Penerapan Klasifikasi Kueri untuk Meningkatkan Efektivitas Mesin Pencari

(Implementation of Query Classification to Improve Effectiveness of Search Engine)

Handy Geraldy[1], Lutfi Rahmatuti Maghfiroh[2]

[1,2] *Politeknik Statistika STIS*
Jl. Otto Iskandardinata No.64C, RT.1/RW.4, Bidara Cina, Kecamatan Jatinegara, DKI Jakarta
E-mail: 221709728@stis.ac.id

ABSTRAK

Dalam menjalankan peran sebagai penyedia data, Badan Pusat Statistik (BPS) memberikan layanan akses data BPS bagi masyarakat. Salah satu layanan tersebut adalah fitur pencarian di *website* BPS. Namun, layanan pencarian yang diberikan belum memenuhi harapan konsumen. Untuk memenuhi harapan konsumen, salah satu upaya yang dapat dilakukan adalah meningkatkan efektivitas pencarian agar lebih relevan dengan maksud pengguna. Oleh karena itu, penelitian ini bertujuan untuk membangun fungsi klasifikasi kueri pada mesin pencari dan menguji apakah fungsi tersebut dapat meningkatkan efektivitas pencarian. Fungsi klasifikasi kueri dibangun menggunakan model *machine learning*. Kami membandingkan lima algoritma yaitu *Naïve-Bayes, K-NN, SVM, Random Forest, dan Gradient Boosting*. Dari lima algoritma tersebut, model terbaik diperoleh pada algoritma SVM. Kemudian, fungsi tersebut diimplementasikan pada mesin pencari yang diukur efektivitasnya berdasarkan nilai *precision* dan *recall*. Hasilnya, fungsi klasifikasi kueri dapat mempersempit hasil pencarian pada kueri tertentu, sehingga meningkatkan nilai precision. Namun, fungsi klasifikasi kueri tidak memengaruhi nilai *recall*.

Kata kunci: mesin pencari, klasifikasi kueri, *machine learning*

ABSTRACT

As a data provider, BPS provides services for the public to access BPS data. One of these services is the search service on the BPS website. However, the search services have not met consumer expectations. One of the efforts that can be done to meet consumer expectations is to increase search effectiveness to make it more relevant to the user's intent. Therefore, this study aims to build a query classification function on a search engine and test whether this function can improve search effectiveness. The query classification function is built using a machine learning model. We compare five algorithms, those are Naïve-Bayes, K-NN, SVM, Random Forest, and Gradient Boosting. Of the five algorithms used, the best model is obtained from the SVM algorithm. Then, the function is implemented on a search engine whose effectiveness is measured based on precision and recall values. As a result, the query classification function can narrow the search results to a particular query, thereby increasing the precision value. However, the query classification function does not affect the recall value.

***Keywords**: search engine, query classification, machine learning*

PENDAHULUAN

Berdasarkan UU No. 16 tahun 1997 tentang Statistik, Badan Pusat Statistik (BPS) bertugas untuk menyebarluaskan hasil statistik yang diselenggarakannya. Dalam pelaksanaan tugas tersebut, BPS melakukan berbagai upaya guna menyebarluaskan data melalui unit Pelayanan Statistik Terpadu (PST). Berdasarkan publikasi BPS yang berjudul Analisis Hasil Survei Kebutuhan Data 2020, 41,88% konsumen di PST BPS Pusat mendapatkan data melalui fasilitas *website* BPS. Namun, atribut pencarian data di *website* BPS termasuk dalam empat atribut dengan persentase kepuasan terendah. Berdasarkan *gap analysis* kepuasan konsumen terhadap pelayanan BPS pada publikasi tersebut, pencarian data di *website* BPS memiliki nilai gap paling jauh, yaitu sebesar -0,31. Hal tersebut menunjukkan bahwa kinerja pada atribut pencarian data masih belum memenuhi harapan konsumen di PST BPS Pusat. Sehingga, layanan pencarian data pada *website* BPS merupakan prioritas perbaikan utama pada Pelayanan Statistik Terpadu (PST) BPS Pusat.

Secara umum, pencarian data di *website* BPS (www.bps.go.id) dapat dilakukan dengan melalui beberapa menu, seperti pencarian berdasarkan subjek data, pencarian publikasi, dan *Allstats Search*. Pada pencarian tabel data berdasarkan subjek, pengguna harus memilih subjek data terlebih dahulu kemudian mencari tabel yang diinginkan berdasarkan daftar yang ditampilkan pada subjek yang dipilih. Di sini, terdapat kolom pencarian yang membatasi pencarian berdasarkan kueri yang diisi oleh pengguna. Untuk

menggunakan kolom pencarian tersebut, kueri pengguna harus tepat sama dengan judul tabel. Kemudian, pada pencarian publikasi, pengguna dapat mencari publikasi yang diterbitkan BPS. Pencarian tersebut terbatas pada judul dan abstraksi serta tidak menunjukkan daftar data yang ada pada setiap publikasi. Selanjutnya, pada pencarian data melalui mesin pencari di *Allstats search*, pengguna dapat memasukkan kueri, membatasi pencarian berdasarkan judul, dan melakukan filter hasil pencarian berdasarkan jenis konten, tahun, dan wilayah. Pencarian data pada *Allstats search* masih sederhana karena setiap pengaturan harus dilakukan secara manual. Selain itu, *Allstats Search* hanya memberikan hasil yang tepat sama dengan kata yang diisi pengguna, belum bisa mengenali kata dasar, salah ketik, dan memahami maksud pengguna. Berdasarkan menu yang ada untuk melakukan pencarian data di *website* BPS, upaya yang dapat dilakukan untuk meningkatkan layanan pencarian data di *website* BPS adalah meningkatkan efektivitas pencarian data pada mesin pencari *Allstats Search*.

Mesin pencari telah menjadi alat utama bagi orang untuk mendapatkan informasi yang diinginkan (Zou, Cheng, & Men, 2017). Menurut Keane, O'Brien & Smyth (2008), pengguna mesin pencari hanya melihat hasil pencarian pada halaman pertama. Untuk memenuhi kebutuhan pengguna, mesin pencari harus menyaring dan menemukan informasi yang paling relevan dengan permintaan pengguna (Meshram et al, 2018). Oleh karena itu, agar layanan pencarian data dapat memenuhi harapan konsumen PST BPS Pusat, mesin pencari yang dibangun harus efektif dalam menampilkan hasil yang relevan dengan maksud pengguna, terutama pada halaman pertama.

Peningkatan efektivitas mesin pencari untuk menampilkan data yang relevan dapat dilakukan dengan menyediakan fungsi pencarian yang memahami maksud pengguna. Menurut Qiu et al (2018), penelitian tentang teknologi pencarian informasi berdasarkan maksud pengguna telah banyak dilakukan setelah taksonomi pencarian web disusun oleh Broder (2002). Salah satu arah penelitian tersebut adalah penerapan klasifikasi kueri pada mesin pencari. Klasifikasi kueri bertujuan untuk mengklasifikasikan kueri ke kategori yang telah ditentukan untuk lebih memahami kebutuhan pengguna (Zhang et al, 2019).

Pada penelitian Qiu et al (2018) telah dibangun metode klasifikasi kueri menggunakan *LSTM similarity* dan *time sequence model* untuk mengenali maksud dari kueri dalam pencarian data individu. Kemudian, penelitian Bortnikova et al (2019) telah membangun model *neural network* yang diintegrasikan dengan mesin pencari guna mengklasifikasikan kueri terhadap lima kategori. Hasilnya, penerapan model tersebut dapat meningkatkan kecepatan pengolahan pada mesin pencari. Selain itu, penelitian Yu & Litchfield (2020) membangun model klasifikasi kueri bertingkat. Pada model tersebut, kueri pengguna tidak dikategorikan sampai tingkatan terdalam jika tingkatan tersebut berpotensi memberikan hasil yang salah. Model tersebut akan mengembalikan kategori pada tingkatan yang paling sesuai dengan mengoptimalkan pertukaran nilai akurasi dan kedalaman kategori,

Atas dasar hal tersebut, peneliti membangun model untuk melakukan klasifikasi kueri secara otomatis pada data BPS dan membandingkan efektivitas mesin pencari yang memuat fungsi klasifikasi kueri terhadap mesin pencari yang tidak memuat fungsi tersebut.

METODE

Pemodelan *Machine Learning*

Pada penelitian ini, fungsi klasifikasi kueri dibangun menggunakan algoritma *machine learning* untuk mengklasifikasikan teks. Klasifikasi teks merupakan proses mengkategorikan dokumen ke dalam satu atau lebih kelas yang telah ditentukan sesuai dengan subjeknya secara otomatis (Kadhim, 2019). Proses klasifikasi teks terdiri dari tahapan *Natural Language Processing* (NLP), *text vectorization*, dan pemodelan *machine learning* (Ikonomakis, Kotsiantis, dan Tampakas, 2005).

Tahap pertama klasifikasi teks adalah NLP. NLP merupakan teknik komputasi yang digunakan untuk merepresentasikan bahasa alami ke dalam representasi yang berguna untuk pemrosesan lebih lanjut (Yogish, Manjunath, & Hegadi, 2018). NLP terdiri dari case folding, membuang tanda baca dan angka, tokenisasi, membuang *stopword*, dan *stemming*. Pemrosesan tersebut dilakukan untuk menyamakan bentuk penulisan dan membuang elemen yang tidak perlu atau mengganggu dalam pembentukan model.

Selanjutnya, dilakukan *text vectorization*. *Text vectorization* dilakukan untuk mengubah data teks yang tidak terstruktur menjadi terstruktur dan memberikan bobot untuk setiap kata. Terdapat beberapa ukuran *text vectorization*, seperti *binary term frequency*, *bag of words* (BOW) *term frequency*, *normalized term frequency*, *normalized term frequency – inverse document frequency* (TF-IDF), dan *word2vec*. Pada penelitian ini digunakan ukuran TF-IDF.

Kemudian, pada tahap pemodelan *machine learning* terdapat beberapa algoritma yang akan diterapkan, yaitu *naıve-Bayes, nearest neighbors, support vector machines, random forest,* dan *gradient boosting*. Peneliti membandingkan *metrics* dari beberapa algoritma *machine learning* tersebut. Setiap algoritma yang digunakan telah melalui proses *tuning* terlebih dahulu agar memberikan hasil yang optimal. Ukuran *metrics* yang digunakan adalah akurasi dan F1-score.

Berikut merupakan penjelasan dari setiap algoritma yang diterapkan:

1. *Naïve-Bayes*

Metode *Naïve-Bayes* merupakan pengklasifikasi probabilitas sederhana yang didasarkan pada teori Bayes. Teori Bayes menyatakan bahwa peluang terjadinya suatu peristiwa dapat dihitung berdasarkan pengalaman di masa lalu. Metode *Naïve-Bayes* didasarkan pada asumsi yang kuat bahwa peluang suatu kejadian bersifat independen terhadap kejadian lainnya (McCallum & Nigam,1998).

2. *K-Nearest Neighbors (K-NN)*

K-Nearest Neighbors (K-NN) merupakan metode klasifikasi nonparametrik yang dikembangkan oleh Evelyn Fix dan Joseph Hodges. Metode ini bekerja dengan menghitung jarak terdekat antara input terhadap keseluruhan dataset, memilih data sejumlah K yang paling dekat dengan input, menentukan label mayoritas dari sejumlah data tersebut, dan menyatakan label input berdasarkan label mayoritas. Pada algoritma K-NN, normalisasi data latih dapat meningkatkan akurasi model secara signifikan. Hal ini terjadi jika fitur yang digunakan berasal menggunakan skala atau satuan yang berbeda (Piryonesi & El-Diraby, 2020).

3. *Support Vector Machines (SVM)*

Dalam hal klasifikasi, *Support Vector Machines* (SVM) merupakan metode yang bekerja dengan mencari *hyperplane* terbaik dengan memaksimalkan jarak antar kelas. *Hyperplane* adalah sebuah fungsi yang dapat digunakan untuk pemisah antar kelas (Franklin, 2005). *Hyperplane* terbaik adalah fungsi yang memiliki jarak terjauh terhadap objek terdekat dari fungsi tersebut. SVM dapat menangani klasifikasi non-linear dengan membangun fungsi hyperplane yang non-linear pula, hal ini disebut *kernel tricks*.

4. *Random Forest*

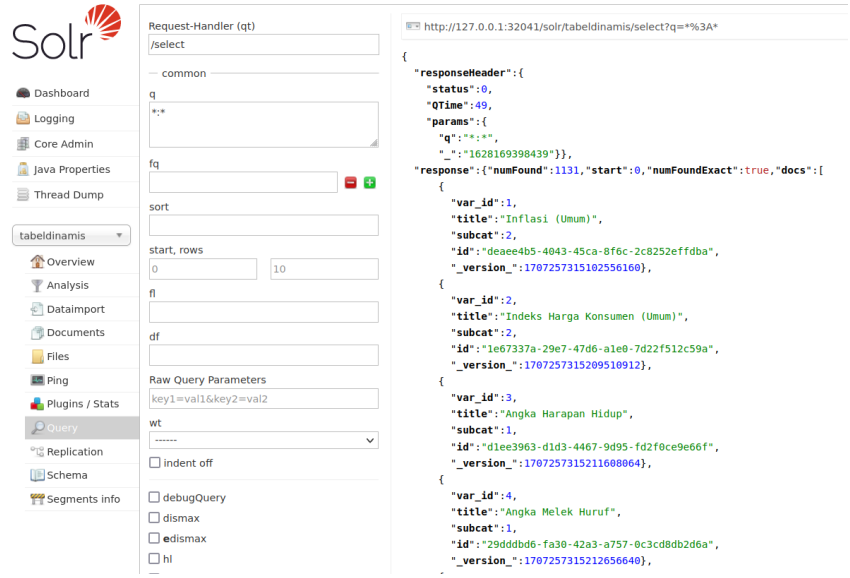
Metode *Random Forest* merupakan metode ensembel yang mengombinasikan sekumpulan *decision tree* yang dapat dibangun tanpa pemotongan. Hasil prediksi pada Random Forest merupakan ringkasan dari prediksi setiap *decision tree* penyusunnya (Lin et al, 2017). Dalam pemodelannya, Random Forest melalui proses pengacakan sampel dan pemilihan variabel bebas sehingga pohon klasifikasi yang dibangkitkan akan memiliki ukuran dan bentuk yang beragam (Liaw & Wiener, 2002).

5. *Gradient Boosting (GBT)*

Metode *Gradient Boosting* (GBT) pertama kali diperkenalkan oleh J.H. Friedman. GBT merupakan metode ensembel yang membangun sejumlah *decision tree*. Setiap *tree* yang dibangun mengacu pada kelemahan *tree* sebelumnya. Dasar pemikiran GBT adalah model terbaik berikutnya akan meminimalkan kesalahan prediksi keseluruhan jika dikombinasikan dengan model sebelumnya (Friedman, 2001).

Pembangunan Mesin Pencari

Tahap selanjutnya adalah pembangunan mesin pencari dengan mengimplementasikan model *machine learning* yang diperoleh pada tahap sebelumnya. Pada penelitian ini, mesin pencari tidak dibangun dari awal, tetapi memanfaatkan platform pencarian yang sudah ada, yaitu Apache Solr. Untuk menggunakan Solr, pertama dilakukan penyusunan indeks dari dokumen pencarian berupa tabel dinamis. Penyusunan indeks merupakan proses identifikasi kata-kata yang terdapat pada dokumen pencarian sehingga tersimpan pada basis data mesin pencari. Penyusunan indeks dilakukan dengan menggunakan fitur *indexing* yang tersedia pada Solr. Gambar 1 menunjukkan indeks yang telah tersimpan pada Apache Solr.



Gambar 1. Indeks mesin pencari pada Apache Solr.

```
url_solr = 'http://127.0.0.1:32041/solr/tabeldinamis'

def akses_solr(query, url_solr, row):
    q = 'q={!dismax qf=title mm=1 v="'+ query + '"} AND subcat:' + str(subcat)
    res_pub = requests.get(solr_url+'/query', params={
        'q': q,
        'start': 0,
        'rows': row,
        'wt': 'json',
    })
    solr_response = res_pub.json()['response']
    df = pd.json_normalize(solr_response, record_path =['docs'])
    return df
```

Gambar 2. Fungsi akses Solr.

Setelah itu, dibangun *back-end* mesin pencari untuk mengolah kueri pengguna, menjalankan model *machine learning*, melakukan *request* data pada Solr. Back-end mesin pencari dibangun menggunakan bahasa pemrograman Python. Gambar 2 menunjukkan fungsi *akses_solr* yang digunakan sebagai *back-end* mesin pencari. Pada Gambar 2, fungsi akses_solr digunakan untuk mengakses data pada Solr. Fungsi ini memerlukan tiga parameter. Parameter *query* merupakan kueri yang dimasukkan oleh pengguna, *url_solr* merupakan *hostname yang* memuat daftar indeks, dan *row* adalah jumlah baris maksimal yang ingin ditampilkan. Fungsi ini memberikan *output* berupa *dataframe* yang memuat hasil pencarian untuk kueri yang dicari.

Evaluasi Hasil Pencarian

Evaluasi dilakukan untuk mengukur apakah mesin pencari yang memuat fungsi klasifikasi kueri meningkatkan efektivitas pencarian. Ukuran efektivitas yang dibandingkan adalah nilai *precision* dan *recall* pada mesin pencari yang memuat fungsi klasifikasi kueri terhadap mesin pencari tanpa fungsi klasifikasi kueri.

Precision dan *recall* merupakan ukuran yang dapat digunakan untuk mengevaluasi efektivitas mesin pencari. *Precision* merupakan ukuran kualitas pencarian informasi yang benar secara akurat (Shafi & Rather, 2005). *Precision* mengukur tingkat ketepatan antara kata kunci yang diminta dengan hasil pencarian yang diberikan oleh mesin pencari. Sedangkan, *recall* mengukur tingkat ketepatan antara kata kunci dengan dengan total dokumen yang relevan. Berikut ini merupakan rumus *precision* dan *recall* menurut Usmani, Pant, & Bhatt (2012):

$$Precision = \frac{Jumlah\ Dokumen\ Relevan\ yang\ Terambil}{Jumlah\ Dokumen\ yang\ Terambil} \quad \text{.....(1)}$$

$$Recall = \frac{Jumlah\ Dokumen\ Relevan\ yang\ Terambil}{Jumlah\ Dokumen\ yang\ Relevan} \quad \text{.....(2)}$$

Untuk membandingkan nilai *precision* dan *recall*, peneliti melakukan eksperimen dengan menjalankan mesin pencari pada kueri tertentu. Kueri yang digunakan dapat dibagi menjadi tiga jenis, yaitu kata (*simple*), frasa (*compound*), dan kalimat (*complex*) (Kumar & Prkash, 2009). Kueri yang digunakan pada eksperimen tersebut antara lain:

1 Inflasi
2 IPM
3 Jumlah peternakan
4 Upah tenaga kerja
5 Produksi perkebunan di Sumatera tahun 2018
6 Persentase Rumah Tangga yang Menguasai Telepon Seluler memiliki rumah sendiri

Data dan Sumber Data

Data yang digunakan pada penelitian ini adalah data tabel dinamis yang diperoleh melalui web API BPS. API BPS merupakan salah satu layanan BPS untuk mengakses data BPS. Data yang dapat diakses meliputi publikasi, berita resmi statistik atau siaran pers, infografis, dan berbagai macam data yang disajikan dalam tabel statis dan tabel dinamis. Akses ke data tersebut bermanfaat untuk mengintegrasikan data BPS dengan aplikasi lain. Sebelum mengakses API BPS, peneliti membuat akun pada portal API BPS untuk mendapatkan API key. API key diperlukan untuk mengakses setiap layanan pada API BPS.

Data tersebut meliputi judul tabel, nama baris dan kolom, isi tabel, dan keterangan tabel seperti subjek data, satuan yang digunakan, dan catatan. Pada pemodelan *machine learning*, variabel prediktor yang digunakan adalah judul tabel, nama baris, dan nama kolom. Sedangkan variabel target yang digunakan adalah *subject category* tabel yang terdiri dari tiga label, yaitu Sosial dan Kependudukan, Ekonomi dan Perdagangan, dan Pertanian dan Pertambangan.

HASIL DAN PEMBAHASAN

Data yang diperoleh dari web API BPS sejumlah 1131 tabel dinamis. Data tersebut dimanfaatkan sebagai data latih pada untuk membangun fungsi klasifikasi kueri. Fungsi klasifikasi kueri dibangun dengan menggunakan model *machine learning* untuk mengklasifikasikan kueri pengguna ke dalam *subject category* BPS, yaitu Sosial dan Kependudukan, Ekonomi dan Perdagangan, serta Pertanian dan Pertambangan. Peneliti membandingkan lima algoritma pemodelan, yaitu *Naïve-Bayes*, *K-NN*, *SVM*, *Random Forest*, dan *Gradient Boosting*. Setiap algoritma dioptimalkan hingga mendapat parameter terbaik, kemudian diuji pada *data test* berupa judul tabel publikasi yang diberi label terlebih dahulu. Hasil evaluasi setiap algoritma terlihat pada Tabel 1.

Berdasarkan Tabel 1, nilai akurasi yang dihasilkan pada setiap algoritma sudah baik. Nilai akurasi selalu di atas 90% kecuali pada algoritma *Naïve-Bayes*. Untuk nilai f1_score, setiap algoritma menghasilkan nilai f1_score tertinggi pada label 2, yaitu Ekonomi dan Perdagangan dan nilai f1_score terendah pada label 3, yaitu Pertanian dan Pertambangan. Dari lima algoritma yang digunakan, SVM memberikan nilai terbaik untuk ukuran akurasi, f1_score label 1, dan f1_score label 3. Sedangkan pada ukuran f1_score label 2, algoritma SVM tidak lebih baik dari algoritma K-NN. Oleh karena itu, model SVM dipilih sebagai model terbaik yang akan digunakan pada fungsi klasifikasi kueri.

Selanjutnya, peneliti membandingkan hasil pencarian pada mesin pencari yang mengkategorikan kueri secara otomatis dengan fungsi klasifikasi kueri terhadap mesin pencari yang tidak memuat fungsi klasifikasi kueri. Tabel 2 menunjukkan nilai *precision* dan *recall* pada mesin pencari yang memuat fungsi klasifikasi kueri dengan mesin pencari yang tidak memuat fungsi tersebut untuk seluruh hasil yang diberikan.

Tabel 1. Nilai metrics algoritma machine learning

Algoritma	Metrics (%)			
	Akurasi	F1_score 1	F1_score 2	F1_score 3
SVM (C=1, gamma=1, kernel='rbf')	92,41	90,24	94,65	85,96
K-NN (n=7, weight='distance', metric='manhattan')	92,32	90,04	94,93	84,57
Random Forest (n_estimators=100, min_samples_split=5)	91,17	88,11	94,47	81,74
Gradient Boosting (n_estimators=200, max_depth=8, learning_rate=0,15)	90,44	86,74	93,22	85,16
Naive-Bayes (multinomial)	86,39	79,02	89,77	85,01

Sumber: Hasil olahan

Tabel 2. Perbandingan nilai precision dan recall pada mesin pencari yang memuat dan tidak memuat fungsi klasifikasi kueri.

Kueri	Precision		Recall	
	Dengan Fungsi	Tanpa Fungsi	Dengan Fungsi	Tanpa Fungsi
Inflasi	100	100	100	100
IPM	100	100	100	100
Jumlah peternakan	45,46	6,58	100	100
Upah tenaga kerja	58	39,66	100	100
Produksi perkebunan di Sumatera tahun	5	1,26	100	100
Persentase Rumah Tangga yang	7	2,49	100	100

Sumber: Hasil olahan

Berdasarkan Tabel 2, nilai *precision* pada mesin pencari dengan fungsi klasifikasi kueri lebih tinggi daripada mesin pencari tanpa fungsi klasifikasi kueri pada 4 kueri, yaitu "jumlah peternakan", "upah tenaga kerja", "produksi perkebunan di Sumatera tahun 2018", dan "Persentase Rumah Tangga yang Menguasai Telepon Seluler memiliki rumah sendiri". Sedangkan, pada kueri "inflasi" dan "IPM", nilai keduanya sama karena seluruh tabel yang relevan dengan kata inflasi maupun IPM berada pada kategori yang sama. Inflasi terletak pada kategori Ekonomi dan Perdagangan, sedangkan IPM terletak pada kategori Sosial dan Kependudukan. Kemudian, istilah inflasi dan IPM merupakan istilah yang spesifik merujuk pada tabel tertentu sehingga nilai *precision* yang dihasilkan sangat baik, yaitu 100%. Untuk nilai *recall*, kedua mesin pencari menghasilkan nilai yang sama. Lima kueri yang diberikan menghasilkan nilai *recall* 100% karena penamaan judul tabel dinamis BPS telah konsisten. Dengan kata lain, istilah yang sama ditulis dengan kata-kata yang sama.

Kemudian, peneliti mengukur nilai *precision* untuk 10 hasil teratas saja. Nilai ini diukur karena menurut penelitian Jansen & Spink (2006) dan Keane, O'Brien & Smyth (2008), pengguna hanya melihat hasil pencarian pada halaman pertama atau 10 hasil teratas saja. Tabel 3 menunjukkan nilai *precision* untuk 10 hasil pertama yang ditampilkan mesin pencari. Berdasarkan Tabel 3, nilai *precision* pada mesin pencari dengan fungsi klasifikasi lebih tinggi daripada mesin pencari tanpa fungsi pencarian pada 3 kueri, yaitu "jumlah peternakan", "produksi perkebunan di Sumatera tahun 2018", dan "Persentase Rumah Tangga yang Menguasai Telepon Seluler memiliki rumah sendiri", sedangkan pada kueri lainnya, nilai kedua mesin pencari tersebut sama.

Tabel 3. Perbandingan nilai precision dan recall pada mesin pencari yang memuat dan tidak memuat fungsi klasifikasi kueri untuk 10 hasil teratas.

Kueri	Precision	
	Dengan Fungsi	Tanpa Fungsi
Inflasi	100	100
IPM	100	100
Jumlah peternakan	80	40
Upah tenaga kerja	20	20
Produksi perkebunan di Sumatera tahun 2018	70	50
Persentase Rumah Tangga yang Menguasai Telepon Seluler memiliki rumah sendiri	70	30

Sumber: Hasil olahan

KESIMPULAN

Pada penelitian ini telah dibangun fungsi klasifikasi kueri berdasarkan model *machine learning* untuk mengkategorikan teks berupa kueri pengguna pada mesin pencari ke dalam *subject category* BPS. Dari lima algoritma yang digunakan, model terbaik diperoleh pada algoritma SVM. Kemudian, model tersebut diimplementasikan untuk membatasi pencarian pada *subject category* tertentu. Evaluasi yang dilakukan menunjukkan bahwa fungsi klasifikasi kueri dapat mempersempit hasil pencarian pada kueri tertentu, sehingga meningkatkan nilai *precision*. Meningkatnya nilai *precision* menunjukkan bahwa mesin pencari dengan fungsi klasifikasi lebih efektif karena dapat mengurangi hasil yang tidak sesuai terhadap kueri pengguna. Untuk nilai *recall*, kedua mesin pencari memberikan nilai yang sama, sehingga dapat dikatakan bahwa fungsi klasifikasi kueri tidak memengaruhi nilai *recall*.

DAFTAR PUSTAKA

Badan Pusat Statistik. (2021). *Badan Pusat Statistik*. https://www.bps.go.id [5 Agustus 2021].

Badan Pusat Statistik. (2021) *Allstats Search*. https://www.bps.go.id/searchengine/ [5 Agustus 2021].

Badan Pusat Statistik. (2020). *Analisis Hasil Survei Kebutuhan Data 2020*. Jakarta: BPS

Bortnikova, V., Nevliudov, I., Botsman, I., & Chala, O. (2019, June). Search Query Classification Using Machine Learning for Information Retrieval Systems in Intelligent Manufacturing. In *ICTERI* (pp. 460-465).

Broder, A. (2002, September). A taxonomy of web search. In *ACM Sigir forum* (Vol. 36, No. 2, pp. 3-10). New York, NY, USA: ACM.

Franklin, J. (2005). The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer*, 27(2), 83-85.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. Annals of statistics, 1189-1232.

Kadhim, A. I. (2019). Survey on supervised machine learning techniques for automatic text classification. *Artificial Intelligence Review*, 52(1), 273-292.

Keane, M. T., O'Brien, M., & Smyth, B. (2008). Are people biased in their use of search engines?. *Communications of the ACM*, 51(2), 49-52.

Kumar, B. S., & Prakash, J. N. (2009). Precision and relative recall of search engines: A comparative study of Google and Yahoo. *Singapore Journal of Library & Information Management*, 38(1), 124-137.

Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. R news, 2(3), 18-22.

Lin, W., Wu, Z., Lin, L., Wen, A., & Li, J. (2017). An ensemble random forest algorithm for insurance big data analysis. *Ieee access*, 5, 16568-16575.

McCallum, A., & Nigam, K. (1998, July). A comparison of event models for naive bayes text classification. In AAAI-98 workshop on learning for text categorization (Vol. 752, No. 1, pp. 41-48).

Meshram, P., Katole, T., Parate, N., Allewar, A., Dafre, S., Patil, S., & Naveed, S. (2018). Survey on Search Engine Localization and Optimization.

Piryonesi, S. M., & El-Diraby, T. E. (2020). Role of data analytics in infrastructure asset management: Overcoming data size and quality problems. *Journal of Transportation Engineering, Part B: Pavements*, 146(2), 04020022.

Raval, V., & Kumar, P. (2012, March). SEReleC (Search Engine Result Refinement and Classification)-a Meta search engine based on combinatorial search and search keyword based link classification. In *IEEE-International Conference On Advances In Engineering, Science And Management* (ICAESM-2012) (pp. 627-631). IEEE.

RI (Republik Indonesia). (1997). Undang-Undang No. 16 Tahun 1997 tentang Statistik. Lembaran Negara RI No. 3683. Sekretariat Negara. Jakarta.

Shafi, S. M., & Rather, R. A. (2005). Precision and recall of five search engines for retrieval of scholarly information in the field of biotechnology. *Webology*, *2*(2), 42-47.

Usmani, T. A., Pant, D., & Bhatt, A. K. (2012). A comparative study of Google and Bing search engines in context of precision and relative recall parameter. *International Journal on Computer Science and Engineering*, 4(1), 21.

Qiu, L., Chen, Y., Jia, H., & Zhang, Z. (2018). Query intent recognition based on multi-class features. *IEEE Access*, 6, 52195-52204.

Yogish, D., Manjunath, T. N., & Hegadi, R. S. (2018, December). Review on natural language processing trends and techniques using nltk. In *International Conference on Recent Trends in Image Processing and Pattern Recognition* (pp. 589-606). Springer, Singapore.

Yu, H., & Litchfield, L. (2020, July). Query Classification with Multi-objective Backoff Optimization. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 1925-1928).

Zhang, H., Song, W., Liu, X., Liu, L., & Zhao, X. (2019, March). Query Classification Based on Automatic Learning Query Representation. In *2019 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)* (pp. 39-42). IEEE.

Zhou, S., Cheng, K., & Men, L. (2017, April). The survey of large-scale query classification. *In AIP conference proceedings* (Vol. 1834, No. 1, p. 040045). AIP Publishing LLC.