

IMPLEMENTATION of MULTI-LABEL CLASSIFICATION to DETERMINE SCIENTIFIC ARTICLES KEYWORD in JOURNALS

Implementasi Multi-Label Classification Untuk Penentuan Keyword Artikel Ilmiah pada Jurnal

Marvel Manueke, Rizal Sengkey, Agustinus Jacobus.

Dept. of Electrical Engineering, Sam Ratulangi University Manado, Kampus Bahu St., 95115, Indonesia

e-mails : 15021106056@student.unsrat.ac.id, rizalsengkey@gmail.com, a.jacobus@unsrat.ac.id

Received:; revised:; accepted:

Abstract — *Scientific articles are used as a communication forum for members of the scientific community for certain disciplines. because they are read, discussed, and disseminated by members of the scientific community concerned. This makes determining keyword arguably an important part of the making of scientific journals. Keyword not only act as a label of the article's topics but also as a way of searching and grouping them. Determining those keywords is usually done personally by the author based on the title or article's body, which makes keywords tend to differ between journals even when their topics are the same. for this very reason, this research aims to make an app that can determine keywords automatically based on a multi-label classification method named binary relevance so that the determined keywords can be grouped well and in a uniform fashion. The result of this research showed that a combination of two methods like BRkNN is able to predict up to 15 popular keywords than a single method like BR-OvA, MLkNN, or BR-SVM.*

Keywords — *Binary Relevance; Journal Keyword; k Nearest Neighbor.*

Abstrak — *Artikel ilmiah digunakan sebagai forum komunikasi bagi anggota masyarakat ilmiah bagi disiplin ilmu tertentu, karena artikel ilmiah dibaca, didiskusikan, dan disebarakan oleh anggota masyarakat ilmiah yang bersangkutan. Hal inilah yang membuat penentuan kata kunci (keyword) bisa dibidang menjadi bagian penting dari pembuatan artikel ilmiah. Keyword tidak hanya bertindak sebagai label suatu topik artikel ilmiah, melainkan juga sebagai sarana pencarian dan pengelompokan suatu topik. Penentuan keyword umumnya dilakukan secara pribadi oleh penulis berdasarkan judul penelitian atau dari tubuh artikel. Hal ini membuat keyword cenderung berbeda untuk setiap artikel ilmiah meski membahas topik yang sama. Untuk itu penelitian ini bertujuan untuk membuat aplikasi yang akan menentukan keyword secara otomatis berdasarkan metode klasifikasi Multi-label yakni Binary Relevance sehingga keyword yang didapat dapat lebih seragam dan dikelompokkan dengan baik. Hasil dari penelitian ini menunjukkan kombinasi dua metode seperti BRkNN memprediksikan sampai 15 keywords populer daripada metode tunggal seperti BR-OvA, MLkNN atau BR-SVM.*

Kata kunci — *Multi-Label; kata kunci jurnal; Binary Relevance, k Nearest Neighbor.*

I. PENDAHULUAN

Artikel ilmiah adalah sebuah publikasi yang diterbitkan secara berkala oleh suatu organisasi profesi atau institusi akademik yang memuat artikel-artikel yang merupakan produk

pemikiran ilmiah secara empiris maupun secara logis dalam bidang ilmu tertentu. Isi dari artikel ilmiah adalah artikel ilmiah (*research articles*) yakni tulisan yang berisi laporan sistematis mengenai hasil kajian atau hasil penelitian yang disajikan bagi audien tertentu, Yang dimaksud dengan audien tertentu antara lain seperti mahasiswa, dosen, peneliti dan ilmuwan [1].

Peran dan fungsi artikel ilmiah (1) sarana komunikasi akademik antara para ilmuwan(dosen/guru), (2) penyebaran (diseminasi) hasil-hasil penelitian, (3) pengembangan budaya akademik di perguruan tinggi, (4) sebagai penukaran informasi untuk menghasilkan ide-ide baru akan ilmu pengetahuan dan teknologi. Fungsi kata kunci pada artikel ilmiah digunakan untuk *filling and searching*, pengelompokan, dan dokumentasi.

Berdasarkan fungsi artikel ilmiah (1) dan (2), penentuan *keyword* menjadi bagian yang penting untuk artikel ilmiah agar artikel ilmiah mudah ditelusuri. Penentuan maupun penulisan *keywords* umumnya dilakukan secara manual oleh penulis dengan tata cara masing-masing, hal ini membuat beberapa *keywords* berbeda meskipun topik yang dibahas sama.

Dengan menggunakan metode *Multi-label Classification* yang adalah salah satu cabang dari ilmu *machine* penentuan *keyword* dapat dilakukan secara otomatis. *Binary Relevance* merupakan salah satu metode untuk pemodelan label-label terhadap satu instansi (*instance*), Teknik dan algoritma dalam pembentukan model ini sangat sesuai untuk penentuan *keywords* (label) dimana artikel ilmiah (*instance*) dapat memiliki lebih dari satu label.

Tujuan penelitian ini adalah untuk membuat aplikasi untuk menentukan *keywords* secara otomatis berdasarkan metode *binary relevance* (BR).

A. Penelitian Terkait

1) Penerapan Metode Adaboost untuk Multi-Label Classification Pada Dokumen Teks, oleh Arimbawa dan Sanjaya [2]. Penelitian ini menggunakan metode *AdaBoost* untuk mengklasifikasikan Teks yakni artikel computer berbahasa inggris, Proses penelitian dimulai dari pengumpulan data artikel, *text processing*, klasifikasi dan evaluasi. Berdasarkan penelitian yang dilakukan algoritma *AdaBoost* memberikan hasil lebih optimal dengan metode pembobotan fitur *TF-IDF* dibandingkan TF.

2) Kategorisasi Dokumen Teks Secara *Multi Label* Menggunakan *Fuzzi C-Means* Dan *K-Nearest Neighbors* Pada Artikel Berbahasa Indonesia, oleh Afrianto dan Kurniawati [3]. Penelitian ini menggunakan metode *Fuzzy C-Means* dan *K-Nearest Neighbors*. Uji coba dilakukan terhadap dokumen berita online sejumlah 175 dokumen yang terdiri atas tiga kategori label. Hasil uji coba menunjukkan bahwa metode yang diusulkan memberikan performa lebih baik dibanding metode lain. Hal ini ditunjukkan dengan nilai F1 sebesar 73,39% dan BEP sebesar 75,22%.

3) Multilabel Text Classification for Automated Tag Suggestion, oleh Katakis, Tsoumakas dan Vlahavas [4]. Penelitian dilakukan dengan menggunakan metode *binary relevance* dengan dataset untuk bookmark 208 *tags* dan 2150 *instance*, untuk bibtex 159 *tags* dan 1836 *instance*. Hasil penelitian menunjukkan nilai *f-measure* 0.0856, 0.0942, 0.0751 yang adalah nilai evaluasi terbaik yang bisa didapat.

B. Artikel ilmiah

Dalam publikasi [1] menunjukkan bahwa artikel ilmiah adalah tulisan yang berisi laporan sistematis mengenai hasil kajian atau hasil penelitian yang di sajikan bagi masyarakat ilmiah tertentu, yang merupakan audiens khusus dengan tujuan menyampaikan hasil kajian dan kontribusi penulis artikel kepada mereka untuk dipikirkan, dikaji kembali, dan diperdebatkan, baik secara lisan maupun secara tertulis.

C. Keywords

Kata kunci (*keywords*) adalah kata atau terminology spesifik bidang ilmu yang dibahas di dalam artikel. Kata kunci (*keywords*) menggambarkan ranah masalah yang diteliti dan istilah-istilah teknis yang berkaitan dengan penelitian yang dilaporkan. Kata kunci dapat diambil dari judul penelitian atau dari tubuh artikel (yang mencerminkan ranah permasalahan yang diteliti) sebanyak kurang lebih 3-5 kata. Fungsi kata kunci digunakan untuk *filling and searching*, pengelompokkan, dan dokumentasi [5].

D. PDF

Portable Document Format adalah format file yang dikembangkan oleh Adobe semenjak tahun 1991 sebagai dokumen digital yang mudah disebarluaskan terlepas dari berbagai *software*, *hardware*, atau *sistem operasi*. Selain teks dan gambar format PDF juga dapat memuat *links*, *buttons*, *form fields*.

E. Term Frequency-Inverse Document Frequency

Dalam klasifikasi teks, *words* atau kata-kata adalah hal krusial bagi penentuan *keyword* suatu dokumen. Umumnya penentuan *keyword* topik ditentukan dengan mencari kata spesial yang menjadi menggambarkan dokumen tersebut. Sebagai contoh, dokumen dengan topik gawai (*gadget*), tentu akan mengandung banyak kata-kata yang berhubungan dengan gawai, kata-kata seperti, “*smartphone*”, “*android*”, “*smartwatch*”, “*ipad*”, “*iphone*”, dan seterusnya. Kata-kata tersebut merupakan kata penting yang sering muncul sebagai acuan untuk mengetahui topik dokumen tersebut.

tf-idf merupakan *term frequency-inverse document frequency*, dimana nilai atau bobot tf-idf dihitung dengan frekuensi kemunculan suatu kata pada suatu dokumen dan frekuensi kata yang jarang muncul pada keseluruhan dokumen. Nilai yang dihasilkan tf-idf dianggap mampu memfilter kata umum selain *stopwords*, serta memberikan bobot nilai tinggi pada kata penting.

$$tf\ idf(t, d) = tf(t, d) \times \log(N/(df + 1)) \quad (1)$$

Dimana tf adalah *term frequency* berapa banyak setiap kata-kata untuk muncul pada suatu dokumen, df adalah *document frequency* dimana kemunculan term pada seluruh dokumen. Df kemudian dibalik (*inverse*) menggunakan skala logaritmik pembagian N sebagai total seluruh dokumen, dengan df+1 untuk menghindari pembagian 0. Bobot tf-idf dapat dihitung sesuai dengan rumus 1 [6].

F. Multi-Label Classification

Dalam kasus klasifikasi tradisional (*Multi-Class Classification*), *classes* bersifat *mutually exclusive* atau hanya tertera satu label untuk satu *instance*. Namun dalam beberapa kasus klasifikasi, dua atau lebih *classes* dapat dikategorikan kedalam satu *instance*. Kasus-kasus tersebut dapat dikategorikan kedalam beberapa kategori (*multiple label*), yang biasa disebut *Multi-Label*.

Multi-label Classification dapat dikategorikan kedalam dua kategori, yaitu *Problem Transformation* dan *Algorithm adaptation*.

ML *Problem Transformation* merupakan metode untuk menyesuaikan data terhadap algoritma yang sudah ada, sedangkan ML *algorithm adaptation* merupakan metode untuk menyesuaikan algoritma yang sudah ada terhadap data [7].

G. Binary Relevance

Binary Relevance (BR) adalah algoritma yang memecah masalah klasifikasi menjadi masalah independen secara biner, dimana setiap masalah akan diberikan label yang memungkinkan (*possible label*). Metode BR termasuk dalam kelompok algoritma *problem transformation* pada permasalahan *multi-label* dimana permasalahan *multi-class (multi-class problem)* diubah menjadi masalah tunggal (*single*) masing-masing label.

$$\mathcal{X} = \mathbb{R}^d \quad \mathcal{Y} = \{\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_q\} \quad (2)$$

$$\mathcal{D}_j = \{(\mathbf{x}^i, \mathbf{y}_j^i) \mid 1 \leq i \leq m\} \quad (3)$$

Pada rumus 2 \mathcal{X} merupakan dimensi *instance* yang adalah *data train*. \mathcal{Y} merupakan *label space* masing-masing *unique label* λ sampai q .

Pada rumus 3 \mathcal{D}_j merupakan *multi-label training set* dimana \mathbf{x}^i merupakan elemen daripada \mathcal{X} yang akan terlibat, \mathbf{y}^i merupakan elemen daripada label yang bernilai *binary*, \mathbf{y}_j^i merupakan indikasi apakah label tersebut relevan atau tidak untuk setiap *instance* \mathbf{x}^i .

Binary classifier (β) dapat diinduksi dengan *multi-label training set* menjadi $g_j \leftarrow \beta(\mathcal{D}_j)$, sehingga data train $(\mathbf{x}^i, \mathbf{y}^i)$

dapat berkontribusi untuk seluruh proses klasifikasi biner $g_j(1 \leq j \leq q)$.

$$Y^* = \{\lambda_j | g_j(x^*) > 0, 1 \leq j \leq q\} \quad (4)$$

Untuk sebuah *unseen instance* x^* , label relevan Y^* ditentukan dengan menghitung output untuk setiap hasil *binary classifier* g_j pada rumus 4.

Tabel 1 Pseudocode *Binary Relevance*

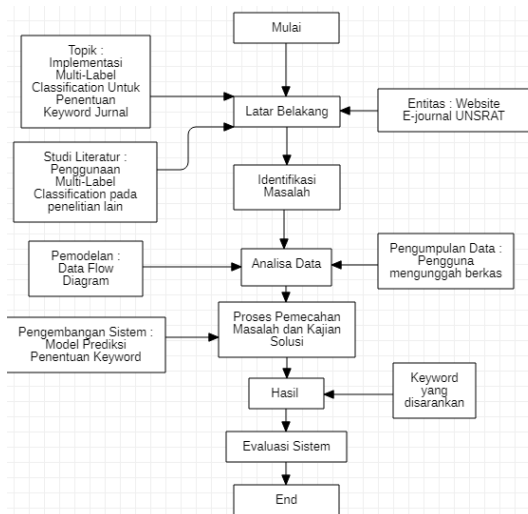
Inputs:
D : Multi-label training set $\{(x^i, y^i) \mid 1 \leq i \leq m\}$
$(x^i \in \mathcal{X}, y^i \in \{-1, +1\}^q, \mathcal{X} = \mathbb{R}^d, Y = \{\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_q\})$
β : algoritma <i>binary learning</i>
x^* : <i>Unseen instance</i> ($x^* \in \mathcal{X}$)
Output:
Y^* : Label terprediksi untuk x^*
Proses:
1: for $j = 1$ to q do
2: Bangun <i>binary training set</i> D_j sesuai rumus 4;
3: Induksi <i>binary classifier</i> $g_j \leftarrow \beta(D_j)$;
4: end for
5: return $Y^* = \{\lambda_j g_j(x^*) > 0, 1 \leq j \leq q\}$

Pada tabel 6 pseudocode *Binary Relevance*. Untuk setiap j sampai q bangun *binary training set* sesuai dengan rumus 4 dimana setiap *unique keyword* dilatih secara independen, lalu rumus 4 diinduksi klasifikasi biner. Proses *for* akan mengembalikan Y^* yang adalah set dari lambda label (λ) dimana *unseen instance* x^* (jurnal yang akan diprediksi) akan melalui proses *for* [8].

II. METODE

A. Kerangka Pikir

Kerangka Pikir merupakan rangkaian bagan yang menggambarkan alur dari proses kerja dalam penelitian penentuan *keyword* jurnal.

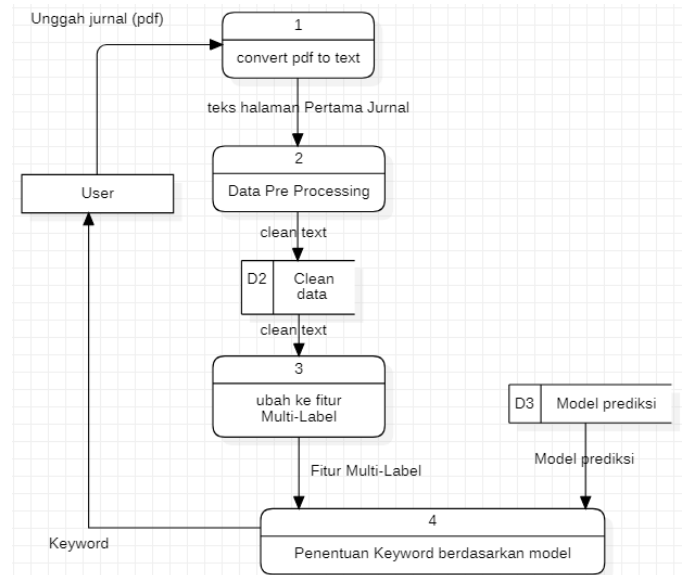


Gambar 1 Kerangka Pikir

Proses penentuan *keyword* dimulai dengan pengguna mengunggah data yang berupa jurnal dengan format *.pdf*, data tersebut hanya akan diambil teks pada halaman pertama yang memuat judul dan abstrak jurnal tersebut, setelah itu dilanjutkan dengan analisa data yang berupa *pre-processing* serta pengembangan sistem model prediksi, kemudian data yang telah melalui *pre-processing* akan ditentukan *keyword* berdasarkan model prediksi dengan hasil akhir yaitu *keyword* yang disarankan.

B. Data Flow Diagram

Data Flow Diagram adalah suatu diagram yang menggambarkan aliran data dari sebuah proses atau sistem. DFD juga menggambarkan informasi mengenai *input* dan *output* diantara proses yang terjadi. Untuk menggambarkan sistem penentuan *keyword* ini akan digunakan DFD Notasi *Gane and Sarson*



Gambar 2 Data Flow Diagram

Proses pertama adalah *convert pdf to text* dimana setelah data yang diunggah *user* berupa pdf akan di-*convert* atau diubah menjadi teks dan hanya diambil halaman pertama yang terdapat judul dan abstrak dari artikel ilmiah tersebut, hasilnya adalah teks halaman pertama artikel ilmiah yang siap dikelola proses kedua.

Proses kedua adalah data *pre-processing* dimana teks halaman pertama akan dibersihkan dengan menghapus kata yang kurang berarti atau *stopword* beserta *cleaning* untuk menghilangkan tanda baca dan karakter yang tidak diperlukan, data yang dihasilkan dari proses ini disimpan sementara pada D2 untuk dapat dipakai ke proses ketiga.

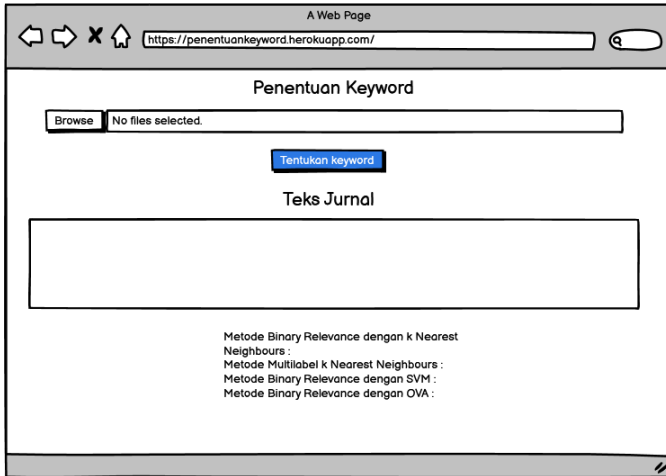
Proses ketiga adalah ubah D2 ke fitur *multi-label*, dimana data teks bersih dari D2 dapat diubah ke fitur *tf-idf* pada halaman pertama artikel ilmiah, hasil dari proses ini akan dipakai ke proses keempat.

Proses keempat adalah proses penentuan *keyword* berdasarkan model prediksi D3 yang telah dibuat terlebih dahulu. Proses ini merupakan proses terakhir dimana model prediksi D3 dapat langsung digunakan sebagai sarana

penentuan keyword bersama dengan fitur *tf-idf* dari proses ketiga. Proses ini menghasilkan *keyword* yang akan disarankan kepada *user*.

C. Rancangan Antar Muka

Rancangan antar muka merupakan desain antarmuka yang akan digunakan dalam pengimplementasian dari sistem yang telah dibuat, Perancangan antar muka aplikasi penentuan keyword artikel ilmiah dengan metode Binary Relevance ini berbasis web dengan menggunakan python sebagai web application.

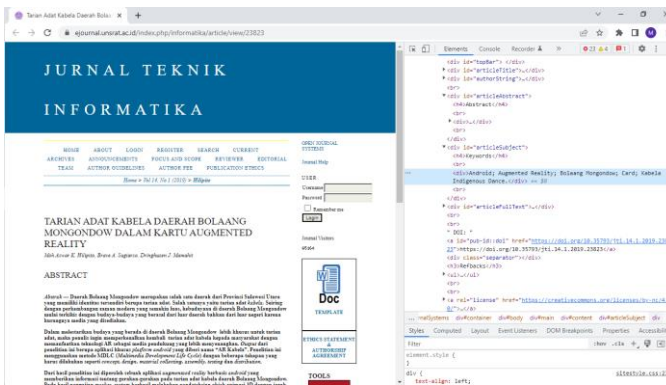


Gambar 3 Rancangan Antar Muka

Gambar 3 merupakan rancangan antar muka aplikasi penentuan *keyword*. *User* dapat mengunggah file artikel ilmiah berformat pdf lalu menekan tombol tentukan keyword untuk menentukan keyword. Keyword akan tertera pada bagian text area bagian bawah sesuai dengan metode yang digunakan.

III. HASIL DAN PEMBAHASAN

A. Implementasi Pengambilan Data Artikel Ilmiah



Gambar 4 Halaman Website e-journal Unsrat

Penelitian dimulai dengan mengambil data artikel ilmiah yang akan digunakan sebagai model prediksi penentuan *keywords*, data artikel ilmiah diambil pada website e-journal Universitas Sam Ratulangi Manado Sulawesi Utara Indonesia [9].

```
9 print("Total Artikel Ilmiah :",len(key_new['Judul']))
```

Total Artikel Ilmiah : 221

```
1 print("Total Keyword :",len(set(all_key)))
```

Total Keyword : 623

Gambar 4 Total Artikel Ilmiah dan *Keyword*

Gambar 4 merupakan hasil pengambilan data artikel ilmiah pada *website* e-journal Universitas Sam Ratulangi yang memuat 221 artikel ilmiah dengan 623 *unique keyword*.

B. Implementasi Pre-Processing

Pre-processing dimulai dengan membersihkan data teks abstrak yang telah diambil dari *website* e-journal Universitas Sam Ratulangi.

Pre-processing meliputi. Pembersihan simbol-simbol, mengubah huruf menjadi huruf kecil. Menghilangkan kata *stopword* atau kata yang sering muncul seperti: "atau", "dan", "di", "jika", dan seterusnya. Serta mengembalikan kata berimbuhan menjadi kata dasar seperti kata "Penggambaran" menjadi "gambar", atau yang biasa disebut *stemming* [10].

Tabel 2 Perbandingan Abstrak Hasil *Pre-Processing*

Abstrak Sebelum <i>Pre-Processing</i>
\\nabstract\n\nsejak akhir tahun 2019, dunia menghadapi ancaman dari sebuah virus baru yang dikenal sebagai novel corona virus, yang kemudian dinamai ulang sebagai sars-cov-2. karena sifat penularannya, maka pembatasan sosial diambil sebagai langkah untuk memperlambat penyebaran virus tersebut. segala sesuatu harus dilakukan dari jarak jauh, tanpa kontak fisik secara langsung, kecuali untuk kondisi-kondisi tertentu. oleh karenanya, berbagai hal harus dilakukan dari rumah, termasuk belajar dan mengajar. makalah ini membahas tentang penggunaan platform pemrograman daring dalam mendukung proses belajar-mengajar pada mata kuliah probabilitas dan statistika, yang diselenggarakan pada program s1 teknik informatika, universitas sam ratulangi, manado.'
Abstrak Sesudah <i>Pre-Processing</i>
'sejak akhir tahun dunia hp ancam buah virus baru kenal novel corona virus kemudian nama ulang sars cov sifat tular batas sosial ambil langkah lambat sebar virus sebut segala harus laku jarak jauh kontak fisik langsung untuk kondisi kondisi tentu karena bagai harus laku rumah masuk ajar ajar makalah bahas platform pemrograman daring dukung proses ajar ajar mata kuliah probabilitas statistika selenggara program teknik informatika universitas sam ratulangi manado'

Tabel 2 merupakan contoh perbandingan antara abstrak artikel ilmiah sebelum dan sesudah melalui *pre-processing*.

File artikel ilmiah unggahan *user* yang telah di ubah kedalam bentuk teks juga akan melalui *pre-processing*.

C. Implementasi Ekstraksi Fitur Data

Implementasi ekstraksi fitur data *tf-idf* dihitung sesuai dengan rumus 1 menggunakan *library* python *tfidfvectorizer* pada *sklearn* [11].

	ajar	anak	android	aplikasi	application	banyak	bas	based	buat	dapat	data
0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
1	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.138704	0.185129	0.262856	0.000000
2	0.000000	0.000000	0.000000	0.000000	0.286335	0.000000	0.077219	0.083848	0.000000	0.000000	0.000000
3	0.178155	0.058864	0.000000	0.117369	0.071408	0.081533	0.000000	0.000000	0.167457	0.052836	0.000000
4	0.395515	0.000000	0.000000	0.195425	0.158531	0.090505	0.000000	0.000000	0.000000	0.000000	0.000000
5	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.129959	0.000000	0.000000	0.000000	0.089142
6	0.101325	0.000000	0.104275	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.060101	0.000000

Gambar 6 Fitur data *tf-idf*

Gambar 6 menunjukkan bobot nilai *tf-idf* pada data abstrak

pada artikel ilmiah sebanyak 50 fitur *tf-idf* yang akan dipakai dalam pembuatan model prediksi.

Unggahan *user* yang berupa teks halaman pertama dan telah melalui *pre-processing* juga akan melalui proses ekstraksi fitur data untuk mengetahui bobot nilai *tf-idf* yang akan digunakan untuk memprediksi *keyword*.

D. Implementasi Pembuatan Model Prediksi

Pembuatan model prediksi berdasarkan hasil ekstraksi fitur data artikel ilmiah. metode utama yakni Binary Relevance menggunakan *library* python yaitu sklearn [11] dan skmultilearn [12]. Model prediksi dibuat berdasarkan empat metode yakni: Binary Relevance - One vs All (BR-OvA) [11], Binary Relevance - Support Vector Machine (BR-SVM) [12], Multi-Label k-Nearest Neighbor (MLkNN) [13], Binary Relevance - k Nearest Neighbor (BRkNN) [14].

BR-OvA dan BR-SVM merupakan kelompok algoritma *multi-label problem transformation*, sedangkan MLkNN merupakan kelompok algoritma *multi-label algorithm adaptation* yang digunakan sebagai metode pembandingan untuk mengetahui kinerja algoritma kNN secara pribadi [7].

Untuk metode BR-kNN merupakan *pairing* antara metode Binary Relevance dan metode k Nearest Neighbor, dimana penggunaan algoritma kNN diperluas agar setiap label mampu memprediksi secara independen, selayaknya metode utama Binary Relevance [14].

Model prediksi yang telah dibuat kemudian disimpan dalam bentuk file untuk dapat langsung digunakan pada aplikasi.

E. Implementasi Pembuatan Aplikasi

Pembuatan aplikasi menggunakan *library* python yaitu flask. Flask berfungsi sebagai WSGI atau *Web Server Gateway Interface*, dimana *web server* terhubung dengan aplikasi python.

Untuk unggahan *user* akan menggunakan *library* PyPDF2 yang berfungsi untuk memproses file berformat pdf, serta mengubah halaman pertama menjadi teks (*convert*).

Hasil *convert* tersebut kemudian akan melalui *pre-processing* serta ekstraksi fitur *tf-idf* yang akan digunakan untuk menentukan *keyword*.

pastikan abstrak artikel ilmiah terdapat pada halaman pertama file pdf sebelum mengunggah.

F. Hasil dan Evaluasi sistem

Hasil prediksi dari keempat model prediksi berdasarkan *data test* hanya dapat memprediksi 15 *keyword*. *Keyword-keyword* ini juga adalah *keyword* yang sering muncul dalam dataset 221 artikel ilmiah.

Tabel 3 Hasil model prediksi

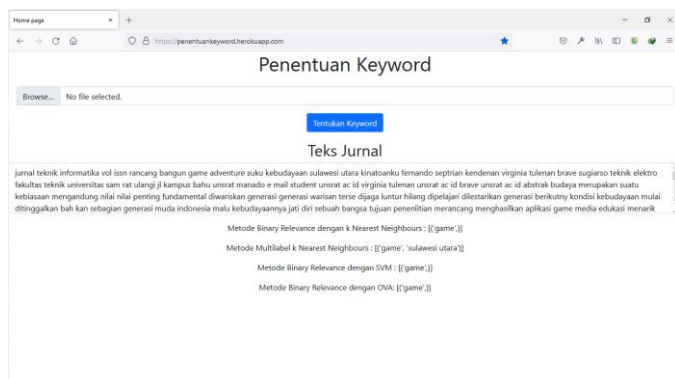
No	BR-OvA	BR-SVM	BRkNN	MLkNN
1	(game,)	(game,)	(game,)	(game,)
2	()	(multimedia development life cycle)	(augmented reality, multimedia development life cycle,)	(augmented reality, multimedia development life cycle,)
3	()	()	(rapid application development,)	(rapid application development,)
4	()	()	(android,)	(android,)
5	()	()	(data mining,)	()
6	()	()	(aplikasi,)	(aplikasi,)
7	()	()	(game,)	(game,)
8	()	()	(android, rapid application development)	(android, rapid application development)
9	()	()	(manado,)	()
10	()	()	(php, sistem informasi)	()
11	()	(game,)	(entrepreneurship, game)	(entrepreneurship, game)
12	()	()	(blender, multimedia development life cycle)	(multimedia development life cycle,)
13	()	()	(addie, e-learning)	(addie, e-learning)
14	()	(multimedia development life cycle,)	(multimedia development life cycle,)	(multimedia development life cycle,)
15	()	()	(sistem informasi,)	0
16	()	()	(android, aplikasi, rapid application development)	(android, aplikasi, rapid application development)

Tabel 3 menunjukkan hasil metode BR-OvA mampu menentukan 1 *keyword*, metode BR-SVM mampu menentukan 2 *keyword*, metode BRkNN mampu menentukan 15 *keyword*, serta metode MLkNN mampu menentukan 9 *keyword*.

Menurut Boutell [7] proses evaluasi sistem pada *multi-label classification* sedikit berbeda dengan proses evaluasi *single-label classification* atau *multi-class classification*, yang menggunakan cara evaluasi umum seperti *precision*, *recall*, *accuracy*, dan *F-measure* [15]. Dalam kasus *multi-label*, evaluasi sistem lebih rumit dikarenakan hasil prediksi pada satu *instance* dapat berupa (Asumsi Label True: C1, C2):

1. C1, C2 (benar 2 label),
2. C1 (benar 1 label, 1 label tidak terprediksi),
3. C1, C3 (benar 1 label, dengan 1 label salah prediksi),
4. C1, C3 C4 (benar 1 label, dengan 2 label salah prediksi),
5. C3, C4 (salah prediksi semua label)

Kelima hasil di atas menunjukkan hasil prediksi dengan derajat kebenaran (*degree of correctness*) yang berbeda



Gambar 7 Aplikasi penentuan keyword

Gambar 7 merupakan aplikasi penentuan *keyword* yang telah di-deploy secara daring melalui Heroku dengan nama website <https://penentuankeyword.herokuapp.com/>. User dapat langsung mengunggah artikel ilmiah yang berformat pdf, dan

Boutell [7] mengajukan α -evaluation sebagai cara untuk mengevaluasi *multi-label* dengan derajat kesalahan berbeda, dengan formula sebagai berikut:

$$\alpha \text{ evaluation score} = 1 - \left(\frac{|\beta M_x + \gamma F_x|}{|Y_x \vee P_x|} \right)^\alpha \quad (5)$$

$$(\alpha \geq 0, 0 \leq \beta, \gamma \leq 1, \beta = 1/4, \gamma = 1)$$

M_x merupakan total *false negatives* yakni *keyword* yang sebenarnya bernilai positif teridentifikasi sebagai positif pada model prediksi, F_x merupakan total *false positives* yakni *keyword* sebenarnya bernilai negatif teridentifikasi sebagai positif pada model prediksi, Y_x merupakan jumlah *true positives* dan *false negatives* yakni *Keyword* sebenarnya bernilai positif teridentifikasi sebagai negatif pada model prediksi, dan P_x merupakan jumlah *true positive* dan *false positives* pada model prediksi.

Nilai β dan γ merupakan konstanta agar skor tidak bernilai negatif, sedangkan nilai α merupakan nilai *forgiveness rate* dimana semakin kecil nilai α , semakin dapat mentolerir hasil prediksi dengan berbagai derajat kesalahan yang ada. Hasil evaluasi skor alpha dapat dilihat pada tabel 4

Tabel 4 hasil evaluasi skor alpha

Jenis Metode	Hasil Evaluasi skor alpha ($\alpha = 1, \beta = 0.25, \gamma = 1$)
BR - OvA	0.75
BR - SVM	0.75
BRkNN	0.70
MLkNN	0.73

Menurut Boutell cara evaluasi alpha adalah versi umum daripada cara evaluasi jaccard similarity. Serta cara evaluasi skor alpha dianggap lebih mampu untuk mentolerir kasus *multi-label* dengan derajat kesalahan berbeda dibandingkan cara evaluasi seperti *accuracy* yang sangat ketat dalam menilai hasil prediksi dengan derajat kesalahan yang beragam [7].

Cara evaluasi lainnya seperti pada publikasi [16] dimana untuk memaksimalkan nilai f1-score (f-measure) dapat dilakukan dengan *pairing* metode LRM (Logistic Regresion Model) dan SVM (Support Vector Machine). f-measure merupakan salah satu cara evaluasi klasifikasi biner yang mengukur tingkat keakurasian data.

Dalam publikasi [4] yang juga penelitian terkait bagi penelitian ini menunjukkan nilai f-measure untuk kasus pencarian *tag/keyword* pada *bibtext* dengan dataset meliputi 159 *tags* dan 1836 *instance* sangat bergantung pada *popular tag* atau *keyword* yang sering digunakan. Nilai f-measure paling tinggi didapat apabila model prediksi hanya akan memprediksi 3 *keyword* paling populer dengan nilai f-measure 0.0740 untuk pencarian *keyword* pada *bibtext*.

f-measure Binary Relevance dengan OvA : 0.01746031746031746
 f-measure Binary Relevance dengan SVM : 0.0407936507936508
 f-measure Binary Relevance k Nearest Neighbor: 0.13451499118165786
 f-measure Multi-label k Nearest Neighbor : 0.12922398589065257

Gambar 8 Evaluasi f-measure

Gambar 8 merupakan perbandingan nilai f-measure antar metode-metode, dapat dilihat kombinasi metode (*pairing*) BRkNN yang mampu memprediksi 15 *keyword* populer

mempunyai nilai f-measure paling tinggi yaitu 0.134 daripada metode BR-OvA, MLkNN, dan BR-SVM.

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Setelah melakukan pengumpulan data hingga menyelesaikan semua proses dalam penelitian ini maka dapat menarik kesimpulan bahwa pengambilan data artikel ilmiah pada website e-artikel ilmiah UNSRAT yang meliputi 221 artikel ilmiah dan 623 *unique keywords* dianggap cukup mumpuni untuk membuat model prediksi yang mampu menentukan 15 *unique keyword* yang sering digunakan pada data training (*popular*). Dan dengan menggunakan kombinasi (*pairing*) dua metode seperti BRkNN, penentuan *keyword* atau kata kunci akan memiliki hasil prediksi *keyword* lebih banyak daripada metode tunggal seperti BR-OvA, BR-SVM, atau MLkNN.

B. Saran

Dalam penelitian dari hasil pengumpulan, labeling, training, dan testing yang telah dilakukan masih ada hal – hal yang perlu dikaji agar dapat menjadi lebih baik. Penelitian selanjutnya dapat menggunakan kasus serta metode yang berbeda dalam pengembangan penelitian yang berkaitan dengan klasifikasi *multi-label*. *Evaluation metric* atau cara evaluasi bagi kasus *multi-label* masih memerlukan penelitian lebih lanjut untuk dapat menilai bobot derajat kesalahan yang lebih akurat.

V. KUTIPAN

- [1] Z. Adnan and I. Zifirdaus, *Merebut Hati Audiens International*, Jakarta: Gramedia, 2005.
- [2] I. G. A. P. Arimbawa and N. A. Sanjaya, "Penerapan Metode Adaboost Untuk Multi-Label Classification Pada Dokumen Teks," *Jurnal Elektronik Ilmu Komputer Udayana*, 2020.
- [3] R. B. Afrianto and L. Y. Kurniawati, "KATEGORISASI DOKUMEN TEKS SECARA MULTI LABEL MENGGUNAKAN FUZZY C-MEANS DAN K-NEAREST NEIGHBORS PADA ARTIKEL BERBAHASA INDONESIA," 2013.
- [4] I. Katakis, G. Tsoumakas and I. Vlahavas, "Multilabel Text Classification for Automated Tag Suggestion," in *ECML/PKDD*, Antwerp, 2008.
- [5] G. Suryoputro, S. Riadi and A. Sya'ban, *Menulis Artikel Untuk Jurnal Ilmiah*, Jakarta: Uhamka Press, 2012.
- [6] A. Rajaraman, J. D. Ullman and J. Leskovec, *Mining of Massive Datasets*, 2011.
- [7] M. R. Boutell, J. Luo, X. Shen and C. M. Brown, "Learning multi-label scene classification," *Pattern Recognition*, pp. 1757-1771, 2004.
- [8] M. L. Zhang, Y. K. Li, X. Y. Liu and X. Geng, "Binary relevance for multi-label learning: an overview," pp. 191-202, 2018.

- [9] "Archives," [Online]. Available: <https://ejournal.unsrat.ac.id/index.php/informatika/index>.
- [10] Z. F. Tala, "A Study of Stemming Effects on Information," 2003.
- [11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.
- [12] P. Szymanski and T. Kajdanowicz, "A scikit-based Python environment for performing multi-label classification," *ArXiv e-prints*, 2017.
- [13] M. L. Zhang and Z. H. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," *Pattern recognition*, pp. 2038-2048, 2007.
- [14] E. Spymitros, G. Tsoumakas and I. Vlahavas, "An Empirical Study of Lazy Multilabel Classification Algorithms," in *5th Hellenic Conference on Artificial Intelligence (SETN 2008)*, Syros, 2008.
- [15] F. Sebastiani, "Machine Learning in Automated Text Categorization," *ACM Computing Surveys*, 2001.
- [16] A. Fujino, H. Isozaki and J. Suzuki, "Multi-label Text Categorization with Model Combination based on F1-score maximization," 2008.



Marvel Manueke. Lahir di Surabaya 19 Agustus 1997. Penulis merupakan anak ke-2 dari 3 orang bersaudara, dan ke-2 orang tua penulis adalah Vicky Tewu Manueke dan Melly Bawango. Penulis mulai menempuh pendidikan di Sekolah Dasar Kristen YBPK-1 Surabaya (2003–2009). Lalu penulis melanjutkan ke Sekolah Menengah Pertama YBPK-1 Surabaya (2009-2012). Dan melanjutkan ke Sekolah Menengah ke Atas YBPK-1 Surabaya (2012-2015). Pada tahun 2015 penulis melanjutkan pendidikan di salah satu perguruan tinggi negeri yang ada di Manado yaitu Universitas Sam Ratulangi Manado, dengan mengambil Program Studi S-1 Teknik Informatika di Fakultas Teknik. Dan penulis mengajukan proposal Skripsi untuk memenuhi syarat dalam meraih gelar sarjana (S1) dengan judul Implementasi Multi-Label untuk Penentuan Keyword Artikel Ilmiah pada Jurnal, skripsi ini dibimbing oleh dua dosen pembimbing, yaitu Rizal Sengkey, ST., MT, dan Agustinus Jacobus, ST, M.Cs.