINGKATAN AKURASI TOPIC MODELING PADA KINERJA KEPOLISIAN REPUBLIK INDONESIA DI TWITTER MENGGUNAKAN ALGORITMA SPELL CHECKER

**Isra Andika Bakhri**

Magister Teknik Informatika  
Universitas Amikom Yogyakarta

Email: [israandikabakhri@students.amikom.ac.id](mailto:israandikabakhri@students.amikom.ac.id)

**Rifqi Mulyawan**

Magister Teknik Informatika  
Universitas Amikom Yogyakarta

Email: [rifqi.mulyawan@students.amikom.ac.id](mailto:rifqi.mulyawan@students.amikom.ac.id)

**Nurfajri Asfa**

Magister Teknik Informatika  
Universitas Amikom Yogyakarta

Email: [nurfajri.asfa@students.amikom.ac.id](mailto:nurfajri.asfa@students.amikom.ac.id)

**Emma Utami**

Magister Teknik Informatika  
Universitas Amikom Yogyakarta

Email: [ema.u@amikom.ac.id](mailto:ema.u@amikom.ac.id)

**Agus Fatkhurohman**

Magister Teknik Informatika  
Universitas Amikom Yogyakarta

Email: [agusfatkhurohman@amikom.ac.id](mailto:agusfatkhurohman@amikom.ac.id)

### ABSTRAK

Kinerja polisi republik Indonesia saat ini menjadi sorotan terutama munculnya banyak komentar dari netizen terutama di Twitter dengan diramaikannya *hashtag* #percumalaporpolisi dan #1hari1oknum. Hal ini mesti direspon cepat oleh kepolisian republik indonesia untuk melakukan *counter issue* dengan pertamanya menggali topik tersembunyi dari sekian banyak *tweet* yang beredar salah satu caranya yaitu menerapkan *topic modeling* pada Twitter. Dalam rangka memetakan *issue* atau topik yang akan dibenahi menurut padangan publik terlebih dahulu. Salah satu kelemahan penerapan *topic modeling* ialah adanya *typo* maupun singkatan disengaja yang dapat mengganggu keakuratan model yang akan dibangun. Maka penting diterapkannya peningkatan perbaikan teks *tweet* yang diakuisisi sebelum dimasukkan ke dalam model. Normalisasi teks dapat dikalaborasi dengan algoritma *spell checker* yang dapat membantu menormalkan *typo* dan singkatan yang disengaja untuk membantu meningkatkan akurasi model. Penerapan algoritma *spell checker* terbukti mampu meningkatkan akurasi model yang mengakibatkan topik yang dihasilkan sebelum dan sesudah diterapkan *spell checker* berbeda, ditandai dengan tidak ditemukannya lagi singkatan atau *typo* pada *term* yang diolah oleh LDA.

**Kata kunci:** *topic modeling*, LDA, *spell checker*, teorema bayesian, normalisasi teks, *text mining*

## ABSTRACT

*Police performance in the Indonesian Republic is currently in the spotlight, especially the emergence of many comments from netizens, especially on Twitter with the hashtags #percumalaporpolisi and #IhariLoknum. The Republic of Indonesia's police must respond swiftly to this subject by conducting a counter-issue by first analyzing the hidden topics of the numerous circulating tweets. One method is using topic modeling on Twitter and mapping the issue or topic to be raised according to the public's view first. One of the drawbacks of applying topic modeling is that some intentional typos or abbreviations can interfere with the model's accuracy. So it is important to implement improvements to the text of the tweets obtained before entering into the model. Text normalization can collaborate with spell-checking algorithms to help normalize intentional typos and abbreviations to help improve model accuracy. The use of a spell-checking algorithm is proved to improve the model's accuracy to generate distinct subjects before and after the spell checker is used, as evidenced by the absence of abbreviations or typos in words processed by LDA.*

**Keywords:** *topic modeling, LDA, spell checker, teorema bayesian, normalisasi teks, text mining*

## 1. PENDAHULUAN

Ramainya pemberitaan tentang kinerja kepolisian republik Indonesia saat ini menjadi hangat karena persoalan beberapa masalah yang sempat viral yang akhirnya banyak dari netizen Indonesia meramaikannya di sosial media dengan cara meramaikan *hashtag* #percumalaporpolisi dan #IhariLoknum. Hal ini pun dijadikan bahan evaluasi oleh kapolri dalam melakukan perbaikan secara menyeluruh. Namun, dalam perkembangan ilmu pengetahuan di bidang teknologi terkhusus bidang *Natural Language Processing* (NLP) telah menawarkan solusi untuk membuka secara luas pola komentar netizen di sosial media untuk menjadi kajian lebih mendalam bagi polisi republik Indonesia yaitu menggunakan *topic modeling* dengan memanfaatkan kumpulan data besar komentar netizen di sosial media seperti Twitter pada topik tertentu [1][2].

Beberapa peneliti di dunia menggunakan *topic modeling* untuk beberapa kepentingan seperti untuk ilmu medis [3][4], ilmu politik [5][6], geografi [7][8] dan *software engineer* [9][10]. Di negara Indonesia sendiri, ada beberapa peneliti yang juga fokus pada *topic modeling* seperti menemukan topik tersembunyi pada penerapan inovasi di Indonesia selama masa pandemi [11], analisis *customer* dalam implementasi M-Payment dalam industri e-commerce [12] dan ada pula penelitian terkait penggalian informasi pada kumpulan hasil penelitian di bidang kesehatan pada jurnal SINTA untuk membantu peneliti menentukan arah penelitian selanjutnya di bidang kesehatan [13].

Salah satu metode terbaik yang digunakan oleh peneliti untuk *topic modeling* saat ini adalah metode *Latent Dirichlet Allocation* (LDA) pada penerapannya untuk pencarian topik tersembunyi di sosial media [14], tidak seperti model lain yaitu *Latent Semantic Analysis* (LSA) serta *Probabilistic Latent Semantic Analysis* (PLSA), mereka lebih cocok pada penerapan pengelompokan komunitas atau sentimen analisis (*sentiment analysis*) di media sosial terkhusus pada platform Twitter.

Seperti kita ketahui Twitter merupakan tempat netizen dalam mengekspresikan pemikiran dan pendapatnya pribadi, maka teks kadang menghasilkan sebuah kesalahan yang mempengaruhi sistem komputasi untuk menganalisis data tweet baik itu dikarenakan *typo* atau singkatan yang disengaja [1]. Oleh karena itu, dalam tahap *pre-processing* data harusnya bebas dari kesalahan dalam menginterpretasikan sebuah kata dengan tujuan peningkatan akurasi model yang akan diterapkan, tentunya *pre-processing* berupa normalisasi teks terbukti berhasil meningkatkan akurasi model [15].

Pada makalah ini penulis akan menerapkan *topic modeling* untuk mengetahui topik tersembunyi pada *tweet* netizen di Twitter tentang kinerja polisi republik Indonesia, serta dilakukannya peningkatan akurasi model dengan penambahan algoritma *spell checker*.

## 2. METODOLOGI PENELITIAN

### 2.1 *Spell Checker* dengan Teorema Bayesian

Penerapan *spell checker* adalah untuk mengembalikan kata kedalam bentuk aslinya. Pada umumnya ditemui kesalahan kata pada data hasil akuisisi dari media sosial dan jarang ditemukan pada dokumen akuisisi dari tempat lain seperti portal berita dan lainnya [14]. Pada makalah ini, pendekatan yang digunakan adalah menggunakan rumus teorema Bayesian untuk menghitung seberapa besar kemiripan antara kata yang dimasukkan dengan kata yang terdapat dalam basis data, yang rumusnya dapat dilihat di bawah.

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)} \quad (1)$$

Catatan:

$P(A|B)$  : Probability hipotesis A terjadi jika evidence B terjadi

$P(B|A)$  : Probability munculnya evidence B, jika hipotesis A terjadi

$P(A)$  : Probability hipotesis A tanpa melihat evidence apapun

$P(B)$  : Probability evidence B tanpa melihat apapun

Teorema Bayes pada pembelajaran mesin terbukti sangat baik diterapkan berdasarkan data *training* disebabkan pada pola perhitungannya menggunakan *conditional probability* sebagai dasarnya. Metode ini juga merupakan sebuah metode yang dapat memberikan hasil berupa sebuah parameter *estimation* dengan cara menggabungkan data dari *sample* dan data lain yang telah ada sebelumnya. Adapun keunggulan lain dari metode ini yaitu dapat menyederhanakan perhitungan *classic* yang dipenuhi integral sehingga dari hasil tersebut mendapatkan model *marginal* [16].

### 2.2 *Latent Dirichlet Analysis (LDA)*

*Latent Dirichlet Analysis (LDA)* adalah model yang digunakan dalam menemukan topik laten (tersembunyi) dibalik dokumen-dokumen. Hal tersebut terjadi karena model LDA menginterpretasikan dokumen sebagai campuran yang *random* (acak) [17]. Dalam konteks pembuatan model teks, model ini akan menghasilkan sebuah probabilitas topik secara eksplisit dari sebuah dokumen. Kemampuan model ini disebabkan model LDA menggunakan pendekatan model Bayesian hirarki 3 tingkat yang di mana tiap *item* yang dikoleksi kemudian dimodelkan sebagai campuran terbatas atas banyaknya rangkaian set topik.

Model LDA melakukan interpretasi topik-topik dalam dokumen dalam model statistik. Proses LDA bersifat *imaginary random* proses pada model bahwa tiap dokumen berasal dari tema atau topik tertentu dan setiap topiknya memiliki struktur distribusi kata-kata atau *term*.

Namun perlu diperhatikan bahwa model LDA adalah termasuk dalam model *unsupervised learning*. Sebelum proses *topic modeling* dilaksanakan, maka perlu penetapan jumlah topik yang akan dihasilkan biasanya penentuan jumlah topik dipilih dari angka 30 sampai 300 topik untuk menentukan topik yang baik [18].

Asumsi *Bag Of Words (BOW)* digunakan oleh model LDA dengan cara mengurutkan kemunculan kata dalam dokumen diabaikan. Asumsi semacam ini sebenarnya tidak realistis namun cukup beralasan karena tujuan LDA hanya menggunakannya untuk menemukan struktur semantik teks [18].

### 2.3 *Dataset*

*Dataset* didapatkan dari hasil *scraping* di Twitter menggunakan library *SNscrapers* yang di mana tanggal pengambilan mulai dari tanggal 5 desember 2021 sampai dengan 28 desember 2021. Dipilihnya tanggal tersebut karena pada rentang tanggal itu sedang viralnya *hashtag*

#percumalaporpolisi dan #1hari1oknum di Twitter. Penulis juga mengambil daya dengan *query* pencarian “kinerja polri” dan “kinerja polisi” dengan jumlah *tweet* sebanyak 9326 data *tweet*.

## 2.4 Pre-processing Data

Data *tweet* akan dilakukan tahap *pre-processing* dimana terdiri dari *case folding* untuk mengubah huruf menjadi huruf kecil lalu masuk ke tahap *lexical analysis* untuk menghapus tanda baca, spasi kosong, link, tanda *retweet*, nama akun dan proses tokenisasi. Setelah proses tersebut maka akan dilakukan preses *stopwords* untuk menghapus kata yang tidak deskriptif dan berulang, lalu itu akan dilaksanakan proses *stemming* untuk menghapus imbuhan pada kalimat.

Setelah proses penghapusan imbuhan maka akan dilaksanakan proses normalisasi teks menggunakan algoritma *spell checker* dengan pendekatan perhitungan teorema Bayesian menggunakan kumpulan kata baku atau kata yang sering digunakan sebagai bahasa gaul di Indonesia. Setelah dinormalkan maka dilaksanakan kembali proses *stemmer* guna memastikan hasil normalisasi tadi tidak ada lagi kata berimbuhan.

## 2.5 Evaluation

Evaluasi model digunakan cara melihat topik yang dihasilkan sebelum dan sesudah normalisasi teks menggunakan algoritma *spell checker*, nantinya akan dilihat berdasarkan term yang muncul apakah term yang muncul tersebut dapat dikatakan wajar atau terjadi bias pada model yang mengakibatkan berkurangnya akurasi model.

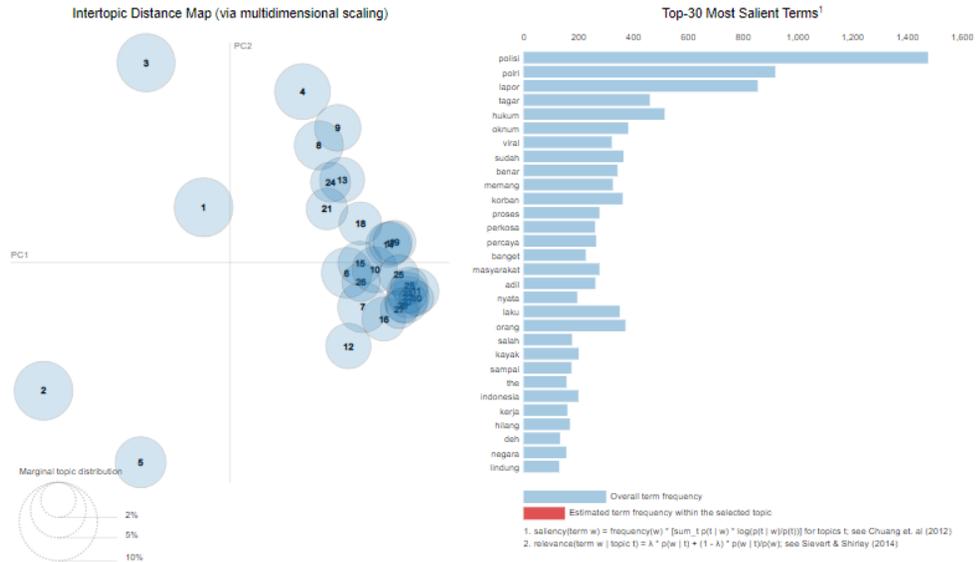
## 3. HASIL DAN PEMBAHASAN

Proses komputasi dilakukan menggunakan google colab dengan spesifikasi sebagai berikut ini: untuk GPU menggunakan GPU Tesla P100-PCIE-16GB sedangkan processor yang digunakan adalah Intel(R) Xeon(R) CPU @ 2,30GHz.

Dari hasil proses topic modeling didapatkan 30 topik karena dalam eksperimen ini ditetapkan pembentukan topik sebanyak 30 topik dengan 10 kata per topik. Namun penulis mengambil 5 topik teratas untuk melihat hasil yang diinginkan. Hasil *topic* (topik) dapat dilihat dalam gambar yang ada bawah ini.

```
Topik 0 0.107*"uang" + 0.061*"masuk" + 0.042*"bagus" + 0.033*"ku" + 0.030*"alas" +  
0.022*"ribu" + 0.021*"juta" + 0.021*"ketemu" + 0.020*"mesti" + 0.017*"jangan"  
  
Topik 1 0.222*"lapor" + 0.144*"polisi" + 0.062*"hilang" + 0.035*"mana" + 0.032*"suruh" +  
0.030*"gerak" + 0.028*"motor" + 0.025*"mending" + 0.020*"terus" + 0.018*"layanan"  
  
Topik 2 0.214*"viral" + 0.095*"kerja" + 0.060*"usut" + 0.054*"cepat" + 0.041*"media" +  
0.036*"tanggap" + 0.030*"tangan" + 0.027*"kapolri" + 0.023*"hormat" + 0.022*"miris"  
  
Topik 3 0.098*"adil" + 0.067*"moga" + 0.047*"cari" + 0.032*"allah" + 0.030*"jalan" +  
0.018*"orang" + 0.017*"laku" + 0.016*"adu" + 0.016*"buat" + 0.016*"selesai"  
  
Topik 4 0.063*"netizen" + 0.054*"tangkap" + 0.041*"berita" + 0.040*"hidup" + 0.037*"mati" +  
0.033*"laku" + 0.032*"teman" + 0.031*"beda" + 0.031*"lo" + 0.031*"dapat"
```

**Gambar 1. Hasil 5 Topik Teratas Yang Didapatkan Setelah Melakukan *Spell Checker***



Gambar 2. Hasil Visualisasi *Latent Dirichlet Allocation (LDA)*

### 3.1 Pembahasan Topik 0

Tabel 1. Hasil pada topik 0 tanpa *spell checker* dan menggunakan *spell checker*

<i>TOSP</i>	<i>POSP</i>	<i>TWSP</i>	<i>PWSP</i>
gerak	0.070	uang	0.107
gimana	0.070	masuk	0.061
trending	0.067	bagus	0.042
polisi	0.050	ku	0.033
sangka	0.045	alas	0.030
maaf	0.043	ribu	0.022
diam	0.034	juta	0.021
kali	0.029	ketemu	0.021
langsung	0.027	mesti	0.020
korban	0.025	jangan	0.017

Catatan:

*TOSP* : Term (kata) Yang Didapatkan Tanpa Algoritma Spell Checker

*POSP* : Nilai Peluang Term (kata) Yang Didapatkan Tanpa Algoritma Spell Checker

*TWSP* : Term (kata) Yang Didapatkan Dengan Algoritma Spell Checker

*PWSP* : Nilai Peluang Term (kata) Yang Didapatkan Dengan Algoritma Spell Checker

Kesimpulan pada Topik 0 ini adalah sebelum *spell checker* diimplementasikan *term* (kata) yang dihasilkan tidak begitu menggambarkan topik yang jelas. Namun setelah algoritma *spell checker* diimplementasikan maka nampak suatu topik yang membahas tentang persoalan masuk polri dengan menggunakan uang atau cara lain, penulis menduga ada semacam sogokan.

### 3.2 Pembahasan Topik 1

**Tabel 2. Hasil pada topik 1 tanpa spell checker dan menggunakan *spell checker***

<i>TOSP</i>	<i>POSP</i>	<i>TWSP</i>	<i>PWSP</i>
aparat	0.080	lapor	0.222
propam	0.059	polisi	0.144
leceh	0.055	hilang	0.062
coba	0.049	mana	0.035
heran	0.044	suruh	0.032
wkwk	0.031	gerak	0.030
foto	0.021	motor	0.028
susah	0.020	mending	0.025
kek	0.020	terus	0.020
alami	0.018	layanan	0.018

Catatan:

*TOSP* : Term (kata) Yang Didapatkan Tanpa Algoritma Spell Checker

*POSP* : Nilai Peluang Term (kata) Yang Didapatkan Tanpa Algoritma Spell Checker

*TWSP* : Term (kata) Yang Didapatkan Dengan Algoritma Spell Checker

*PWSP* : Nilai Peluang Term (kata) Yang Didapatkan Dengan Algoritma Spell Checker

Kesimpulan pada Topik 1 ini adalah sebelum *spell checker* diimplementasikan *term* (kata) yang dihasilkan tidak begitu menggambarkan topik yang jelas. Namun setelah algoritma *spell checker* diimplementasikan maka nampak suatu topik yang membahas tentang persoalan pencurian motor yang sering terjadi dan laporannya masuk di pelayanan polisi dan sementara polisi dituntut cepat dalam melaksanakan tugasnya.

### 3.3 Pembahasan Topik 2

**Tabel 3. Hasil pada topik 2 tanpa spell checker dan menggunakan *spell checker***

<i>TOSP</i>	<i>POSP</i>	<i>TWSP</i>	<i>PWSP</i>
gw	0.072	viral	0.214
gk	0.065	kerja	0.095
udh	0.061	usut	0.060
lu	0.044	cepat	0.054
tinggal	0.039	media	0.041
mati	0.039	tanggap	0.036
ni	0.034	tangan	0.030
berani	0.026	kapolri	0.027
minggu	0.025	hormat	0.023
ilang	0.020	miris	0.022

Catatan:

*TOSP* : Term (kata) Yang Didapatkan Tanpa Algoritma Spell Checker

*POSP* : Nilai Peluang Term (kata) Yang Didapatkan Tanpa Algoritma Spell Checker

*TWSP* : Term (kata) Yang Didapatkan Dengan Algoritma Spell Checker

*PWSP* : Nilai Peluang Term (kata) Yang Didapatkan Dengan Algoritma Spell Checker

Kesimpulan pada Topik 2 ini adalah sebelum *spell checker* diimplementasikan *term* (kata) yang dihasilkan tidak begitu menggambarkan topik yang jelas. Namun setelah algoritma *spell checker* diimplementasikan maka nampak suatu topik yang membahas tentang persoalan kasus bunuh diri seorang

wanita yang dihormati oleh seorang oknum polisi yang viral di rentang waktu pengambilan data, terlihat netizen meminta segera diusut oleh kapolri sehingga tidak terulang kasus yang miris tersebut.

### 3.4 Pembahasan Topik 3

**Tabel 4. Hasil pada topik 3 tanpa spell checker dan menggunakan *spell checker***

<i>TOSP</i>	<i>POSP</i>	<i>TWSP</i>	<i>PWSP</i>
to	0.045	adil	0.214
you	0.047	moga	0.095
perempuan	0.036	cari	0.060
miris	0.026	allah	0.054
laki	0.026	jalan	0.041
benar	0.022	orang	0.036
be	0.020	laku	0.030
that	0.020	adu	0.027
are	0.019	buat	0.023
muka	0.019	selesai	0.022

Catatan:

*TOSP* : Term (kata) Yang Didapatkan Tanpa Algoritma Spell Checker

*POSP* : Nilai Peluang Term (kata) Yang Didapatkan Tanpa Algoritma Spell Checker

*TWSP* : Term (kata) Yang Didapatkan Dengan Algoritma Spell Checker

*PWSP* : Nilai Peluang Term (kata) Yang Didapatkan Dengan Algoritma Spell Checker

Kesimpulan pada Topik 3 ini adalah sebelum *spell checker* diimplementasikan term (kata) yang dihasilkan tidak begitu menggambarkan topik yang jelas. Namun setelah algoritma *spell checker* diimplementasikan maka nampak suatu topik yang membahas tetap seperti topik 2 sebelumnya berisi tentang harapan-harapan netizen agar pelaku dihukum dengan seadil-adilnya.

### 3.5 Pembahasan Topik 4

**Tabel 5. Hasil pada topik 4 tanpa spell checker dan menggunakan *spell checker***

<i>TOSP</i>	<i>POSP</i>	<i>TWSP</i>	<i>PWSP</i>
hukum	0.196	netizen	0.063
amp	0.052	tangkap	0.054
tegak	0.048	berita	0.041
indonesia	0.030	hidup	0.040
kasus	0.027	mati	0.037
harga	0.024	laku	0.033
jaga	0.024	teman	0.032
sm	0.019	beda	0.031
ganti	0.017	lo	0.031
selesai	0.015	dapat	0.031

Catatan:

*TOSP* : Term (kata) Yang Didapatkan Tanpa Algoritma Spell Checker

*POSP* : Nilai Peluang Term (kata) Yang Didapatkan Tanpa Algoritma Spell Checker

*TWSP* : Term (kata) Yang Didapatkan Dengan Algoritma Spell Checker

*PWSP* : Nilai Peluang Term (kata) Yang Didapatkan Dengan Algoritma Spell Checker

Kesimpulan pada Topik 4 ini adalah sebelum *spell checker* diimplementasikan term (kata) yang dihasilkan tidak begitu menggambarkan topik yang jelas. Namun setelah algoritma *spell checker* diimplementasikan maka nampak suatu topik yang membahas tentang persoalan netizen yang

menunjukkan ketakannya dalam memviralkan suatu kasus disebabkan karena kasus tersebut akan diproses jika diviralkan.

#### 4. KESIMPULAN

Dalam penelitian ini, kami telah melakukan *topic modeling* terkait kinerja polisi Indonesia dengan 2 metode berbeda. Hasil studi ini, dapat disimpulkan bahwa pendeteksian penerapan *topic modeling* menggunakan algoritma *spell checker* terbukti berhasil meningkatkan akurasi model sehingga menghasilkan topik yang relevan terhadap keluhan dan kemarahan *netizen* dalam mengkritik kinerja kepolisian republik Indonesia.

#### DAFTAR PUSTAKA

- [1] D. Ramamonjisoa, "Topic Modeling on Users ' s Comments," pp. 177–180, 2014.
- [2] S. Tan et al., "Interpreting the Public Sentiment Variations on Twitter," vol. 26, no. 5, pp. 1158–1170, 2014.
- [3] Z. Jiang and X. Zhou, "Using Link Topic Model to Analyze Traditional Chinese Medicine Clinical Symptom-Herb Regularities," pp. 15–18, 2012.
- [4] M. J. Paul and M. Dredze, "You Are What You Tweet : Analyzing Twitter for Public Health," pp. 265–272, 2011.
- [5] B. Chen, L. Zhu, D. Kifer, and D. Lee, "What Is an Opinion About ? Exploring Political Standpoints Using Opinion Scoring Model," pp. 1007–1012.
- [6] R. Cohen and D. Ruths, "Classifying Political Orientation on Twitter : It ' s Not Easy !," pp. 91–99, 2012.
- [7] B. O. Connor, N. A. Smith, and E. P. Xing, "A Latent Variable Model for Geographic Lexical Variation," 2008.
- [8] M. Cristani, A. Perina, U. Castellani, and D. Informatica, "Geo-located image analysis using latent representations," 2008.
- [9] M. Gethers and D. Poshyvanyk, "Using Relational Topic Models to Capture Coupling among Classes in Object-Oriented Software Systems," 2010.
- [10] H. U. Asuncion, A. U. Asuncion, and R. N. Taylor, "Software Traceability with Topic Modeling," pp. 95–104, 2010.
- [11] N. I. V. K. JOVITA ANGELA, "PANDEMI COVID-19 BAGI PENERAPAN INOVASI DI INDONESIA: ANALISIS PENAMBANGAN TEKS BERITA DENGAN MODEL TOPIK LATENT DIRICHLET ALLOCATION," UGM, 2021.
- [12] Y. A. W. DAMAWAN BAYU AJI, "Analisis Ulasan Pelanggan dalam Penerapan M-Payment di Industri E-Commerce," UGM, 2019.
- [13] R. Sistem, Y. Sahria, and D. H. Fudholi, "JURNAL RESTI Analisis Topik Penelitian Kesehatan di Indonesia Menggunakan Metode," vol. 1, no. 10, pp. 336–344, 2021.
- [14] H. Jelodar, Y. Wang, C. Yuan, X. Feng, X. Jiang, and Y. Li, "Latent Dirichlet allocation ( LDA ) and topic modeling : models , applications , a survey," pp. 15169–15211, 2019.
- [15] A. Krouska, C. Troussas, and M. Virvou, "The effect of preprocessing techniques on Twitter Sentiment Analysis."

- [16] M. Maybury, "New Direction In Question Answering," AAAI Press. MIT Press, 2004.
- [17] J. C. Campbell, A. Hindle, and E. Stroulia, "Latent Dirichlet Allocation: Extracting Topics from Software Engineering Data," *Art Sci. Anal. Softw. Data*, vol. 3, pp. 139–159, 2015, doi: 10.1016/B978-0-12-411519-4.00006-9.
- [18] D. Blei, L. Carin, and D. Dunson, "Probabilistic topic models," *IEEE Signal Process. Mag.*, vol. 27, no. 6, pp. 55–65, 2010, doi: 10.1109/MSP.2010.938079.
- [19] K. Wang, "PANet: Few-Shot Image Semantic Segmentation with Prototype Alignment," pp. 9197–9206.