KLASIFIKASI BIDANG FOKUS PUBLIKASI JURNAL PENELITIAN YANG TERINDEKS PADA PORTAL GARUDA DENGAN METODE *MULTICLASS SUPPORT VECTOR MACHINES (SVM)*

LAPORAN TUGAS AKHIR

Laporan ini disusun guna memenuhi salah satu syarat untuk menyelesaikan program studi Teknik Informatika S-1 pada Fakultas Teknologi Industri Universitas Islam Sultan Agung

DISUSUN OLEH :

DANI NASRULLAH

NIM 32601601038

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS ISLAM SULTAN AGUNG

SEMARANG

2023

CLASSIFICATION OF RESEARCH JOURNAL PUBLICATION FOCUS CLASSIFICATION ON THE GARUDA PORTAL USING THE MULTICLASS SUPPORT VECTOR MACHINES (SVM) METHOD

Proposed to complete the requirement to obtain a bachelor's degree (S-1) at Informatics Engineering Departement of Industrial Technology Faculty Sultan Agung Islamic University

Arranged By :

DANI NASRULLAH

32601601038

MAJORING OF INFORMATICS ENGINEERING

INDUSTRIAL TECHNOLOGY FACULTY

SULTAN AGUNG ISLAMIC UNIVERSITY

SEMARANG

2023

LEMBAR PENGESAHAN PEMBIMBING

Laporan Tugas Akhir dengan judul "Klasifikasi Bidang Fokus Publikasi Jurnal Penelitian yang Terindeks pada Portal Garuda dengan Metode Multiclass Support Vector Machines (SVM)" ini disusun oleh :

Nama : Dani Nasrullah

NIM : 32601601038

Program Studi : Teknik Informatika

Telah disahkan oleh dosen pembimbing pada :

Hari : .....

Tanggal : .....

Mengesahkan,

Pembimbing I	Pembimbing II
Imam Much Ibnu Subroto, S.T, M.Sc, Ph.D	Sam Farisa C, S.T, M.Kom
NIDN. 0613037301	NIDN. 0628028602

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Teknologi Industri
Universitas Islam Sultan Agung

Ir. Sri Mulyono, M.Eng
NIDN.0626066601

LEMBAR PENGESAHAN PENGUJI

Laporan tugas akhir dengan judul "Klasifikasi Bidang Fokus Publikasi Jurnal Penelitian yang Terindeks pada Portal Garuda dengan Metode Multiclass Support Vector Machines (SVM)" ini telah dipertahankan di depan dosen penguji Tugas Akhir pada :

Hari : .....

Tanggal : .....

TIM PENGUJI

Ketua Penguji	**Anggota I**
Andi Riasyah, S.T, M.Kom	Dedy Kurniadi, ST, S.T, M.Kom
NIDN.0609108802	NIDN.0622058802

SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini :

Nama	: Dani Nasrullah
NIM	: 32601601038
Judul Tugas Akhir	: Klasifikasi Bidang Fokus Publikasi Jurnal Penelitian yang Terindeks pada Portal Garuda dengan Metode Multiclass Support Vector Machines (SVM)

Dengan bahwa ini saya menyatakan bahwa judul dan isi Tugas Akhir yang saya buat dalam rangka menyelesaikan Pendidikan Strata Satu (S1) Teknik Informatika tersebut adalah asli dan belum pernah diangkat, ditulis ataupun dipublikasikan oleh siapapun baik keseluruhan maupun sebagian, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka, dan apbila di kemudian hari ternyata terbukti bahwa judul Tugas Akhir tersebut pernah diangkat, ditulis ataupun dipublikasikan, maka saya bersedia dikenakan sanksi akademis. Demikian surat pernyataan ini saya buat dengan sadar dan penuh tanggung jawab.

Semarang,13 Maret 2023

Yang Menyatakan,

Dani Nasrullah

PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

Saya yang bertanda tangan dibawah ini :

Nama	: Dani Nasrullah
NIM	: 32601601038
Program Studi	: Teknik Informatika
Fakultas	: Teknologi industri
Alamat Asal	: Klasifikasi Bidang Fokus Publikasi Jurnal Penelitian yang Terindeks pada Portal Garuda dengan Metode Multiclass Support Vector Machines (SVM)

Dengan ini menyatakan Karya Ilmiah berupa Tugas akhir dengan Judul : Klasifikasi Bidang Fokus Publikasi Jurnal Penelitian yang Terindeks pada Portal Garuda dengan Metode Multiclass Support Vector Machines (SVM) Menyetujui menjadi hak milik Universitas Islam Sultan Agung serta memberikan Hak bebas Royalti Non-Eksklusif untuk disimpan, dialihmediakan, dikelola dan pangkalan data dan dipublikasikan diinternet dan media lain untuk kepentingan akademis selama tetap menyantumkan nama penulis sebagai pemilik hak cipta. Pernyataan ini saya buat dengan sungguh-sungguh. Apabila dikemudian hari terbukti ada pelanggaran Hak Cipta/Plagiarisme dalam karya ilmiah ini, maka segala bentuk tuntutan hukum yang timbul akan saya tanggung secara pribadi tanpa melibatkan Universitas Islam Sultan agung.

Semarang, 13 Maret 2023

Yang menyatakan,

Dani Nasrullah

KATA PENGENTAR

Dengan mengucap syukur alhamdulillah atas kehadirat Allah SWT yang telah memberikan rahmat dan karunianya kepada penulis, sehingga dapat menyelesaikan Tugas Akhir dengan judul "Klasifikasi Bidang Fokus Publikasi Jurnal Penelitian yang Terindeks pada Portal Garuda dengan Metode Multiclass Support Vector Machines (SVM)" ini untuk memenuhi salah satu syarat menyelesaikan studi serta dalam rangka memperoleh gelar sarjana (S-1) pada Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Islam Sultan Agung Semarang. Tugas Akhir ini disusun dan dibuat dengan adanya bantuan dari berbagai pihak, materi maupun teknis, oleh karena itu saya selaku penulis mengucapkan terima kasih kepada :

1. Rektor UNISSULA Bapak Prof. Dr. H. Gunarto, SH., M.Hum yang mengijinkan penulis menimba ilmu di kampus ini.
2. Dekan Fakultas Teknologi Industri Ibu Dr. Novi Marlyana, ST., MT.
3. Dosen pembimbing I penulis Imam Much Ibnu Subroto, ST., M.Sc., Ph.D yang telah meluangkan waktu dan memberi ilmu.
4. Dosen pembimbing II penulis Sam Farisa C, S.T, M.Kom yang telah memberikan banyak nasehat dan saran.
5. Orang tua penulis yang telah mengijinkan untuk menyelesaikan laporan ini.
6. Dan kepada semua pihak yang tidak dapat saya satu persatu.

Dengan segala kerendahan hati, penulis menyadari masih banyak terdapat banyak kekurangan-kekurangan dari segi kualitas atau kuantitas maupun dari ilmu pengetahuan dalam penyusunan laporan, sehingga penulis mengharapkan adanya saran dan kritikan yang bersifat membangun demi kesempurnaan laporan ini.

Semarang, 13 Maret 2023

Dani Nasrullah

DAFTAR ISI

HALAMAN JUDUL ..... **i**
LEMBAR PENGESAHAN PEMBIMBING ..... **i**Error! Bookmark not defined.
LEMBAR PENGESAHAN PENGUJI ..... **iv**
SURAT PERNYATAAN KEASLIAN TUGAS AKHIR ..... **v**
PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH ..... **vi**
KATA PENGENTAR ..... **vii**
DAFTAR ISI ..... **viii**
DAFTAR GAMBAR ..... **x**
DAFTAR TABEL ..... **xi**
ABSTRAK ..... **xiii**
BAB I PENDAHULUAN **1**
1.1 Latar Belakang 1
1.2 Perumusan Masalah 3
1.3 Pembatasan Masalah 3
1.4 Tujuan 4
1.5 Sistematika Penulisan 4
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI **6**
2.1 Penelitian Terdahulu 6
2.2 Dasar Teori 9
2.2.1 *Text Mining* 9
2.2.2 *Data Preprocessing* 12
2.2.3 Pembobotan *Term* (TF IDF) 16
2.2.4 *Support Vector Machine* (SVM) 18
BAB III METODE PENELITIAN **23**
3.1 Studi Literasi 23
3.2 Pengumpulan Data 23
3.3 *Preprocessing Data* 24
3.4 Pembobotan Kata TF IDF 25

3.5 Klasifikasi dengan Metode SVM 26
3.6 Perancangan Sistem 29
3.7 Pengujian 29
BAB IV HASIL DAN ANALISIS PENELITIAN 31
4.1 Pengambilan Data 31
4.2 *Preprocessing Data* 32
4.2.1 *Case Folding* 33
4.2.2 *Text Cleaning* 34
4.2.3 *Tokenizing* 34
4.2.4 *Stemming Word* 36
4.2.5 *Stopword Remover* 38
4.3 Pembobotan Kata dengan TF IDF 39
4.4 Klasifikasi Metode SVM 44
4.4.1 Perancangan Model Klasifikasi *Multiclass* SVM metode OVA 45
4.4.2 Pengujian dan Evaluasi Model 49
4.4.3 Hasil Pengujian Klasifikasi *Multiclass* SVM metode OVA 56
4.4.4 Hasil Perbandingan Akurasi dari Setiap Kelas 57
4.4.5 Analisis Prediksi Hasil Klasifikasi 58
BAB V KESIMPULAN DAN SARAN 63
5.1 Kesimpulan 63
5.2 Saran 63

DAFTAR GAMBAR

Gambar 2. 1 Arsitektur Text Mining 11
Gambar 2. 2 Proses Text Mining 11
Gambar 2. 3 Tahapan Preprocessing Data 13
Gambar 2. 4 Flowchart Preprocessing text 15
Gambar 2. 5 Flowchart TF IDF 17
Gambar 2. 6 Visualisasi Metode SVM 19
Gambar 2. 7 Visualiasi margin kelas 20
Gambar 2. 8 SVM multiclass OVA 21
Gambar 2. 9 Flowchart SVM OVA 22
Gambar 3. 1 Diagram proses Penelitian 23
Gambar 3. 2 Flowchart SVM OVA 26
Gambar 3. 3 Proses Training dataset SVM Multiclass 27
Gambar 4. 1 Proporsi dataset 31
Gambar 4. 2 Dataset setelah proses preprocessing 33
Gambar 4. 3 Hasil dari Proses TF IFD 40
Gambar 4. 4 Hasil uji model klasifikasi OVA 48
Gambar 4. 5 Hasil Confusion Matrix klasifikasi 49
Gambar 4. 6 Matrix confusion binary clasifications I 52
Gambar 4. 7 Matrix confusion binary clasifications II 52
Gambar 4. 8 Matrix confusion binary clasifications III 53
Gambar 4. 9 Matrix confusion binary clasifications IV 54
Gambar 4. 10 Matrix confusion binary clasifications V 55
Gambar 4. 11 Perbandingan Hasil pengujian 56
Gambar 4. 12 perbandingan akurasi per kategori kelas 58
Gambar 4. 13 Hasil tahapan text processing teks 59
Gambar 4. 14 code program panggil fugsi prediction 60

DAFTAR TABEL

Tabel 3. 1 Tahapan Case Folding 24
Tabel 3. 2 Tahapan Text Cleaning 24
Tabel 3. 3 Tahapan Tokenizing 25
Tabel 3. 4 Tahapan Stemming Word 25
Tabel 3. 5 Tahapan Stopword Remover 25
Tabel 3. 6 SVM Binery Metode One vs All 28
Tabel 3. 7 Confusion Matrix 30
Tabel 4. 1 Sample Dataset berdasarkan per kelas 32
Tabel 4. 2 Hasil Case Folding 33
Tabel 4. 3 Hasil Text Cleaning 34
Tabel 4. 4 Hasil Tokenizing 35
Tabel 4. 5 Hasil Stemming Word 36
Tabel 4. 6 Hasil Stopword Remover 38
Tabel 4. 7 Ranking nilai bobot 10 tertinggi 40
Tabel 4. 8 Analisis mencari nilai TF dan df 41
Tabel 4. 9 Analisis nilai IDF dan TF IDF 42
Tabel 4. 10 Analisis hasil dari TF IDF 43
Tabel 4. 11 Gambaran One vs All pada Klasifikasi Jurnal 44
Tabel 4. 12 Tabel data confusion matriks All 49
Tabel 4. 13 Hitung Precission kelas 50
Tabel 4. 14 Hitung Recall kelas 51
Tabel 4. 15 Mencari nilai F-1 Score 51
Tabel 4. 16 Confusion Matrix binary clasifications I 52
Tabel 4. 17 Confusion Matrix binary clasifications II 53
Tabel 4. 18 Confusion Matrix binary clasifications II 53
Tabel 4. 19 Confusion Matrix binary clasifications IV 54
Tabel 4. 20 Confusion Matrix binary clasifications V 55
Tabel 4. 21 Hasil pengujian SVM OVA 57
Tabel 4. 22 Hsdil tahapan bobot kata teks 59

Tabel 4. 23 Data Hasil Prediksi..... 61

ABSTRAK

Penelitian ini memiliki tujuan untuk membuat model klasifikasi bidang fokus penelitian berupa jurnal yang terindeks pada portal Garuda dengan menggunakan metode *Multiclass support vector machines* (SVM). Tahapan dalam pelaksanaan penelitian ini pertama-tama mengumpulkan *dataset* jurnal yang akan diuji terdiri dari lima kategori yaitu *Arts & Humanities, Life Sciences & Medicine, Natural Sciences, Engineering & Technolog,* dan *Social Sciences & Management.* Selanjutnya tahap *preprocessing* yang terdiri dari *case folding, text cleaning, tokenisasi, stemming word,* dan *stopword removal.* Tahap selanjutnya yaitu pembobotan kata (*term*) untuk menghitung nilai bobot setiap token serta pemisahan *data training* dan *data testing* dengan perbandingan 7:3. Proses klasifikasi bidang fokus penelitian dilakukan menggunakan metode *multiclass* SVM dengan pendekatan OVA (*One vs All*). Hasil dari proses klasifikasi diuji serta dievaluasi untuk mendapatkan besarnya tingkat nilai akurasi prediksi model perolehan akurasi sebesar 48 % dan untuk perbandingan akurasi dari setiap kelas ketegori, kelas "*Art & Humanities*" mempunyai akurasi tertinggi dengan nilai sebesar 82% dan akurasi terkecil terdapat pada kelas "*Life Sciences & Medicine*" serta kelas "*Natural Sciences*" dengan nilai sebanyak 78%.

Kata Kunci : Klasifikasi, Multiclass SVM, Jurnal Penelitian, Text Mining

ABSTRACT

This study aims to create a classification model for research focus areas in the form of indexed journals on the Garuda portal using the Multiclass support vector machines (SVM) method. The stages in conducting this research are first to collect a dataset of journals to be tested consisting of five categories, namely Arts & Humanities, Life Sciences & Medicine, Natural Sciences, Engineering & Technolog, and Social Sciences & Management. Next is the preprocessing stage which consists of case folding, text cleaning, tokenization, word stemming, and stopword removal. The next stage is term weighting to calculate the weighted value of each token and the separation of training data and testing data with a ratio of 7:3. The process of classifying research focus areas was carried out using the multiclass SVM method with the OVA (One vs All) approach. The results of the classification process were tested and evaluated to obtain the magnitude of the accuracy value of the prediction model for an accuracy of 48% and for a comparison of the accuracy of each category class, the "Art & Humanities" class had the highest accuracy with a value of 82% and the smallest accuracy was found in the "Art & Humanities" class. Life Sciences & Medicine" and the "Natural Sciences" class with a score of 78%.

Keywords : Classification, Multiclass SVM, Research Journal, Text Mining

BAB I
PENDAHULUAN

1.1 Latar Belakang

Publikasi ilmiah merupakan sebuah terbitan karya tulis ilmiah yang bersifat akademis dan teruji, dimana sebagian besar karya yang diterbitkan terpublish di *system* atau portal Scopus dalam bentuk jurnal ilmiah, buku, makalah konferensi, dan sebagainya. Perkembangan publikasi ilmiah di Indonesia sendiri yang terindeks pada portal Internasional seperti Scopus dalam lima tahun belakang ini mengalami peningkatan yang cukup pesat. Berdasarkan ulasan dan visualiasi pada *system* Scimago, Negara Indonesia dalam penerbitan publikasi ilmiah telah unggul dan berhasil menduduki peringkat pertama pada tahun 2019 sampai 2021 dengan jumlah publikasi terindeks Scopus se-ASEAN dengan capaian angka 50.965 pada tahun 2020 dan 49.350 di tahun 2021 kemarin diikuti Negara Malaysia dan Singapura dengan jumlah publikasi 41.938 dan 26.486 publikasi.

Setiap informasi tentang publikasi lokal terutama jurnal yang berstandar internasional (terindeks scopus) dapat dilihat dan direview pada *platform* Garuda maupun portal Sinta yang merupakan suatu *system* sumber informasi publikasi ilmiah di Indonesia dan media informasi pendataan maupun pengukuran kinerja iptek yang meliputi kinerja publikasi penelitian, jurnal, *author* (dosen ataupun peneliti) sampai kinerja masing-masing afiliasi (instansi iptek) yang dikelola oleh Kemdikbud. Setiap infomasi publikasi pada *platform* tersebut bersifat *open access* sehingga dapat dimanfaatkan oleh sivitas akademika sebagai referensi dalam penyusunan dan pelaksanaan penelitian, membuat tugas perkuliah, dan sebagainya. Dengan adanya portal sumber informasi tersebut serperti Garuda ataupun Sinta, sumber referensi akan lebih tercukupi. Tidak hanya dari buku, namun publikasi artikel-artikel jurnal ter-*update* seiring dengan perkembangan ilmu terkini.

Publikasi karya ilmiah yang terpublish di *platform* Garuda maupun Sinta mencakup semua bidang ilmu pengetahuan seperti humaniora, seni, ilmu sosial, ilmu perilaku, teknik, matematika & *computer*, kimia & biologi, fisika, dan sebagainya, setiap jurnal sudah terbagi berdasarkan bidang fokus jurnal sehingga dapat mempermudah *visitor* maupun civitas dalam pencarian publikasi sesuai dengan bidang fokus yang diinginkan. Namun tidak semua publikasi seperti halnya jurnal yang sudah terklasifikasi berdasarkan bidang fokus penelitian, terutama jurnal-jurnal *up to date*. Adapun *problem* yang dialami oleh banyak pengembang sistem adalah dalam menentukan klasifikasi bidang fokus dari publikasi-publikasi baru yang masih berdasarkan prediksi sepihak, karena dalam menentukan klasifikasi jenis bidang fokus jurnal hanya berdasarkan pada perkiraan isi konten maupun hasil *reviewer* yang memungkinkan membutuhkan waktu yang cukup lama.

Sehingga dibutuhkannya suatu sistem untuk penentuan klasifikasi bidang fokus pada publikasi-publikasi terutama pada jurnal penelitian yang terindeks pada portal Sinta ataupun Garuda yang bekerja secara otomatis dengan kategori bidang fokus yang digunakan berdasarkan *subject* atau kategori pada portal QS Top Universities (QS World University Ranking). Portal QS World University Ranking adalah portal publikasi tahunan dalam perangkingan tingkat universitas yang dikelola oleh Quacquarelli Symonds (QS). Sebelumnya portal QS World University Rankings dikenal sebagai THE-QS World University Rankings. Sistem portal QS sekarang terdiri dari beberapa subjek peringkat global yang terdiri dari lima regional independen seperti Asia, Asia Tengah, Eropa, Amerika Latin, dan Wilayah Arab serta BRICS. Hal ini menjadi satu-satunya portal yang mempresentasikan tentang perangkingan tingkat internasional yang telah mendapatkan persetujuan dari lembaga International Ranking Expert Group (IREG).(O'Callaghan, 2010)

Teknik *Text mining* merupakan teknik hasil dari perkembangan metode *data mining* yang dapat diimplementasikan dalam mengatasi *problem* tentang pengklasifikasian bidang fokus jurnal-jurnal penelitian berdasarkan yang uraian permasalah diatas dengan pengkategorian bidang fokus

berdasarkan pada *subject* atau kategori portal QS World University Ranking. Hal ini dapat membantu pihak pengembang maupun pengelola Sistem Garuda ataupun Sinta dalam penentuan pengelompokan bidang fokus jurnal baru secara otomatis dengan hasil akurasi pengujian yang relavan.

Melihat dari uraian seperti diatas, maka penulis mengusulkan klasifikasi publikasi karya ilmiah berupa jurnal penelitian berdasarkan bidang Fokus pembahasan pada penelitan dengan judul "Klasifikasi Bidang Fokus Publikasi Jurnal Penelitian yang Terindeks pada Portal Garuda dengan metode *Multiclass Support Vector Machines* (SVM)".

1.2 Perumusan Masalah

Berdasarkan paparan permasalahan latar belakang pada sub bab sebelumnya, maka diperoleh suatu perumusan masalah tentang bagaimana peneliti dapat membangun sebuah model mekanisme dengan mengimplementasikan *Text mining* dan algoritma SVM *Multiclass* dalam menguji klasifikasi publikasi jurnal penelitian yang terindeks pada portal Garuda berdasarkan bidang fokus penelitian.

1.3 Pembatasan Masalah

Dalam menyusun laporan tugas akhir ini, penulis memberikan dan menetapkan batasan masalah berdasarkan perumusan masalah diatas yaitu :

1. Modul klasifikasi ini merupakan mekanisme *system* yang nantinya digunakan dalam penentuan klasifikasi dengan ruang lingkup pembagian publikasi karya ilmiah berupa jurnal-jurnal penelitian berdasarkan kategori bidang fokus penelitian.
2. *Dataset* yang dipakai dalam penelitian ini bersumber pada portal Garuda ataupun Sinta berupa data jurnal penelitian dan hanya mengambil data judul beserta label kategori bidang fokus penelitian.
3. Data-data jurnal yang diambil di portal Garuda ataupun Sinta berupa data jurnal yang terindeks pada Scopus.

4. Kategori kelas yang digunakan sesuai dengan bidang fokus pada portal QS Top Universities (QS World University Ranking), diantaranya : *Arts & Humanities, Life Sciences & Medicine, Natural Sciences, Engineering & Technolog,* dan yang terakhir *Social Sciences & Management.*
5. Implementasi algoritma *Multiclass Support Vector Machines* yang akan diimplementasikan pada penelitian ini berfokus pada penggunaan OVA (*One vs All*) sebagai metode dalam klasifikasi bidang fokus jurnal dengan 5 kelas.
6. Luaran *system* akan menginfomasikan hasil pengujian klasifikasi data jurnal-jurnal berdasarkan bidang fokus masing-masing berupa *presentase* hasil nilai akurasi pengujian.

1.4 Tujuan

Adapun tujuan yang akan dicapai dalam pelaksanaan tugas akhir ini adalah :

1. membantu dalam pengkategorian bidang fokus jurnal penelitian terbaru (*up to date*) pada portal Garuda ataupun Sinta yang terindeks Scopus secara otomatis.
2. Menghasilkan suatu model program yang dapat mengklasifikasi jurnal-jurnal penelitian berdasarkan kategori kelas masing-masing yang bekerja secara otomatis.

1.5 Sistematika Penulisan

Sistematika penulisan yang akan digunakan oleh penyusun serta penulis dalam pembuatan laporan tugas akhir adalah sebagai berikut :

BAB 1: PENDAHULUAN

Pada bab ini penulis mengungkapkan latar belakang dalam penetapan judul, rumusan masalah, pembatasan masalah, tujuan, serta sistematika penulisan.

BAB 2: TINJAUAN PUSTAKA DAN DASAR TEORI

Bab ini menyinggung pada penelitian-penelitian terdahulu serta dasar teori yang dapat mendukung dalam membantu penulis untuk

mempelajari teori-teori tentang *Text mining* berupa metode untuk klasifikasi dengan sumber data teks untuk penelitian ini.

BAB 3: METODE PENELITIAN

Bab ini mengungkapkan bagaimana proses dan tahapan penelitian yang dimulai dari memperoleh *dataset* hingga proses klasifikasi data.

BAB 4: HASIL PENELITIAN

Pada bab ini penulis mengungkapkan hasil penelitian yakni hasil penentuan akurasi pengujian klasifikasi publikasi jurnal berdasarkan dengan bidang fokus.

BAB 5: KESIMPULAN DAN SARAN

Bab ini penulis memaparkan hasil kesimpulan dari hasil pelaksanaan dan langkah-langkah penelitian dari langkah awal sampai akhir.

BAB II
TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Penelitian Terdahulu

Penelitian-penelitian terdahulu yang konsep penelitian mirip dan berhubungan dengan pokok permasalahan pada penelitian ini yaitu Penerapan algoritma SVM dan *Text mining* pada klasifikasi dengan sumber data berupa teks telah banyak dilakukan di penelitian–penelitan sebelumya. Salah satunya permasalahan pada aktifitas utama pengguna twitter dalam menunjukan kepribadian mereka melalui *tweet*. Kepribadian menjadi suatu entitas yang unik terhadap seseorang dalam bersikap serta berperilaku tentang segala hal yang mengarah ke dirinya, sehingga setiap individu memilki perbedaan. Lantas untuk mengetahui jenis kepribadian seseorang dari suatu *tweet* yang mereka *publish* di Twitter. Dapat dilakukan dengan teknik klasifikasi data teks mengimplementasikan metode *Support Vector Machines Multiclass* dengan pendekatan ilmu *One vs One*(OVO) dan *One vs All*(OVA). Dengan studi kasus penelitian tersebut dihasilkan klasifikasi dalam bidang kepribadian seseorang terhadap penggunaan *tweet* tergantung pada suasana yang dirasakan masing-masing pengguna. Dimana hasil yang peroleh dari penggunaan *case folding* ataupun tidak dihasilkan nilai akurasi dengan *One vs One* sebesar 84% dan 83%. sedangkan menggunakan *One vs All* yaitu 88.3% dan 88% tepat (Fikriani *et al.*, 2019).

Meningkatnya pertumbuhan karya ilmiah penelitian yang membuat kondisi sulitnya dalam penyimpanan arsip penelitian tersebut. Kesulitan yang akan muncul yaitu ketika terdapat peneliti lain yang butuh suatu penelitian dimana penelitian tersebut dapat mendukung penelitian yang sedang dilaksanakan. Oleh karena itu, perlunya media simpan untuk berkas penelitian yang dilakukan berdasakan pada pembagian kategori. Sedangkan pengkategorian penelitian secara manual tentunya memerlukan waktu yang cukup lama. Pemberian label kategori secara otomatis dapat dilaksanakan

dengan implementasi algoritma *Support Vector Machines* (SVM) dalam mengembangkan suatu sistem klasifikasi jurnal penelitian berdasarkan pada kategori topik penelitian. Hasil akurasi yang diperoleh dari pengkategorian yang dihasilkan metode ini sudah sangat baik dengan tingkat akurasi sebesar 90% (Jumeilah, 2017).

Pantun yang merupakan jenis puisi lama dari kesusastraan Indonesia dimana masyarakat mengenal mengenai pantun yang bersifat profan ataupun jenis pantun lainnya. Sehingga penggunaan suatu pantun di lingkup masyarakat dapat berkembang sesuai dengan jiwa dari pantun tersebut. Namun pengetahuan tentang macam-macam jenis pantun sekarang ini masih terbilang terbatas untuk dikenal oleh pakar yang ahli di bidang kesustraan. Maka dari itu, dibutuhkannya model sistem yang mampu mengklasifikasi jenis pantun berdasarkan kategori masing-masing dengan mengimplementasikan teknik *text processing* metode *Support Vector Machines* (SVM). Hasil yang diperoleh pada pengujian ini cukup baik dalam klasifikasi jenis tema topik pantun dengan capaian akurasi 81,91%. Selain itu perolehan *nilai precision, recall,* dan spesifitas pada pantun anak lebih tinggi dari pantun lainnya, yaitu sebesar 90.63%, 87,88%, dan 95.08% (Irmanda and Astriratma, 2020).

Banyaknya pengguna media sosial berupa Twitter di Indonesia yang dapat dijadikan sebagai media komunikasi sebagai penyaluran keluhan, saran, ataupun pertanyaan oleh para pelaku bisnis seperti PT KAI pada @KAI121 yang berkaitan dengan pelayanan ataupun apresiasi kepuasan pelanggan sebagai *respond* dalam penilaian suatu hasil produk bisnis tersebut. Kelemahan dalam pemakaian media Twitter yaitu *data text tweet* yang belum terstruktur dengan jumlah yang cukup banyak mencapai 2400 *tweet* tiap hari. Hal ini dapat mempersulit pebisnis dalam mengetahui tingkat *sentiment public*. Sehingga perlunya klasifikasi ke dalam kategori netral, positif, dan negatif yang dapat mempemudah mengetahui *sentiment public* terhadap @KAI121. Klasifikasi dilakukan dengan penelitian klasifikasi data Twitter

@KAI121 untuk mengetahui kelompok kategori terbaik dari nilai akurasi yang didapatkan dengan implementasi algoritma *Multiclass Support Vector Machines* (SVM). Untuk hasil yang diperoleh dari penelitian memperoleh nilai akurasi tertinggi dengan metode *multiclass* SVM OAA sebesar 80.59 %. Sedangkan untuk perolehan setiap kelompok, *sentiment* positif sebesar 11%, *sentiment* netral sebesar 58%, dan *sentiment* negatif sebanyak 31% (Fitriana and Sibaroni, 2020).

Music memiliki pengaruh yang cukup besar dalam menggugah emosi manusia, hal ini dikarenakan pada suatu lirik lagu bisa mendeskripsikan makna emosi tersirat dari lagu tersebut sebagaia karya curahan perasaan pribadi dari penulis lirik lagu tersebut. sampai saat ini jumlah *music* atau lagu yang terus bertambah banyak, membuat sulitnya menentukan emosi pada lagu tersebut. Oleh karena itu, perlunya penelitian untuk membangun suatu model klasifikasi menggunakan *text classification* menggunakan metode *multiclass* SVM pada kasus klasifikasi emosi berdasarkan jenis emosi lirik lagu. Hasil dari penelitian tersebut berdasarkan hasil pengujian pada setiap model memperoleh performa model sistem yang paling baik dengan tingkat *accuracy* sebanyak 92,13%. Kenaikan ataupun penurunan nilai pengujian dapat berpengaruh apabila menggunakan *dataset* perbaris dengan penambahan nilai *accuracy* sebanyak + 3,32%, sedangkan pada *dataset* perbait nilai *accuracy* hanya + 2,16% dan terakhir pada *dataset* keseluruhan lagu menghasilkan *accuracy* nilai hanya -1,33% (Sastypratiwi *et al.*, 2022).

Ulasan produk dalam suatu *platform marketplace* dapat berpengeruh dalam hasil penjualan dengan dampak yang cukup signifikan kepada minat beli konsumen lain sebelum membeli produk yang tepat. Analisis pada hasil ulasan produk biasanya dapat dilakukan dengan melihat jumlah perolehan bintang yang diberikan konsumen ataupun dapat dilakukan dengan membaca satu persatu ulasan produk, namun jika ulasan produk jumlahnya cukup banyak akan mempersulit dan memakan waktu yang cukup lama dalam penilaian suatu produk tersebut. Solusi dalam masalah tersebut dapat diselesaikan dengan penelitian terkait analisis *respond* pelanggan berupa

ulasan produk dengan sistem klasifikasi menggunakan algoritma SVM berdasarkan pengelompokan kelas positif dan negatif. Hasil evaluasi yang diperoleh dalam penelitian ini menghasilkan akurasi *sentiment* dengan perolehan nilai sebanyak 90% dan tingkat akurasi terendah sebanyak 81% (Setiawan *et al.*, 2022).

2.2 Dasar Teori

2.2.1 *Text Mining*

Teknik *Text mining* yaitu suatu ilmu yang mempelajari bagaimana *user* dapat memahami kumpulan-kumpulan dokumen dengan memanfaatkan *tools* analisis, dengan mengekstraksi suatu informasi yang diperlukan, menarik, pola data informasi baru ataupun tersirat yang bersumber dari kumpulan data-data yang bervariasi. Sumber data dari *Text mining* sendiri dapat berupa data dokumen yang tidak atapun belum terstruktur misalkan berupa *data text* yang tidak atau belum terstruktur. (Feldman and Sanger, 2006)

Text mining menjadi salah satu teknik khusus di bidang ilmu *data mining* yang artinya sendiri menambang data dari sumber data teks, dimana sumber data biasanya didapatkan dari suatu dokumen. Tujuan teknik ini adalah mencari kata-kata penting yang dapat menjadi *point* penting dari suatu dokumen sehingga bisa dilakukan suatu analisis keterhubungan masing-masing dokumen (Kurniati *et al.*, 2015). *Text mining* mampu menganalisa dan mengelompokan dokumen berdasarkan dari kata-kata yang terkandung dalam dokumen tersebut, serta dapat menentukan kesamaan kata antar dokumen untuk mengetahui bagaimana setiap dokumen saling terhubung dengan variabel lainnya. Implementasi yang paling umum dilakukan teknik *Text mining* saat ini misalnya analisa sentimen, penyaringan suatu *spam* teks, mengukur preferensi pelanggan, pengelompokan topik penelitian, meringkas dokumen, dan lainnya.

Teknik *Text mining* sendiri mempunya beberapa cabang tipe antara lain : (Kurniati *et al.*, 2015)

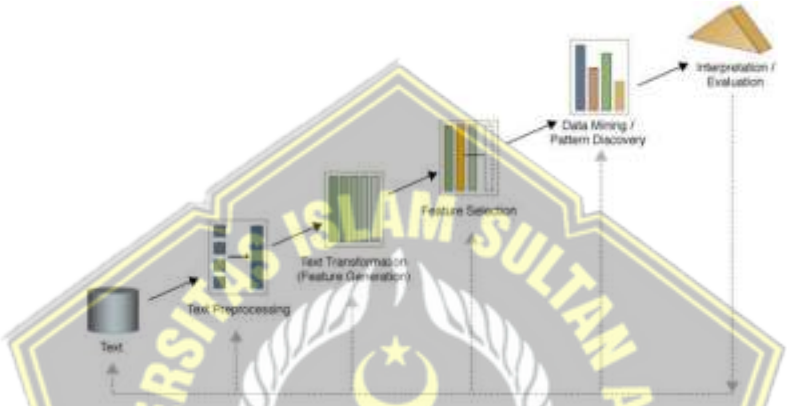
a. *Search and Information Retrieval* yaitu penyimpanan serta pencarian ulang dokumen berupa teks, seperti mesin pencarian dengan *keyword* pencarian.
b. *Document Clustering* yaitu pengkategorian dan pengelompokan data seperti dokumen, paragraf, potongan, dan istilah menggunakan metode *mining*.
c. *Document Classification* yaitu pengkategorian dan pengelompokan data seperti dokumen, paragraf, potongan, dan istilah menggunakan metode *document classification.*
d. *Web Mining Data dan Text mining* pada data *internet* berfokus skala dan hubungan dari relasi antar *website*.
e. *Information Extraction* yaitu ekstraksi data-data berupa fakta yang relevan.
f. *Natural Language Processing* yaitu pemrosesan bahasa rakitan yang digunakan dalam bahasa komputer.
g. *Concept Extraction* yaitu pengelompokan kata dan *frase* dalam *group* yang sama Tahap *Preprocessing Text*.

Teknik *Text mining* memiliki perbedaan dengan teknik *data mining*. Dimana perbedaan tersebut terletak pada sumber data yan dikelola, teknik *Text mining* menggunakan *source* data dari kumpulan-kumpulan dokumen teks yang pada umumnya berbentuk data tidak atau belum terstrukur (*unstructured text)*. Sedangkan teknik *Data Mining* untuk mencari relasi satu bagian data dengan yang lain berdasarkan langkah-langkah tertentu dengan sumber data yang sudah rapi atau terstruktur.

Arsitektur fungsional pada suatu sistem dengan Teknik *Text mining* sendiri terdiri dari 4 bidang tingkat fungsional sistem *Text mining*, yaitu dapat dilihat pada Gambar 2.1 (Feldman and Sanger, 2006) :

Gambar 2. 1 Arsitektur Text Mining

Untuk lebih jelasnya dalam pemahaman tentang *text mining* berikut bagaimana tahapan-tahapan ataupun proses dalam *text mining* dapat dilihat pada Gambar 2.2.(Latif, 2018).



Gambar 2. 2 Proses Text Mining

a. *Text*

Permasalahan yang akan dihadapi dalam implementasi teknik *text mining* sama dengan tekniik *data mining* yaitu pada jumlah data yang cukup besar, dimensi yang tinggi, struktur data yang berubah-ubah, dan data *noise*. Dari permasalahn tersebut, yang paling menonjol dari implementasi *text mining* sendiri yaitu data yang akan diolah berupa data tidak terstuktur (*unstructured data)* atau minimal *semi structured*. Hal ini menjadi tantangan pada teknik *text mining* berupa struktur *text* yang *complex* dan tidak lengkap, bisa juga dari suatu terjemahan yang tidak sesuai dan penggunaan bahasa yang berbeda ditambah hasil translasi yang tidak akurat.

b. *Text Preprocessing*

Pada tahapan ini analisis sematik yaitu kebenaran arti dan sintaktik dengan kebenaran susunan terhadap suatu teks. Tujuannya untuk mempersiapkan data teks menjadi data yang siap untuk

pengolahan lebih lanjut. Untuk lebih jelasnya tentang *text preprocessing* akan dibahas pada sub bab selanjutnya.

c. *Text Transformation*

Tahapan ini berfokus pada proses untuk memperoleh hasil representasi dokumen dengan mengubah kata-kata ke bentuk dasarnya (root data) dan pengurangan dimensi kata pada suatu dokumen.

d. *Feature Selection*

Tahapan ini masih termasuk dalam proses *text transformation* dalam pengurangan dimensi kata pada suatu dokumen.

e. *Pattern Discovery*

Pattern Discovery atau lebih dikenal dengan *Data Mining* dimana tahapan dalam mendapatkan suatu pola *knowledge* dari keseluruhan teks. Dimana proses pada data mining berupa pencarian data yang dilakukan secara otomatis dengan informasi yang berguna dalam penyimpanan data yang berukuran cukup besar (Maulana, 2018). Dalam pencarian pola atau informasi penting pada teks, hal ini perlunya penggunaan metode ataupun algoritma pada *data mining* yang mana *text mining* sendiri termasuk dalam bagian dari *data mining*.

f. *Interpretation* / *evaluation*

Pada tahapan ini proses *mining* dapat presentasikan kedalam bentuk tertentu, kemudian dapat dilakukan proses evaluasi. Evaluasi dilakukan dengan iterasi ke satu atau beberapa tahapan sebelumnya. Serta hasil iterasi akan menghasilkan data dalam bentuk *visual* ataupun nilai akurasi persentasi dari hasil uji pada *text mining.*

2.2.2 *Data Preprocessing*

Preprocessing Data adalah proses yang dapat mengolah data mentah ke bentuk data yang lebih terstruktur serta mudah dipahami baik umtuk *user* ataupun mesin. *Preprocessing data* ini sangat penting untuk dilaksanakan hal ini karena suatu data mentah pada umumnya tidak memiliki format yang terstruktur dan teratur. Selain itu, teknik pada *data mining* tidak dapat

mengolah data mentah sehingga dari penerapan proses ini mempermudah sistem dalam menolah data ke tahapan berikutnya yaitu analisis data.

Pada proses pengolah kata atau *Preprocessing data*, Dokumen dapat terbagi menjadi beberapa bab, sub bab, paragraf, kalimat, kata, dan bahkan suku kata atau token. Umumnya pada suatu dokumen mempunyai struktur yang tidak teratur dan sembarangan. Maka dari itu, perlunya suatu proses yang dapat mengubah bentuk struktur data yang sebelumnya tidak teratur dan terstruktur ke dalam bentuk data yang terstruktur.(Jumeilah, 2017)

Preprocessing data dieksekusi agar data yang akan digunakan bersih dari *noise*, mempunyai dimensi data yang lebih kecil, serta lebih teratur atau terstruktur, agar data tersebut dapat diproses ke tahapn selanjutnya. Tahapan pada *Preprocessing data* memiliki beberapa proses, diantaranya : *case folding, text Cleaning, Normalization, Stemming Word, tokenizing,* dan *stopwords removing* (Sastypratiwi *et al.*, 2022). berikut diagram tahapan dalam *text* processing dapat dilihat pada Gambar 2.3 dimana tahapan-tahapan berikut sesuai dengan *step* yang dibutuhkan dalam penelitian kali ini.

Gambar 2. 3 Tahapan *Preprocessing Data*

a. Case Folding

Tahapan pada *Text Preprocessing* yang pertama kali dieksekusi adalah tahap *case folding,* dimana *Case folding* dilakukan untuk menyeragamkan karakter data dengan mengubah seluruh huruf menjadi huruf kecil. Pada proses ini karakter-karakter *alphabet* ‘A’ sampai ‘Z’ yang terdapat pada data diubah menjadi karakter alphabet ‘a’ sampai ‘z’.

b. Text Cleaning

Data Cleaning adalah proses membersihkan data teks dari karakter seperti angka, *link* url, email dan karakter-karakter yang tidak memiliki makna dan hubungan dengan data informasi pada dokumen, seperti (0-9,!@#$%&*_+-={}[]:;”’/?<>.) yang merupakan Karakter selain huruf

alphabet akan dihapus dari data yangdianggap sebagai karakter *delimeter*. *Delimeter* sendiri merupakan urutan satu atau lebih suatu karakter untuk menentukan batas pemisah antar karakter tersebut.

c. *Tokenizing*

Tokenizing atau dapat disebut dengan *Lexical Analysis* adalah suatu proses pemotongan data *text* atau kalimat menjadi suku kata yang lebih kecil yang bisa disebut dengan istilah token.

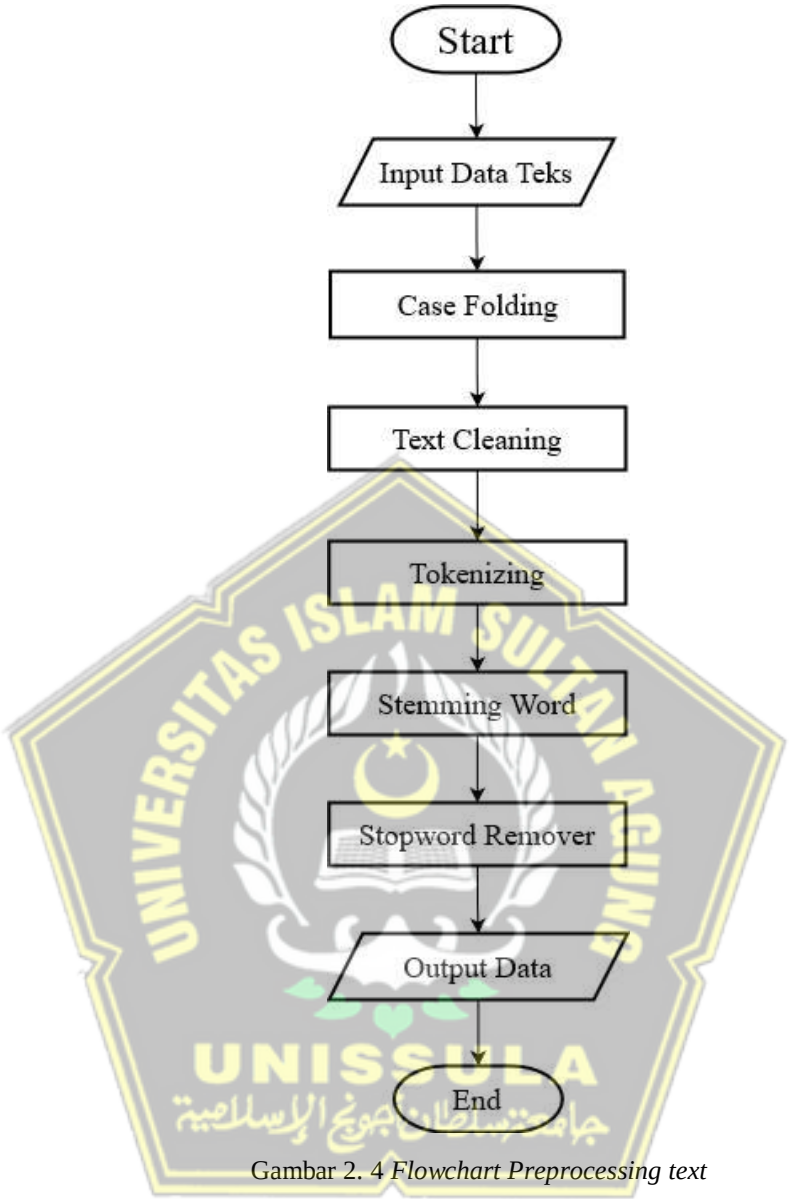
d. *Stemming Word*

Stemming merupakan proses untuk mengkonversi bentuk kata menjadi kata dasar atau dapat diartikan sebagai tahapan dalam mencari *root* dari kata tersebut. Dalam *stemming* varsi bahasa Indonesia menghilangkan kata dari awalan kata seperti kata imbuhan kata dasar (*root*). Misalkan jika suatu kata mengandung awalan dan imbuhan seperti (be-, di-, ke-, -ku, -mu, atau -nya) maka imbuhan atau awalan tersebut akan dihilangkan dan kata tersebut menjadi kata dasarnya (*root*).

e. *Stopwords Removing*

Stopword Removing merupakan suatu metode yang sering digunakan untuk menghapus suku kata yang dianggap tidak penting untuk dijadikan sebagai kata kunci pada saat melakukan proses klasifikasi. Pada penelitian ini tahapan *stopword removing* menggunakan versi Internasional (*english*) yang didapatkan dari implementasi *library* nltk untuk *filtering* terhadap *dataset*. Suatu kata yang terdapat pada *stopword_list()* akan dihapus dari *dataset*.

Berikut gambar *Flowchart* dari proses *Preprocessing* data pada pada penelitian ini, terliha pada Gambar 2.4.



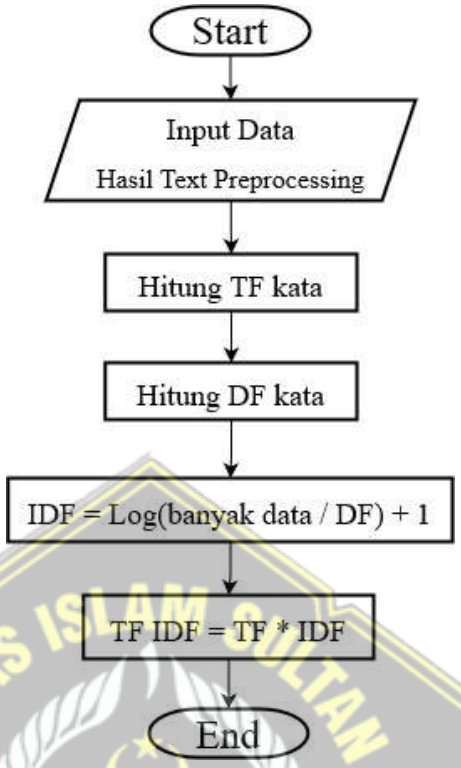
Gambar 2. 4 *Flowchart Preprocessing text*

Pada Gambar 2.4, merupakan proses *flowchart* dari *text preprocessing* pada tahapan penelitian ini. Dimana *data input* berupa teks dari judul jurnal yang nantinya akan melalui tahapan-tahapan tersebut. Untuk memperoleh *dataset* yang bersih dari *noise data*, karater yang tidak perlu, dan kata yang tidak memiliki makna untuk siap diolah pada pembobotan *term*.

2.2.3 Pembobotan *Term* (TF IDF)

Pembobotan kata atau dikenal denga *term weighting* merupakan suatu proses yang sangat penting dalam implementasi *text mining* setelah melewati proses *Preprocessing*, tujuan dari pembobotan kata adalah untuk mengubah data yang belum sekiranya belum teratur dan terstruktur menjadi lebih terstruktur serta memiliki bobot nilai, setelah itu hasil data yang terstruktur tersebut dapat di proses untuk klasifikasi menggunakan metode *classifier* (Ramadhan *et al.,* 2021). Perolehan nilai bobot setiap kata dapat berbeda-beda berdasarkan metode yang dipakai dalam pembobotan kata tersebut. Dalam proses penentuan pembobotan kata terdapat algoritma-algoritma yang dapat digunakan seperti TF, IDF, TF-IDF, RF, TF.RF, WIDF dan lainnya. *Term Frequency-Inverse Document Frequency* (TF-IDF) adalah salah satu algoritma yang dapat digunakan dalam metode pembobotan kata yang sering digunakan dari kategori *unsupervised term weighting* dimana cocok untuk melakukan bembobotan nilai teks. Metode pembobotan ini hasil indeks dari dua metode yaitu metode TF dan metode IDF.

Metode TF-IDF adalah metode untuk menentukan nilai frekuensi dari sebuah kata dalam dokumen ataupun artikel serta frekuensi pada banyaknya dokumen tersebut. Hasil perhitungan dapat menentukan seberapa relevan suatu kata tersebut di dalam sebuah dokumen (Arifidin, 1991). metode TF-IDF memberikan nilai bobot pada setiap kata kunci per kategori untuk mencari suatu kemiripan *key word* dengan kategori kata yang tersedia. Sebelum *dataset* diolah dalam pembobotan *term. Dataset* akan dieksekusi terlebih dahulu pada tahapan *Text Preprocessing* seperti tahap *case folding, tokenizing, filtering, dan stemming*, dan *stopword removal.* selanjutnya dieksekusi pada proses menghitung bobot setiap kata dengan metode TF-IDF untuk mencapai nilai bobot *query relevance* serta nilai *bobot similarity.* Berikut *flowchart* dari pemberian pembobotan TF IDF, ditampilkan pada Gambar 2.5.



Gambar 2. 5 *Flowchart* TF IDF

Pada Gambar 2.5 diatas, merupakan *flowchart* dalam proses pada pemberian nilai bobot setiap kata *dataset* judul jurnal yang sudah melewati proses *Text processing*. Pemberian nilai bobot dilakukan dengan metode TF IDF yang terdiri dari perhitungan TF untuk meghitung kata yang muncul pada dokumen. Perhitungan DF untuk menghitung banyaknya jumlah dokumen yang mengandung kata kunci tertentu, dan TF IDF yaitu hasil perkalian dari metode TF dan metode IDF untuk memperoleh\ nilai bobot dari setiap kata.

Metode TF-IDF pada dasarnya hasil integrasi dari perhitungan metode TF (*Term Frequency*) dan metode IDF (*Inverse Document Frequency*). Cara yang perlu dilakukan untuk menentukan suatu nilai dari integritas kedua metode statistik tersebut. Berikut ini merupakan rumus persamaan implementasi metode TF-IDF, dapat dilihat pada persamaan 1:

$$W_{ij} = tf_{ij}\, x\, \log\left(\frac{D}{df_j}\right) \quad \text{.....(1)}$$

Keterangan :

W_{ij} : bobot *term* (tj) pada sebuah dokumen (d)

tf_{ij} : jumlah kemunculan *term* (tj) pada sebuah dokumen (d)

D : jumlah seluruh dokumen yang diuji

df_j : jumlah dokumen yang terdapat *term* (tj) (min 1 *term*)

Seberapa besarnya nilai tf_{ij}, jika D = df_j, maka akan diperoleh hasil 0 (nol), hal tersebut karena hasil aritmatika dari log 1 yaitu 0, sehingga untuk perhitungan pada IDF ditambahkan dengan nilai 1 pada sisi belakang rumus IDF, sehingga perhitungan bobotnya menjadi seperti pada persamaan 2 berikut :

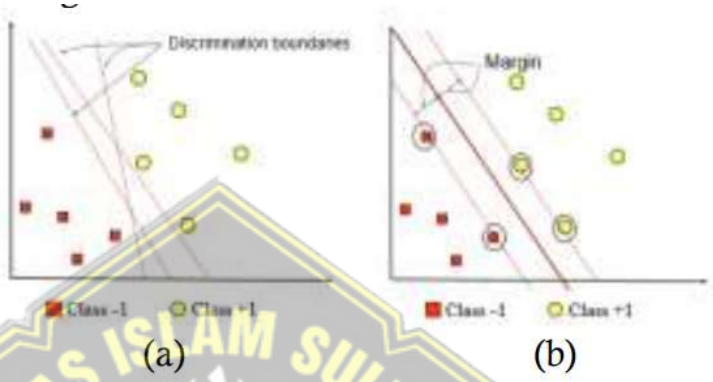
$$W_{ij} = tf_{ij}\ x\ \log\left(\frac{D}{df_j}\right) + 1 \quad \text{.....(2)}$$

2.2.4 *Support Vector Machine*s (SVM)

Support Vector Machines (SVM) adalah metode untuk melakukan pengklasifikasian data *linear* dan *non linear*. Cara kerja dari algoritma SVM adalah dengan menggunakan *node* peta *non linear* untuk mengubah data *training* ke dimensi yang lebih tinggi dan mencari garis *hyperlane* pemisah yang paling optimal. Data yang berada pada margin *hyperlane* disebut sebagai *support vector* (Fitriana and Sibaroni, 2020). Metode *Support Vector Machines* (SVM) menjadi metode *supervised learning* yang biasanya digunakan dalam proses klasifikasi seperti (*Support Vector Classification*) dan regresi (*Support Vector Regression*). Dalam pembuatan model klasifikasi algoritma SVM mempunyai konsep yang lebih matang dan lebih jelas secara matematis dibandingkan dengan algoritma klasifikasi lainnya. Serta metode SVM dapat mengelola klasifikasi dan regresi data *linear* maupun *non linear*.

Metode SVM diperlukan untuk mencari *hyperline* terbaik dengan memaksimalkan jarak antar kelas. *Hyperpline* merupakan fungsi yang digunakan untuk memisah antar kelas. Dengan visualisasi 2-D fungsi untuk

proses klasifikasi antar kelas disebut *sebagai line whereas*, fungsi yang digunakan untuk klasifikasi antas kelas dalam visualisasi 3-D disebut *plane similarly*, dan *hyperlane* adalah fungsi yang digunakan untuk klasifikasi di dalam ruang kelas dimensi yang lebih tinggi. Berikut contoh dari cara kerja SVM terlihat pada Gambar 2.6 (Delimayanti *et al.*, 2021)



Gambar 2. 6 Visualisasi Metode SVM

SVM adalah metode yang sudah sering diterapkan untuk keperluan jenis penelitian pengolahan data dan teknik *Text mining*. hal tersebut karena SVM menghasilkan perfoma yang lebih baik dari algoritma lainnya (Alita *et al.*, 2020). Cara kerja dari Metode SVM yaitu sistem pembelajaran yang engan ruang hipotesis berupa fungsi-fungsi *linier* dalam sebuah ruang fitur berdimensi tinggi, namum pada metode klasifikasi SVM hanya mampu menjalankan klasifikasi data dalam dua kelas (*clasification binar*). Pengembangan ilmu pada algoritma SVM terus dilakukan seiring dengan berkembangnya dan peningkatan penelitian sehingga metode SVM mampu untuk mengklasifikasikan data lebih dari dua kelas berupa model pendekatan *One Vs One* (OVO) dan *One Vs Rest* atau *One Vs All* (OVA), model pendekatan tersebut mampu mengatasi permasalahan dalam klasifikasi *multiclass* seperti pada penelitian dimana jumlah yang akan diklasifikasi lebih dari dua kelas (Alita *et al*., 2020).

Gambar 2. 7 Visualiasi margin kelas

Gambar 2.7 himpunan vektor masing-masing kelas dipisahkan dengan sepasang pembatas yang sejajar. Untuk pembatas pertama menjadi pembatas kelas 1 dan pembatas kedua menjadi pembatas bukan kelas 1 (kelas selain kelas 1) dan pembatas tengah disebut dengan *hyperlane* berikut persamaan yang diperoleh :

$$xi.w + b \geq +1 \; for yi = +1$$

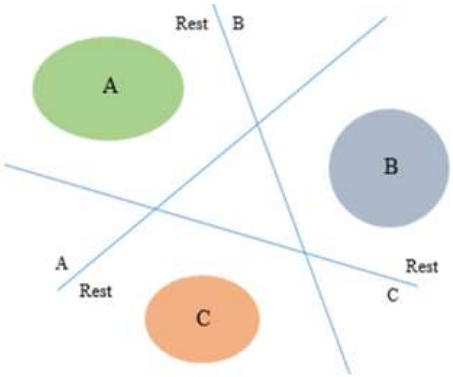
$$xi.w + b \leq +1 \; for yi = -1 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3)$$

Keterangan :

w : Bidang Normal.

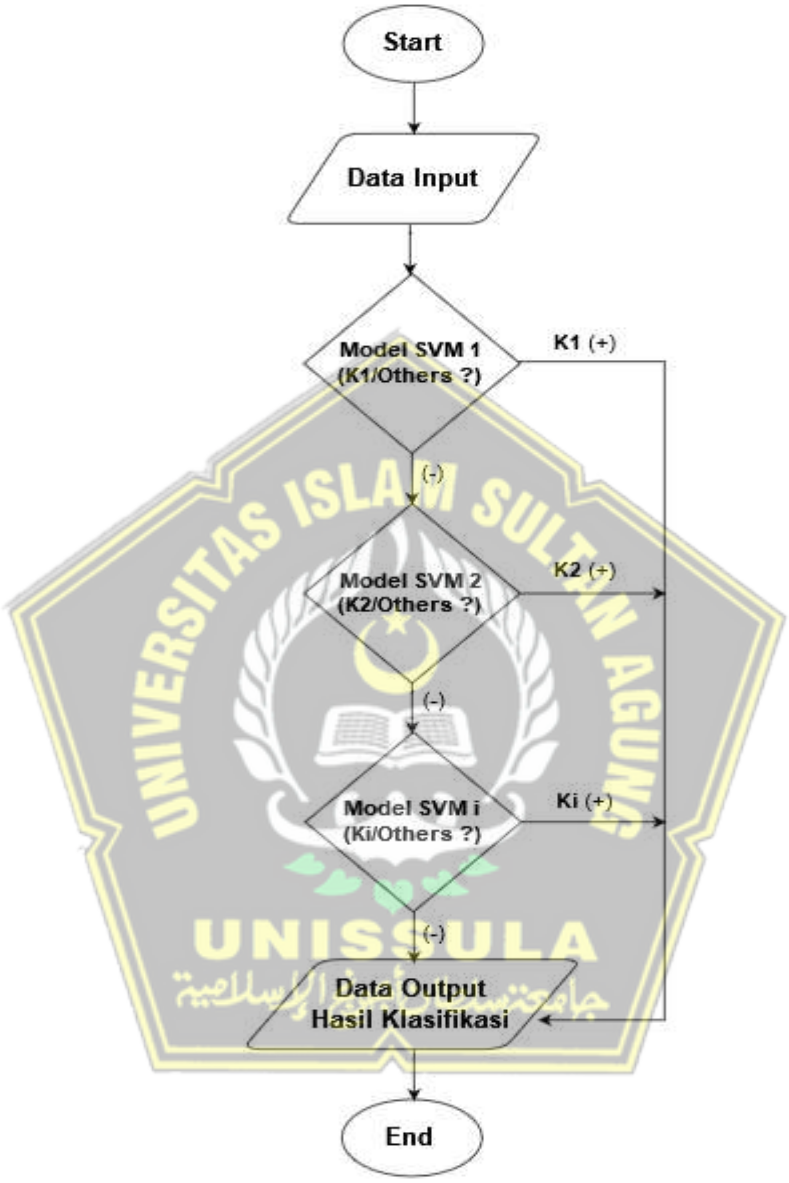
b : Bidang *relative* terhadap titik pusat.

Pada metode *One Vs All* (OVA) dimana metode ini melibatkan proses *training* satu klasifikasi per kelas dengan ketatapan sampel kelas tertentu sebagai *sample positif*, dan untuk semua sampel selain kelas tersebut sebagai *sample negative* seperti pada visualisasi Gambar 2.5. Metode ini menkhusukan proses klasifikasi agar menghasilkan akurasi sebagai *ouput*, tidak hanya berupa label kelas. Dari label kelas saja dapat menyebabkan ambigu, hal ini karena beberapa kelas diprediksi hanya untuk pengambilan keputusan sampel tunggal. Contoh pengambaran dari implementasi metode OVA terlihat pada Gambar 2.8 dan Gambar 2.9.



Gambar 2. 8 SVM *multiclass* OVA

Pada Gambar 2.8 merupakan diagram dari implementasi SVM *Multiclass* OVA dimana pada gambar 2.8 tersebut ada 3 kelas klasifikasi yaitu kelas A, B, dan C. untuk klasifikasi kelas A nantinya dibandingkan dengan kelas selain kelas A yaitu kelas B dan C. dan untuk kelas B dibandingkan dengan kelas A dan C. dan seterusnya untuk kelas C.

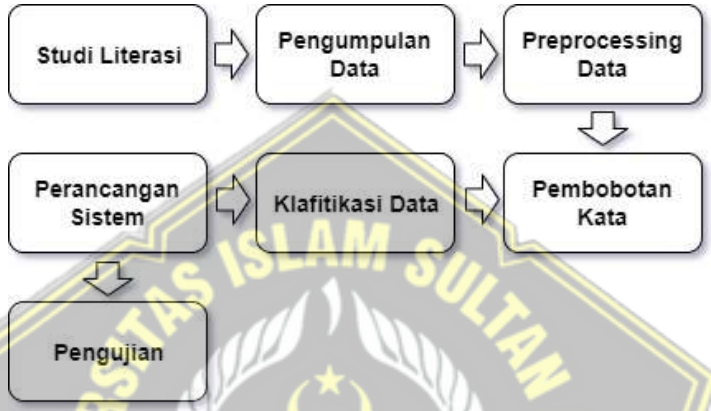


Gambar 2. 9 *Flowchart* SVM OVA

Gambar 2.9 merupakan *flowchart* dari SVM *Multiclass* OVA yang terdiri dari 3 kelas dimana pada *chart* kondisi kelas klasifikasi utama dibandingkan dengan kelas lainnya selain dari kelas tersebut. Sehingga dari proses klasifikasi dihasilkan *output* nilai akurasi klasifikasi tersebut.

BAB III
METODE PENELITIAN

Metode penelitian adalah suatu tata cara ataupun prosedur yang digunakan untuk menggumpulkan data, dengan perantara teknik tertentu. Dalam penelitian ini akan menjalankan beberapa metode penelitian pada Gambar 3.1:



Gambar 3. 1 Diagram proses Penelitian

3.1 Studi Literasi

Dengan melakukan studi literasi, penulis dapat mempelajari teori tentang *Text mining* tentang *Preprocessing* Text baik berupa proses pada metode *Natural Language Processing* (NLP) serta impelentasi metode klasifikasi berupa metode *Multiclass Support Vector Machines* (SVM) dari berbagai sumber literasi seperti halnya jurnal, artikel, buku, ataupun situs-situs *website*. Selain itu perlunya juga mempelajari beberapa teori lainnya yang dirasakan perlu dan berguna dalam penelitian ini.

3.2 Pengumpulan Data

Pengambilan data bersumber dari data pada portal Garuda ataupun portal SINTA, dengan data berupa data-data jurnal yang terindeks scopus. *Dataset* nantinya terbagi menjadi 2 bagian yaitu *data Training* dan *data testing* dimana perbandingan *dataset* sebanyak 7:3 dengan jumlah

keseluruhan data sebanyak 2.500 *dataset* jurnal penelitian, masing-masing kategori sebanyak 500 *dataset* per kategori pada penelitian ini.

Pada proses *Data training* dan *data testing* akan diubah menjadi data vektor. *Data training* nantinya akan digunakan untuk proses *training* algoritma atau metode dalam menentukan dan pengembangan model yang sesuai, sedangkan *data testing* nantinya digunakan untuk *testing* pada model untuk mengetahui performa dari model yang didapatkan pada tahapan *training*.

3.3 *Preprocessing Data*

Pada tahapan ini berupa *Preprocessing* data yaitu tahapan sebelum proses klasifikasi data yang dimana tahapan *Preprocessing data* untuk mengelola teks berupa judul-jurnal penelitian yang akan digunakan pada penelitian ini. Berikut proses-proses dalam *Preprocessing text* berupa :

a. *Case Folding*

Tahap *case folding* digunakan untuk merubah semua karakter terutama huruf atau karakter *alphabet* menjadi huruf kecil.

Tabel 3. 1 Tahapan *Case Folding*

Data Sebelum *Case Folding*	Data Setelah *Case Folding*
Management and control of the read out processors (TPPs) of the aleph time projection chamber.	management and control of the read out processors (tpps) of the aleph time projection chamber.

b. *Text Cleaning*

Tahap *text cleaning* untuk membersihkan *dataset text* dari angka, url, email dan karakter-karakter tanda baca seperti karakter selalin huruf ‘a’ sampai ‘z’.

Tabel 3. 2 Tahapan *Text Cleaning*

Data Sebelum *Text Cleaning*	Data Setelah *Text Cleaning*
management and control of the read out processors (tpps) of the aleph time projection chamber.	management and control of the read out processors tpps of the aleph time projection chamber

c. *Tokenization*

Tahap tokenisasi untuk memotong teks menjadi bagian yang lebih kecil berupa token.

Tabel 3. 3 Tahapan *Tokenizing*

Data Sebelum Tokenisasi	Data Setelah Tokenisasi
management and control of the read out processors tpps of the aleph time projection chamber	'management', 'and', 'control', 'of', 'the', 'read', 'out', 'processors', 'tpps', 'of', 'the', 'aleph', 'time', 'projection', 'chamber'

d. *Stemming Word*

Tahap *stemming word* untuk mencari serta mengubah kata imbuhan menjadi kata dasar atau *root* kata.

Tabel 3. 4 Tahapan *Stemming Word*

Data Sebelum *Stemming Word*	Data Setelah *Stemming Word*
'management', 'and', 'control', 'of', 'the', 'read', 'out', 'processors', 'tpps', 'of', 'the', 'aleph', 'time', 'projection', 'chamber'	'manag', 'and', 'control', 'of', 'the', 'read', 'out', 'processor', 'tpp', 'of', 'the', 'aleph', 'time', 'project', 'chamber'

e. *Stopword Remover*

Tahap *stopword remover* untuk menghilangkan kata yang dianggap tidak terlalu penting pada proses klasifikasi.

Tabel 3. 5 Tahapan *Stopword Remover*

Data Sebelum *Stopword Remover*	Data Setelah *Stopword Remover*
'manag', 'and', 'control', 'of', 'the', 'read', 'out', 'processor', 'tpp', 'of', 'the', 'aleph', 'time', 'project', 'chamber'	'manag', 'control', 'read', 'processor', 'tpp', 'aleph', 'time', 'project', 'chamber'

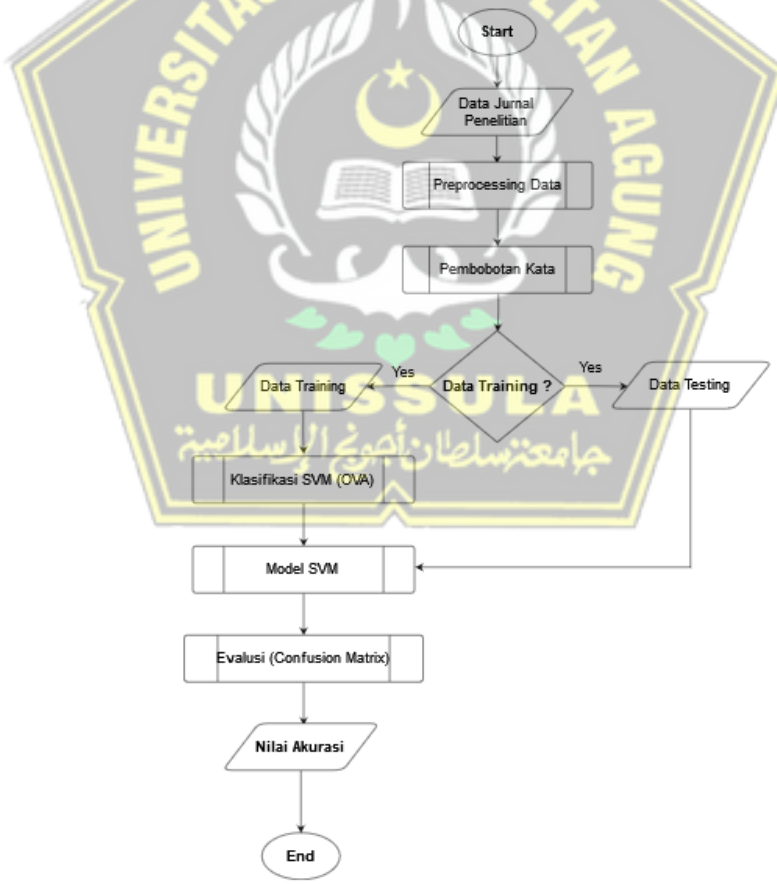
3.4 Pembobotan Kata TF IDF

TF-IDF adalah teknik pembobotan fitur kata pada data jurnal berupa judul jurnal penelitian. Dalam penentuan nilai bobot pada per *term* proses

pembobotan kata dapat dilakukan dengan perhitungan formula pada persamaan (1) atau persamaan (2), tergantung pada kondisi penggunaan *dataset* yang akan proses, rumus formula yang akan diimplementasikan pada penelitian ini adalah persamaan (2) sesuai dengan *library* pemrograman yang diimplementasikan pada penelitian ini.

3.5 Klasifikasi dengan Metode SVM

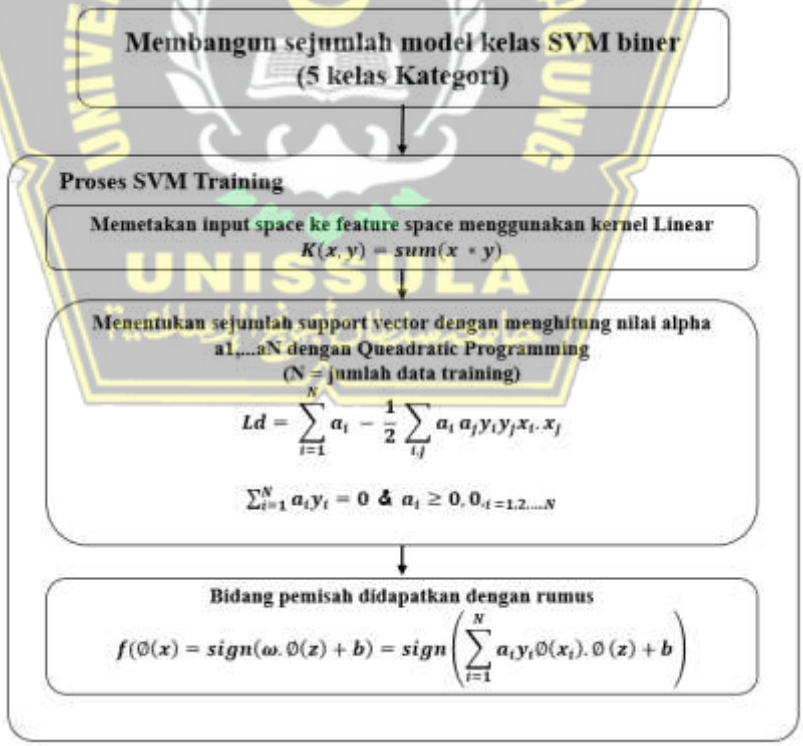
Pada penelitian, dalam pengklasifikasian jurnal penelitian berdasarkan kategori bidang fokus dengan menggunakan metode klasifikasi *Multiclass Support Vector Machine* serta mengimplementasikan tahapan-tahapan awal dengan yang perlu dilalui. Adapun *flowchart* yang ditawarkan ditunjukan pada Gambar 3.2.



Gambar 3. 2 *Flowchart* SVM OVA

Pada Gambar 3.2 diatas, *flowchart* data hasil dari proses *preprocessing* dan pembobotan kata dapat di implementasikan dengan algoritma SVM. Proses *data Training* algoritma SVM untuk menentukan vektor α (*support Vector*) serta konstanta b untuk menentukan fungsi pemisah (*Hyperplane*) dengan besar *margin* yang optimal. Untuk vektor $\alpha > 0$ disebut *support vector* serta mendeskripsikan *data Training* yang perlu sebagai fungsi keputusan yang optimal. Dan untuk konstanta b sebagai penentu dalam posisi fungsi pemisah relatif terhadap permulaan (Alita, 2021).

Selama proses *training*, perlunya menginputkan *dataset input* atau *output* dengan dokumen kelas positif dan negatif atau dalam *binary* 1 dan 0. Untuk SVM *non linier*, perlu mengimplementasikan fungsi. Fungsi *kernel* yang akan dipakai pada pelaksanaan ini yaitu fungsi *kernel Linear*. Berikut adalah diagram proses *Training* menggunakan SVM yang dapat dilihat pada Gambar 3.3.



Gambar 3. 3 Proses *Training dataset* SVM *Multiclass*

Dengan menggunakan implementasi metode SVM dengan analisis OVA (*one vs all*), pada klasifikasi kelas-k, kita mencari fungsi pemisah k, dimana k merupakan jumlah kelas. Misal terdapat fungsi pemisah yaitu ρ. Pada metode ini, data ρ di proses *training* dengan semua data misal kelas *Engineering and Technology* berlabel positif atau 1 dan untuk semua data selain kelas *Engineering and Technology* berlabel negatif -1 atau bisa dengan 0 (Null). Tahap selanjutnya yaitu menghitung suatu nilai x. Nilai x pada *table* akan berperan dalam perhitungan *kernel* atau perhitungan *dot product* dengan fungsi *kernel Linear* dengan formula pada persamaan (4).

$$\boldsymbol{K(x,y) = sum(x * y)} \text{.....(4)}$$

Selanjutnya, kumpulan data dari fitur dimensi lama harus di-*kernel* untuk mendapatkan kumpulan data dengan fitur dimensi tinggi yang baru. Dengan persamaan (4), Kumpulan data dengan dimensi Nx1 akan mendapatkan dimensi baru NxN, dimana N adalah jumlah datanya.

Point penting dalam metode ini yaitu membangu k buah model SVM binery (k merupakan jumlah kelas). Model klasifikasi masing-masing (k ke-i) dilatih serta dibandingkan dengan keseluruhan data kelas lain, untuk menemukan suatu kesimpulan. Contoh klasifikasi dengan jumlah i kelas. *Data training* digunakan sejumlah i pada SVM biner seperti pada Tabel 3.6 (Alita *et al.*, 2020)

Tabel 3. 6 SVM *Binery* Metode *One vs All*

Yi = 1	**Yi = -1**	**Hipotesis**
Kelas 1	Bukan Kelas 1	F1(x) = (w1)x + b1
Kelas 2	Bukan Kelas 2	F2(x) = (w2)x + b2
Kelas 3	Bukan Kelas 3	F3(x) = (w3)x + b3
…..	…..	…..
Kelas i	Bukan Kelas i	Fi(x) = (wi)x + bi

3.6 Perancangan Sistem

Untuk merancang model Klasifikasi Bidang Fokus Publikasi Jurnal Penelitian yang Terindeks pada Portal Garuda dengan metode *Multiclass Support Vector Machine* (SVM) pada penelitian, pihak penulis menggunakan *tools*. Selanjutnya dalam pengembangan model ini diperlukan *library-library* pendukung dalam pengembangan model ini, *library* yang di perlukan yaitu *library* nltk untuk pemrosesan data *text* seperti halnya proses *Text Processing* mulai dari tokenisasi, *stemming* dan sebagainya. Dan *library* Scikit-learn yang digunakan untuk pembobotan *term* (proses TR-IDF) serta sebagai klasifikasi dengan implementasi matode SVM.

Selanjutnya sistem pada penelitian ini dibuat berdasarkan data-data jurnal penelitian yang bersumber ataupun terindeks pada portal Garuda dan SINTA. Yang nantinya *dataset* tersebut akan dibagi menjadi *data training* dan *data testing* dengan perbandingan 7:3 sebagai *resource* pengujian penelitian ini.

3.7 Pengujian

Pengujian ini perfokus pada bagaimana tingkat akurasi yang akan didapatkan atau dihasilkan dari penerapan metode *Multiclass Support Vector Machines* (SVM) di Model penelitian ini. Pengujian pada penelitian ini menggunakan metode evaluasi *Confusion Matrix*. Metode ini digunakan sebagai tolak ukur dalam menghitung kinerja model klasifikasi *supervised learning* pada kecerdasan buatan. Dalam proses aritmatika pada *Confusion Matrix* dilakukan dengan membandingkan hasil klasifikasi model dengan hasil klasifikasi yang actual atau seharusnya. *Confusion Matrix* berupa matriks yang dapat memberikan gambaran hasil dari kinerja model klasifikasi pada *data testing*. untuk menghitung pengujian baik berupa akurasi, *precision, recall*, dan *F-1 Score* dari evaluasi kasus klasifikasi adalah dengan persamaan-persamaan berikut.

$$Accuracy = \frac{TP+TN}{TP+FN+TN+FN} \quad \text{.....(5)}$$

$$Precision = \frac{TP}{TP+FP} \quad \text{……..……..... .(6)}$$

$$Recall = \frac{TP}{TP+FN} \quad \text{…….....….(7)}$$

$$F\text{-}1\ Score = \frac{(2*Recall*Precision)}{(Recall + Precision)} \text{…(8)}$$

Tabel 3. 7 *Confusion Matrix*

		Aktual	
		1	**0**
Prediksi	**1**	**True Positive (TP)**	**False Positive (FP)**
	0	**False Negative (FN)**	**True Negative (TN)**

Pada Tabel 3.7 diatas dapat dilihat terdapat empat macam kondisi atau kejadian dari masing-masing kolom kelas yang jelaskan dibawah ini :

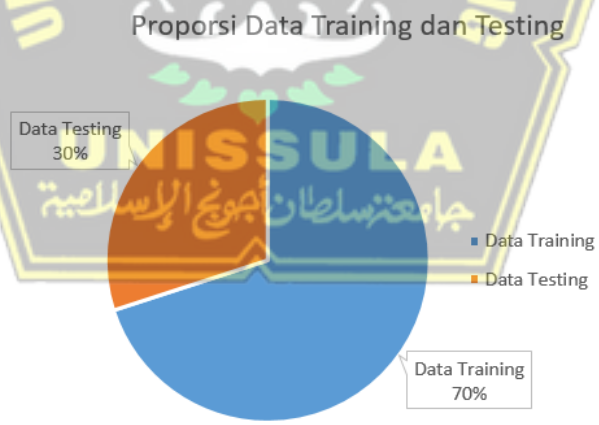
a. TP (*True Positive*) = Jumlah data *real* atau nyata yang nilainya *True* diprediksi *True*.
b. TN (*True Negative*) = Jumlah data *real* atau nyata yang nilainya *False* diprediksi *False*.
c. FP (*False Positive*) = Jumlah data *real* atau nyata yang nilainya *True* diprediksi *False*.
d. FN (*False Negative*) = Jumlah data *real* atau nyata yang nilainya *Fasle* diprediksi *True*.

BAB IV
HASIL DAN ANALISIS PENELITIAN

4.1 Pengambilan Data

Data yang diambil pada penelitian ini yaitu data jurnal portal Garuda yang terindeks pada scopus, data-data yang akan diambil berdasarkan Title jurnal beserta label kelas bidang fokus. untuk kelas kategori yang akan diimplementasikan pada penelitan terdiri dari 5 label kelas pembagian bidang fokus, diantaranya : *Arts & Humanities, Life Sciences & Medicine, Natural Sciences, Engineering & Technolog,* dan yang terakhir *Social Sciences & Management.*

Dataset yang akan diolah pada penelitian ini terbagi menjadi 2 bagian yaitu *data Training* dan *data testing,* dimana perbandingan data sebanyak 7:3 dengan jumlah keseluruhan data sebanyak 2500 data jurnal penelitian, masing-masing kategori sebanyak 500 data per kategori pada penelitian ini. Berikut proporsi data terlihat pada Gambar 4.1.



Gambar 4. 1 Proporsi *dataset*

Pembagian data dengan perbandingan sesuai dengan Gambar 4.1 di atas menghasilkan perolehan data sebanyak 750 *dataset* untuk *data testing* dan 1.750 *dataset* untuk data *training* yang nantinya *data training* tersebut akan diproses ke dalam klasifikasi.

Tabel 4. 1 Sample Dataset berdasarkan per kelas

Judul Jurnal	Label
Primate population studies at Polonnaruwa. III. Somatometric growth in a natural population of toque	Arts & Humanities
The problem of iconicity	Arts & Humanities
Automatic Tuning of PID Cascade Controllers for Servo Systems	Engineering & Technology
Polymers and oligomers with transverse aromatic groups and tightly controlled chain conformations	Engineering & Technology
Pest management of psocids in milled rice stores in the humid tropics	Life Sciences & Medicine
Negative Ion Photoelectron Spectroscopy of Coordinatively Unsaturated Group VI Metal Carbonyls: Cr(C	Life Sciences & Medicine
Multidimensional femtosecond correlation spectroscopies of electronic and vibrational excitons	Natural Sciences
Mineral and organic particles in astronomy	Natural Sciences
Optimization modeling for sewer network management	Social Sciences & Management
Some iterative techniques for general monotone variational inequalities	Social Sciences & Management

Tabel 4.1 di atas merupakan sebagian contoh dari dataset data training yang diolah pada proses klasifikasi penelitian ini, dimana dataset di atas merupakan data sample berupa title atau judul jurnal beserta label klasifikasi bidang fokus penelitian yang terdiri dari 5 jenis kelas kateogi.

4.2 *Preprocessing Data*

Preprocessing pada penelitian ini terdiri dari 5 tahap yang dimana *Prepocessing* berfungsi untuk membersihkan *dataset* dari *data noise*, menciptakan dimensi data agar menjadi lebih kecil dan terstruktur, sehingga dapat dikelola lebih lanjut pada proses selanjutnya.

	text before	text after	label
0	Higher-order approximations for frequency doma...	higher order approxim frequenc domain time ser...	Arts & Humanities
1	On the behavior of inconsistent instrumental v...	behavior inconsist instrument variabl estim	Arts & Humanities
2	Artificial decision-making and artificial ethi...	artifici decis make artifici ethic manag concern	Arts & Humanities
3	Bayesian model selection and prediction with e...	bayesian model select predict empir applic	Arts & Humanities
4	Local polynomial estimators of the volatility ...	local polynomi estim volatil function nonparam...	Arts & Humanities
...	...	...	...
2495	Representation of magnetisation curves over a ...	represent magnetis curv wide region use non in...	Social Sciences & Management
2496	Decision support system for purchasing managem...	decis support system purchas manag larg project	Social Sciences & Management
2497	Studies on peat in the coastal plains of Sumat...	studi peat coastal plain sumatra borneo part p...	Social Sciences & Management
2498	The Indonesian Transmigration Program in Histo...	indonesian transmigr program histor perspect	Social Sciences & Management
2499	Husband's approval of contraceptive use in met...	husband approv contracept use metropolitan ind...	Social Sciences & Management

Gambar 4. 2 *Dataset* setelah proses *preprocessing*

Gambar 4.2 merupakan hasil dari pengolahan pada *preprocessing data*. Dimana pada Gambar 4.2 diatas data hasil *preprocessing* hanya tersisa data penting yang bermakna, bersih dari karater tanda baca, ataupun *noise*. Pada kolom *text before* merupakan data mentah atau data sebelum mesuk ke proses *preprocessing*. Dan hasil dari proses *text processing* dapat dilihat pada kolom *text after*. Untuk lebih jelasnya berikut penguraian tahapan dalam *text processing* denga data sampel.

4.2.1 *Case Folding*

Tahapan pertama dalam *preprocessing* dimana pada tahap ini dilakukan untuk menyeragamkan karekter huruf *capital* menjadi huruf kecil dengan sampel *data training* berikut berdasakan kelas bidang fokus masing-masing.

Tabel 4. 2 Hasil *Case Folding*

Sebelum	Hasil Case Folding	Label
Primate population studies at Polonnaruwa. III. Somatometric growth in a natural population of toque	primate population studies at polonnaruwa. iii. somatometric growth in a natural population of toque	Arts & Humanities
Automatic Tuning of PID Cascade Controllers for Servo Systems	automatic tuning of pid cascade controllers for servo systems	Engineering & Technology
Negative Ion Photoelectron Spectroscopy of Coordinatively Unsaturated Group VI Metal Carbonyls: Cr(C	negative ion photoelectron spectroscopy of coordinatively unsaturated group vi metal carbonyls: cr(c)	Life Sciences & Medicine

Mineral and organic particles in astronomy	mineral and organic particles in astronomy	Natural Sciences
Optimization modeling for sewer network management	optimization modeling for sewer network management	Social Sciences & Management

4.2.2 *Text Cleaning*

Setelah data sudah seragam dengan karakter berupa huruf kecil. selanjutnya ke tahap *text cleaning* untuk membersihkan data teks dari *numbering*, karakter tanda baca yang sekiranya tidak diperlukan pada klasifikasi nantinya. Berikut contoh dari hasil tahapan *text cleaning* terlihat pada tabel 4.3. tahap text cleaning pada penelitian ini menggunakan library *Reguler expression.*

Tabel 4. 3 Hasil *Text Cleaning*

Sebelum	**Hasil Text Cleaning**	**Label**
primate population studies at polonnaruwa. iii. somatometric growth in a natural population of toque	primate population studies at polonnaruwa iii somatometric growth in a natural population of toque	Arts & Humanities
automatic tuning of pid cascade controllers for servo systems	automatic tuning of pid cascade controllers for servo systems	Engineering & Technology
negative ion photoelectron spectroscopy of coordinatively unsaturated group vi metal carbonyls: cr(c)	negative ion photoelectron spectroscopy of coordinatively unsaturated group vi metal carbonyls cr c	Life Sciences & Medicine
mineral and organic particles in astronomy	mineral and organic particles in astronomy	Natural Sciences
optimization modeling for sewer network management	optimization modeling for sewer network management	Social Sciences & Management

4.2.3 *Tokenizing*

Setelah data bersih dari adanya angka, karakter, sampai dengan noise, selanjutnya *dataset* akan dipecah menjadi per suku kata yang disebut

dengan token. Berikut contoh dari tahap tokenizing, terlihat pada Tabel 4.4 implementasi *library* nltk dengan *word_tokenize()* berperan dalam tahap tokenisasi penelitian ini.

Tabel 4. 4 Hasil *Tokenizing*

Sebelum	Hasil Tokenizing	Label
primate population studies at polonnaruwa iii somatometric growth in a natural population of toque	primate', 'population', 'studies' 'at', 'polonnaruwa', 'iii', 'somatometric', 'growth', 'in', 'a', 'natural', 'population', 'of', 'toque'	Arts & Humanities
automatic tuning of pid cascade controllers for servo systems	'automatic', 'tuning', 'of', 'pid', 'cascade', 'controllers', 'for', 'servo', 'systems'	Engineering & Technology
negative ion photoelectron spectroscopy of coordinatively unsaturated group vi metal carbonyls cr c	'negative', 'ion', 'photoelectron', 'spectroscopy', 'of', 'coordinatively', 'unsaturated', 'group',	Life Sciences & Medicine

	'vi', 'metal', 'carbonyls', 'cr', 'c'	
mineral and organic particles in astronomy	'mineral', 'and', 'organic', 'particles', 'in', 'astronomy'	Natural Sciences
optimization modeling for sewer network management	'optimization', 'modeling', 'for', 'sewer', 'network', 'management'	Social Sciences & Management

4.2.4 *Stemming Word*

Hasil dari tahap ini berupa pengubahan bentuk kata yang akan dijadikan sebagai kata dasar (*root*) untuk mempermudah dalam proses pengujian klasifikasi nantinya. Dimana tahap ini menghilangkan kata imbuhan awalan maupun akhiran dalam sub kata. Tahap ini menggunakan *library* nltk dengan *PorterStemmer*(). Berikut hasil dari proses *stemming word* terlihat pada Tabel 4.5.

Tabel 4. 5 Hasil *Stemming Word*

Sebelum	Hasil Stemming	Label
primate', 'population', 'studies' 'at', 'polonnaruwa', 'iii', 'somatometric', 'growth',	'primat', 'popul', 'studi', 'at', 'polonnaruwa', 'iii', 'somatometr', 'growth',	Arts & Humanities

'in', 'a', 'natural', 'population', 'of', 'toque'	'in', 'a', 'natur', 'popul', 'of', 'toqu'	
'automatic', 'tuning', 'of', 'pid', 'cascade', 'controllers', 'for', 'servo', 'systems'	'automat', 'tune', 'of', 'pid', 'cascad', 'control', 'for', 'servo', 'system'	Engineering & Technology
'negative', 'ion', 'photoelectron', 'spectroscopy', 'of', 'coordinatively', 'unsaturated', 'group', 'vi', 'metal', 'carbonyls', 'cr', 'c'	'negative', 'ion', 'photoelectron', 'spectroscopy', 'of', 'coordinatively', 'unsaturated', 'group', 'vi', 'metal', 'carbonyls', 'cr', 'c'	Life Sciences & Medicine
'mineral', 'and', 'organic', 'particles', 'in', 'astronomy'	'miner', 'and', 'organ', 'particl', 'in', 'astronomi'	Natural Sciences
'optimization', 'modeling', 'for',	'optim', 'model', 'for',	Social Sciences & Management

'sewer', 'network', 'management'	'sewer', 'network', 'manag'	

4.2.5 *Stopword Remover*

Selanjutnya tahap terakhir dari proses *preprocessing* pada penelitian ini yaitu menghapus atau menghilangkan kata-kata yang sekiranya tidak terlalu penting sebagai kata kunci pada proses klasifikasi penelitian ini. *Stopword remover* pada penelitian ini menggunakan *library* nltk pada *platform* python *stopwords()*.

Tabel 4. 6 Hasil *Stopword Remover*

Sebelum	**Hasil Stopword Remover**	**Label**
'primat', 'popul', 'studi', 'at', 'polonnaruwa', 'iii', 'somatometr', 'growth', 'in', 'a', 'natur', 'popul', 'of', 'toqu'	'primat', 'popul', 'studi', 'polonnaruwa', 'iii', 'somatometr', 'growth', 'natur', 'popul', 'toqu'	Arts & Humanities
'automat', 'tune', 'of', 'pid', 'cascad', 'control', 'for', 'servo', 'system'	'automat', 'tune', 'of', 'pid', 'cascad', 'control', 'for', 'servo', 'system'	Engineering & Technology

'negative', 'ion', 'photoelectron', 'spectroscopy', 'of', 'coordinatively', 'unsaturated', 'group', 'vi', 'metal', 'carbonyls', 'cr', 'c'	'negative', 'ion', 'photoelectron', 'spectroscopy', 'of', 'coordinatively', 'unsaturated', 'group', 'vi', 'metal', 'carbonyls', 'cr', 'c'	Life Sciences & Medicine
'miner', 'and', 'organ', 'particl', 'in', 'astronomi'	'miner', 'organ', 'particl', 'astronomi'	Natural Sciences
'optim', 'model', 'for', 'sewer', 'network', 'manag'	'optim', 'model', 'sewer', 'network', 'manag'	Social Sciences & Management

Dari uraian tahapan *preprocessing* diatas, dimana menjelaskan gambaran bagaimana proses *preprocessing* data-data judul jurnal penelitian yang akan di uji pada penelitian ini. kelima tahapan tersebut telah dilakukan sehingga kita mendapatkan data teks judul jurnal penelitian yang sudah bersih dan siap digunakan pada proses selanjutnya.

4.3 Pembobotan Kata dengan TF IDF

Proses pembobotan kata yang akan dieksekusi pada penelitian ini menggunakan metode pembobotan TF IDF dengan penggunaan rumus pada

persamaan 2. Karena penggunaan rumus persamaan 2 sesuai dengan *library* yang digunakan dari SKlearn yaitu *TfidfVectorizer()* untuk perhitungan pembobotan teks *dataset* judul jurnal pada penelitian ini. Berikut gambaran dari hasil bagaimana proses TF IDF terlihat pada Gambar 4.3.

	aa	ab	abat	abb	abdomin	abdullah	abil	ablat	abnorm	absent	...
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...
...	...	...	...	...	...	...	...	...	...	...	...
2495	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...
2496	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...
2497	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...
2498	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...
2499	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...

Gambar 4. 3 Hasil dari Proses TF IFD

Pada Gambar 4.3 merupakan hasil dari pemvektoran pada TF IDF dalam bentuk tabel *array*, dimana Vektorisasi merupakan proses konvert data teks menjadi data *numeric* dengan nilai angka banyaknya kemunculan kata dalam suatu dokumen. Hal ini dilakukan karena pada mesin komputer hanya dapat mengerti serta memproses data numerik, sehingga perlunya pembobotan *term*.

Tabel 4. 7 Ranking nilai bobot 10 tertinggi

***Term* (Token)**	**Nilai Bobot**
studi	106.828.205
effect	104.163.194
use	91.577.053
structur	82.559.783
model	78.804.279
system	73.491.254
complex	73.010.009
sup	72.123.016
measur	66.694.616
Analysi	62.382.193

Tabel 4.7 merupakan hasil nilai dari proses pembobotan token dengan metode TF IDF dimana hasil tersebut merupakan 10 data token tertinggi dari keseluruhan nilai bobot token. Untuk lebih jelas dalam proses pembobotan *term* dengan metode TF IDF, berikut uraian analisis perhitungan metode TF IDF dengan persamaan 2 menggunakan data *sample* yang sudah diolah pada proses *text preprocessing*.

Data judul jurnal

Data 1 (D1) : "*automat tune cascad servo system*".

Data 2 (D2) : "*energet hole transfer dna*".

Tabel 4. 8 Analisis mencari nilai TF dan df

10	**TF**		**df**	**Perhitungan**	**Hasil**
	D1	**D2**		**D/df**	**D/df**
automat	1	0	1	2/1	2
cascad	1	0	1	2/1	2
dna	0	1	1	2/1	2
energet	0	1	1	2/1	2
hole	0	1	1	2/1	2
servo	1	0	1	2/1	2
system	1	1	2	2/2	1
transfer	0	1	1	2/1	2
tune	1	0	1	2/1	2

Tahap pertama dalam menentukan nilai bobot dari suatu karakter kata (token) dimulai dengan mendeklarasikan dan mengelompokan masing-masing token dari tiap dokumen atau data serperti Pada Tabel 4.8 diatas. Untuk mencari nilai TF dari masing-masing data (D) dengan menghitung seberapa banyak token tersebut muncul di data atau dokumen tersebut. Misal untuk pada token "*automat*" dimana token tersebut hanya terdapat pada data 1 (D1) sekali, sehingga pada kolom D1 diberi nilai 1 dan untuk D2 diberi nilai 0 karena pada D2 tidak terdapat token tersebut. Hasil analisa data dilihat pada Tabel 4.8 pada kolom TF di D1 dan D2.

Untuk mencari nilai *df* dilakukan dengan cara menghitung jumlah Data (D) yang mengandung token tersebut. Misalkan pada token "*automat*"

dimana token tersebut hanya terdapat pada (D1) dan tidak terdapat pada D2 maka nilai dari *df* adalah 1. dan untuk tahap selanjutnya menghitung nilai dari hasil pembagian dari *D/df* yang mana D sendiri merupakan jumlah seluruh data atau dokuemen yang di proses. Dan *df* dari masing-masing token. Analisis dan hasil perhingan dari df dan *D/df* dapat dilihat pada Tabel 4.8 kolom *df* dan perhitungan *D/df* diatas.

Tabel 4. 9 Analisis nilai IDF dan TF IDF

Term (Token)	**D/df**	**Perhitungan**	**Hasil**	**TF IDF**	
		IDF	**IDF**	**D1**	**D2**
automat	2	LOG(2)+1	1,301029996	1 X 1,301029996	0 X 1,301029996
cascad	2	LOG(2)+1	1,301029996	1 X 1,301029996	0 X 1,301029996
dna	2	LOG(2)+1	1,301029996	0 X 1,301029996	1 X 1,301029996
energet	2	LOG(2)+1	1,301029996	0 X 1,301029996	1 X 1,301029996
hole	2	LOG(2)+1	1,301029996	0 X 1,301029996	1 X 1,301029996
servo	2	LOG(2)+1	1,301029996	1 X 1,301029996	0 X 1,301029996
system	1	LOG(1)+1	1	1 X 1	1 X 1
transfer	2	LOG(2)+1	1,301029996	0 X 1,301029996	1 X 1,301029996
tune	2	LOG(2)+1	1,301029996	1 X 1,301029996	0 X 1,301029996

Untuk mencari nilai dari IDF dilakukan degan mencari nilai logaritma dari *D/df* dan ditambahkan nilai 1. Analisis perhitungan dapat dilihat pada tabel 4.9 kolom perhitungan IDF diatas dan untuk hasil dapat dilihat pada bagian Hasil IDF. Kemuadian tahap terakhir untuk mencari nilai bobot token dengan TF IDF dimana nilai dari TF dari masing-masing document dan token dikalikan dengan nilai dari IDF. Misalkan pada token "*automat*" dimana hanya muncul atau terdapat pada D1 dengan nilai 1 dikalikan dengan nilai pada IDF token tersebut dan untuk D2 dengan nilai 0 sama dikalikan dengan dengan hasil IDF sehingga hasil dari perhitungan tersebut merupakan nilai bobot dari masing-masing token. Untuk analisa perhitungan lebih jelas dapat

dilihat pada Tabel 4.9 diatas. Dan untuk hasil dari analisa tersebut secara lengkap dapat dilihat pada Tabel 4.10.

Tabel 4. 10 Analisis hasil dari TF IDF

Term (Token)	TF		df	D/df	IDF	TF IDF	
	D1	D2				D1	D2
automat	1	0	1	2	1,301029996	1,301029996	0
cascad	1	0	1	2	1,301029996	1,301029996	0
dna	0	1	1	2	1,301029996	0	1,301029996
energet	0	1	1	2	1,301029996	0	1,301029996
hole	0	1	1	2	1,301029996	0	1,301029996
servo	1	0	1	2	1,301029996	1,301029996	0
system	1	1	2	1	1	1	1
transfer	0	1	1	2	1,301029996	0	1,301029996
tune	1	0	1	2	1,301029996	1,301029996	0

Pada tabel 4.10, merupakan hasil analisa dari perhitungan pada proses metode TF IDF dengan rumus persamaan 2. Hasil perolehan nilai bobot tersebut dapat berubah-ubah berdasarkan pada jumlah data atau dokumen (D) yang digunakan dan token kemunculan dari masing-masing data.

Untuk penjelasan *code* program dari perhitungan pembobotan nilai dengan metode TF IDF, dapat dilakukan menggunakan *library Scikit Learn* pada platform pemrograman python, berikut *code program* dalam mencari nilai TF IDF :

```
from sklearn.feature_extraction.text import
TfidfVectorizer

tfidf_vectorizer = TfidfVectorizer().fit(text)
tfidf_text = tfidf_vectorizer.transform(text)
```

Penjelasan dari *pseudo code* diatas, dimana pada baris pertama untuk memanggil *library Scikit Learn* dengan meng-*import* fungsi

`TfidfVectorizer()` dimana berfungsi untuk menghitung nilai bobot dari masing-masing token pada *dataset*. Dataset dari keseluruhan data yang diuji disimpat pada *variable* `text` terdiri dari data judul jurnal yang sudah melalui proses *preprocessing data*.

4.4 Klasifikasi Metode SVM

Proses paling utama dalam penelitian ini adalah klasifikasi dengan implementasi metode *Support Vector Machines* SVM dengan fokus pembahasan. Hasil *dataset* dari proses-proses sebelumnya seperti *Preprocessing* data dan pembobotan *Term*, data-data tersebut akan diproses pada tahapan ini. Implementasi pada penelitian ini menggunakan *library* SVM pada *Sklearn Python* dengan metode *multiclass classification* SVM OVA (*One vs All*). Hal ini karena penggunaan kelas lebih dari 2 (*Binery*) yang terdiri dari 5 kelas, Gambaran SVM *multiclass* pada klasifikasi bidang fokus penelitian metode SVM OVA (*One vs All*) dapat dilihat pada Tabel 4.11.

Tabel 4. 11 Gambaran One vs All pada Klasifikasi Jurnal

	Kelas 1	**Kelas -1**
Binary Classification 1	*Engineering and Technology*	*Arts & Humanities, Life Sciences & Medicine, Natural Sciences, Social Sciences & Management*
Binary Classification 2	*Arts & Humanities*	*Engineering and Technology, Life Sciences & Medicine, Natural Sciences, Social Sciences & Management*
Binary Classification 3	*Life Sciences & Medicine*	*Engineering and Technology, Arts & Humanities, Natural Sciences, Social Sciences & Management*
Binary Classification 4	*Natural Sciences*	*Engineering and Technology, Arts & Humanities, Life Sciences & Medicine, Social Sciences & Management*
Binary Classification 5	*Social Sciences & Management*	*Engineering and Technology, Arts & Humanities, Life Sciences & Medicine, Natural Sciences*

Berdasarkan pada tabel 4.11 diatas, diperlukannya 5 *binary* klasifikasi dimana misal pada *binary classification* 1 untuk menyaring kelas bidang fokus "*Engineering and Technology*" perluya pemberian label berupa angka 1 (positif) dan untuk kelas yang lain selain dari kelas "*Engineering and Technology*" akan diberi label dengan 0 atau -1 (negatif).

Untuk penyaringan kelas pada binary classification 2 yaitu kelas "*Arts & Humanities*". Kelas dengan label positif diberikan kepada kelas "*Arts & Humanities*" selanjutnya dibandingkan dengan kelas selain "*Arts & Humanities*" yaitu "*Engineering and Technology, Life Sciences & Medicine, Natural Sciences, Social Sciences & Management*" yang nantinya berlabel *negative*.

Sedangkan untuk *binary classification* kelas selanjutnya, yaitu untuk 3 nilai label positif diberikan kepada kelas "*Life Sciences & Medicine*" dan untuk nilai label negative diberikan kepada kelas selain dari kelas "*Life Sciences & Medicine*". Dan seterusnya untuk *binary classification* 4 dan 5.

Dari uraian diatas dapat disimpulkan dalam klasifikasi *Multiclass* SVM dengan metode *One Vs All*. Kondisi membandingkan kelas utama dengan kelas lainnya secara bersamaan. Pendekatan pada penelitian ini dapat dilakukan dengan secara manual untuk mengetehui akurasi setiap kelas dengan SVM *Binary* ataupun dengan implementasi dilakukan dengan *library* pada *library* SVM pada Sklearn Python *OneVsRestClassifier()* dengan nilai akurasi keseluruhan.

4.4.1 Perancangan Model Klasifikasi *Multiclass* SVM metode OVA

Untuk merancang model Klasifikasi Bidang Fokus Publikasi Jurnal Penelitian yang Terindeks pada Portal Garuda dengan algoritma *Multiclass Support Vector Machine* (SVM) dengan metode OVA (*One vs All*) pada penelitian ini, penulis menggunakan *tools* bahasa pemrograman python dengan memanfaatkan *library* untuk pengolahan *machine learning* yaitu Scikit-learn. *Dataset* yang nantinya akan diolah terbagi menjadi 2 yaitu *dataset* terdiri dari x dan y dalam bentuk $\{(x1, y1), \ldots, (xn, yn)\}$ dimana x

disebut dengan *vector* dan y adalah label kelasnya. Berikut *code program* dari untuk deklarasi dataset yang akan diimplementasikan.

a. Pembagian *dataset*

```
from sklearn.model_selection import
train_test_split

X_train, X_test, y_train, y_test =
train_test_split(tfidf_text, label, test_size = 0.30,
random_state = 123)
```

Untuk data yang akan diolah sebelumnya tersimpan pada *variable* tfidf_text *dan* label yang terdapat *dataset* judul jurnal penelitian beserta pelabelan dari data tersebut. Baris pertama pada program yaitu untuk *import* fungsi dari *library* Scikit-learn yaitu *train_test-split()*. Fungsi *train_test-split()* untuk membagi seluruh *dataset* menjadi 2 yaitu *data training* dan *data testing* dimana tersimpan pada *variable code :* `X_train, X_test, y_train, y_test` *data training* berada pada variable `X_train, y_train` dan data testing disimpan pada *variable* `X_test, y_test`.

Berikut *code* untuk `train_test_split(tfidf_text, label, test_size = 0.30, random_state = 123)` dalam pemisahan *dataset* dimana *tfidf_text* menyimpan data judul jurnal penelitian dan label menyimpan data label atau kelas jurnal tersebut. `test_size = 0.30` untuk membagi *dataset* tersebut dimana untuk data testing 0,3 atau 30% yaitu 750 dataset dan data training sisanya 70% yaitu sebanyak 1.750 *dataset*. Dan `random_state = 123` sebagai parameter pembagian data secara *random*.

b. Model klasifikasi metode

Setelah didapatkan dataset dari proses pembagian sebelumnya selanjutnya ambil dataset training untuk penetapan model SVM yang

dimana proses ini hanya memakai data training. Berikut *code program* dalam pembuatan model klasifikasi.

```
from sklearn.svm import LinearSVC
from sklearn.multiclass import OneVsRestClassifier
```

Perlunya melakukan *import* fungsi pada *library* yang digunakan untuk mengoperasikan pembuatan model ini. Yang pertama `LinearSVC` merupakan fungsi untuk penetapan klasifikasi *support vector* atau lebih dikenal dengan SVC pada metode SVM yang mengimplementasikan kernel Linier. `OneVsRestClassifier` Berfungsi untuk proses klasifikasi dengan implementasi metode One vs All (OVA).

```
svm = LinearSVC(random_state=none)
ovr_classifier = OneVsRestClassifier(svm)
model = ovr_classifier.fit(X_train, y_train)
```

selanjutnya pendeklarasian objek klasifikasi model SVM yang menggunakan parameter kernel=linear terlihat pada *code program* `svm = LinearSVC(random_state=none)` dimana deklarasi tersebut ditampung atau dinamakan dengan variable svm. `ovr_classifier` Merupakan pendeklarasian untuk implementasi fungsi dari metode OVA yaitu `OneVsRestClassifier(svm)`. Dan selanjutnya untuk model klasifikasi dilakukan pada dengan mengoperasikan fungsi variable `ovr_classifier.fit()` dimana data yang digunakan yaitu `X_train` untuk judul jurnal dan `y_train` untuk pelabelan jurnal tersebut. Sehingga dari proses tersebut didapatkan suatu model klasifikasi yang siap untuk diuji keakuratannya .

c. Pengujian akurasi model

Untuk mengetahui tingkat keberhasilan dari model yang sudah dibangun dalam penelitian ini, dapat dilakukan dengan mengoutputkan hasil akurasi dari penelitian ini, ouput yang dapat dilihat dari keberhasilan model yang dibangun dalam penelitian ini yaitu akurasi, *precission*, *recall*, dan F-1 *Score*.

Dalam menguji suatu model yang sudah dibangun dari model klasifikasi berikut, dapat dilakukan dengan *code program* `classification_report()` yang dimana berfungsi untuk menampilkan laporan pengujian dari model klasifikasi. Untuk data yang digunakan dalam pengujian model ini menggunakan data testing yaitu `X_test` sehingga code program terlihat seperti berikut ini.

```
pred_svm = model.predict(X_test)
print(classification_report(pred_svm, y_test))
```

Pred_svm adalah variable yang mengeksekusi model klasifikasi yang menguji atau memprediksi data dari *dataset* `X_test`. Dengan menjalankan fungsi `predict(X_test)`. Selanjutnya untuk menampilkan hasil dari pengujian model klasifikasi dilakukan dengan code program `classification_report(pred_svm, y_test))` yang dimana fungsi tersebut membutuhkan parameter `pred_svm` sebagai variable eksekusi dari model klasifikasi dan `y_test` sebagai label dari data testing dataset yang diuji. Hasil dari eksekusi *code program* tersebut dapat dilihat pada Gambar 4.4 berikut.

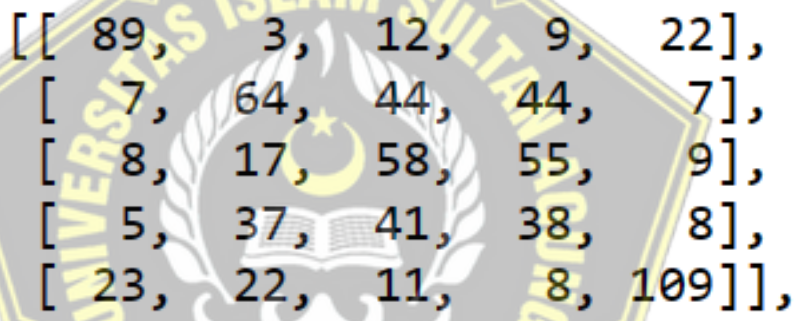
	precision	recall	f1-score	support
Arts & Humanities	0.67	0.66	0.67	135
Engineering & Technology	0.45	0.39	0.41	166
Life Sciences & Medicine	0.35	0.39	0.37	147
Natural Sciences	0.25	0.29	0.27	129
Social Sciences & Management	0.70	0.63	0.66	173
accuracy			0.48	750
macro avg	0.48	0.47	0.48	750
weighted avg	0.49	0.48	0.48	750

Gambar 4. 4 Hasil uji model klasifikasi OVA

Pada Gambar 4.4 diatas adalah hasil dari implementasi uji pada model klasifikasi yang dimana pada gambar tersebut, dapat dilihat seberapa besar tingkat akurasi yang didapatkan pada pengujian tersebut yaitu sebesar 0,48 , dan seberapa besar nilai dari *precission, recall*, dan F-1 *Score* dari masing-masing kelas.

4.4.2 Pengujian dan Evaluasi Model

Tingkat keberhasilan dalam implementasi dari hasil pengujian pada metode SVM OVA klasifikasi jurnal penelitian dapat di uji dengan menggunakan metode evaluasi *Confusion Matrix*. Metode ini sendiri sebagai tolak ukur dalam kinerja model klasifikasi *supervised learning* pada kecerdasan buatan. Dalam aritmatika *Confusion Matrix* dilakukan dengan membandingkan hasil dari proses klasifikasi pada model dengan *output* klasifikasi yang *real* atau nyata atau seharusnya. *Confusion Matrix* berupa matriks yang dapat memberikan gambaran hasil dari kinerja model klasifikasi pada *data testing*. Berikut hasil dari *confusion matrix* hasil klasifikasi pada keseluruhan kelas kategori penelitian ini.



Gambar 4. 5 Hasil Confusion Matrix klasifikasi

Pada Gambar 4.5 merupakan hasil dari pengolahan data uji (*Testing*) klasifikasi pada penelitian ini implementasi alogritma *Multiclass* SVM dengan metode *One Vs All* dengan keseluruhan kelas berupa data *Matrix Confusion*. Berikut Tabel penjabaran dari matrix diatas.

Tabel 4. 12 Tabel data *confusion* matriks *All*

	Prediksi				
Aktual	**AH**	**ET**	**LM**	**NS**	**SM**
Arts & Humanities (AH)	89	3	12	9	22
Engineering & Technology (ET)	7	64	44	44	7
Life Sciences & Medicine (LM)	8	17	58	55	9
Natural Sciences (NS)	5	37	41	38	8
Social Sciences & Management (SM)	23	22	11	8	109

a. Mencari nilai akurasi :

Untuk mencari nilai dari akurasi pada hasil pengujian klasifikasi penelitian ini, dapat dilakukan menghitung dengan rumus persamaan 5. Berikut implementasi dari perhitungan tersebut.

$$Accuracy = \frac{89+67+58+38+109}{750\ (Jml\ Data)}$$

$$Accuracy = \frac{358}{750} = 0{,}477333333\ (48\ \%)$$

Dihasilkan nilai akurasi sebesar 48%

b. Mencari nilai *precission* :

Untuk mencari nilai precission dari masing-masing kelas pengujian model klasifikasi pada penelitian ini dilakukan dengan mengimplementasi rumus pada persamaan 6. Berikut analisis implementasi tersebut terlihat pada Tabel 4.13.

Tabel 4. 13 Hitung *Precission* kelas

	AH	**ET**	**LM**	**NS**	**SM**
TP	89	67	58	38	109
FP	7+8+5+23 = 43	3+17+37+22 = 79	12+44+41+11 = 108	9+44+55+8 = 116	22+7+9+8 = 46
Precission	89/(89+43) = 0,674242424	64/(64+79) = 0,447552448	58/(58+108) = 0,34939759	38/(38+116) = 0,246753247	109/(109+46) = 0,703225806
persentasi	67%	45%	35%	25%	70%

Dari hasil perhitungan pada Tabel diatas dihasilkan perolehan nilai presentasi precission dari setiap kelas kategori yang diuji pada penelitian ini dengan nilai presentasi yang variatif.

c. Mencari nilai *Recall* :

Untuk mencari nilai *Recall* dari masing-masing kelas pengujian model klasifikasi penelitian ini dilakukan dengan mengimplementasi rumus pada persamaan 7. Berikut analisis implementasi tersebut terlihat pada Tabel 4.15.

Tabel 4. 14 Hitung *Recall* kelas

	AH	ET	LM	NS	SM
TP	89	67	58	38	109
FP	3+12+9+22 = 46	7+44+44+7 = 102	8+17+55+9 = 89	5+37+41+8 = 91	23+22+11+8 = 64
Recall	89/(89+46) = 0,659259259	67/(67+102) = 0,385542169	58/(58+89) = 0,394557823	38/(38+91) = 0,294573643	109/(109+64) = 0,630057803
persentasi	66%	39%	39%	29%	63%

Dari hasil perhitungan pada Tabel diatas dihasilkan perolehan nilai presentasi *Recall* dari setiap kelas kategori yang diuji pada penelitian ini dengan nilai presentasi yang variatif.

d. Mencari nilai *F-1 Score* :

Untuk menentukan nilai dari F-1 Score pada setiap kelas pengujian model klasifikasi penelitian ini dapat dilakukan dengan menghitung rumus persamaan 8 dengan implementasi seperti pada aritmatika berikut.

Tabel 4. 15 Mencari nilai *F-1 Score*

	AH	ET	LM	NS	SM
Precission	0,674242	0,447552	0,349398	0,246753	0,703226
Recall	0,659259	0,385542	0,394558	0,294574	0,630058

$$F\text{-}1\ Score\ (AH) = \frac{2X(0{,}674242\ \text{X}\ 0{,}659259)}{(0{,}674242 + 0{,}659259)} = 0{,}666666667\ (67\%)$$

$$F\text{-}1\ Score\ (ET) = \frac{2X(0{,}447552\ \text{X}\ 0{,}385542)}{(0{,}447552 + 0{,}385542)} = 0{,}414239482\ (41\%)$$

$$F\text{-}1\ Score\ (LM) = \frac{2X(0{,}349398\ \text{X}\ 0{,}394558)}{(0{,}349398 + 0{,}394558)} = 0{,}370607029\ (37\%)$$

$$F\text{-}1\ Score\ (NS) = \frac{2X(0{,}246753\ \text{X}\ 0{,}294574)}{(0{,}246753 + 0{,}294574)} = 0{,}268551237\ (27\%)$$

$$F\text{-}1\ Score\ (SM) = \frac{2X(0{,}703226\ \text{X}\ 0{,}630058)}{(0{,}703226 + 0{,}630058)} = 0{,}664634146\ (66\%)$$

Untuk mencari besar nilai akurasi dari masing-masing kelas kategori bidang fokus penelitian berdasarkan *binary clasifications* bidang fokus pada tabel 4.11 diatas, dapat dicari dengan mengimplementasikan metode *Confusion Matriks* berikut perhitungan dari msaing-masing *binary clasifications*.

a. *Binary Clasifications* 1 (*Engineering and Technology*)

```
array([[600, 137],
       [  7,   6]], dtype=int64)
```

Gambar 4. 6 *Matrix confusion binary clasifications I*

Gambar 4.6 merupakan hasil *matrix confusion* dari hasil dari pengolahan data uji (*Testing*) *binary clasifications I* untuk kelas *Engineering and Technology* klasifikasi yang disajikan berupa data *Matrix Confusion*. Berikut Tabel 4.16 penjabaran dari *matrix*.

Tabel 4. 16 *Confusion Matrix binary clasifications I*

	OK	**ET**
Others Kelas (OK)	600	137
Engineering and Technology (ET)	7	6

Untuk mencari nilai akurasi dapat mengimplementasikan rumus persamaan 5 berikut aritmatika dari hasil confusion matrix tersebut.

$$Accuracy = \frac{600+6}{750\ (Jml\ Data)}$$

$$Accuracy = \frac{606}{750} = 0{,}808\ (81\ \%)$$

Dihasilkan nilai akurasi dari kelas *Engineering and Technology (ET)* sebesar 81%

b. *Binary Clasifications* 2 (*Arts & Humanities*)

```
array([[617, 131],
       [  1,   1]], dtype=int64)
```

Gambar 4. 7 *Matrix confusion binary clasifications II*

Gambar 4.7 merupakan hasil *matrix confusion* dari hasil dari pengolahan data uji (*Testing*) *binary clasifications II* untuk kelas *Arts & Humanities* klasifikasi yang disajikan berupa data *Matrix Confusion*. Berikut Tabel penjabaran dari *matri*.

Tabel 4. 17 *Confusion Matrix binary clasifications II*

	OK	**AH**
Others Kelas (OK)	617	131
Arts and Humanities (AH)	1	1

Untuk mencari nilai akurasi dapat mengimplementasikan rumus persamaan 5 berikut aritmatika dari hasil *confusion matrix* tersebut.

$$Accuracy = \frac{617+1}{750\ (Jml\ Data)}$$

$$Accuracy = \frac{618}{750} = 0{,}824\ (82\ \%)$$

Dihasilkan nilai akurasi dari kelas *Arts and Humanities (AH)* sebesar 81%

c. *Binary Clasifications* 3 (*Life Sciences & Medicine*)

```
array([[580, 164],
       [  4,   2]], dtype=int64)
```

Gambar 4. 8 *Matrix confusion binary clasifications III*

Gambar 4.8 merupakan hasil *matrix confusion* dari hasil dari pengolahan data uji (*Testing*) *binary clasifications III* untuk kelas *Life Sciences & Medicine* klasifikasi yang disajikan berupa data *Matrix Confusion*. Berikut Tabel penjabaran dari *matrix*.

Tabel 4. 18 *Confusion Matrix binary clasifications II*

	OK	**LS**
Others Kelas (OK)	580	164
Life Sciences & Medicine (LS)	4	2

Untuk mencari nilai akurasi dapat mengimplementasikan rumus persamaan 5 berikut aritmatika dari hasil *confusion matrix* tersebut.

$$Accuracy = \frac{580+2}{750\ (Jml\ Data)}$$

$$Accuracy = \frac{582}{750} = 0{,}776\ (78\ \%)$$

Dihasilkan nilai akurasi dari kelas *Life Sciences & Medicine (LS)* sebesar 78%

d. Binary Clasifications 4 (*Natural Sciences*)

```
array([[586, 153],
       [ 10,   1]], dtype=int64)
```

Gambar 4. 9 *Matrix confusion binary clasifications IV*

Gambar 4.9 merupakan hasil *matrix confusion* dari hasil dari pengolahan data uji (*Testing*) *binary clasifications IV* untuk kelas *Natural Sciences* klasifikasi yang disajikan berupa data *Matrix Confusion*. Berikut Tabel penjabaran dari *matrix*.

Tabel 4. 19 *Confusion Matrix binary clasifications IV*

	OK	**NS**
Others Kelas (OK)	586	153
Natural Sciences (NS)	10	1

Untuk mencari nilai akurasi dapat mengimplementasikan rumus persamaan 5 berikut aritmatika dari hasil *confusion matrix* tersebut.

$$Accuracy = \frac{586+1}{750\ (Jml\ Data)}$$

$$Accuracy = \frac{587}{750} = 0{,}782666667\ (78\ \%)$$

Dihasilkan nilai akurasi dari kelas *Life Sciences & Medicine (LS)* sebesar 78%

e. *Binary Clasifications V* (*Social Sciences & Management*)

```
array([[595, 155],
       [  0,   0]], dtype=int64)
```

Gambar 4. 10 *Matrix confusion binary clasifications V*

Gambar 4.10 merupakan hasil *matrix confusion* dari hasil dari pengolahan data uji (*Testing*) *binary clasifications V* untuk kelas *Social Sciences & Management (SM)* klasifikasi yang disajikan berupa data *Matrix Confusion*. Berikut Tabel penjabaran dari *matrix*.

Tabel 4. 20 *Confusion Matrix binary clasifications V*

	OK	**NS**
Others Kelas (OK)	595	155
Social Sciences & Management (SM)	0	0

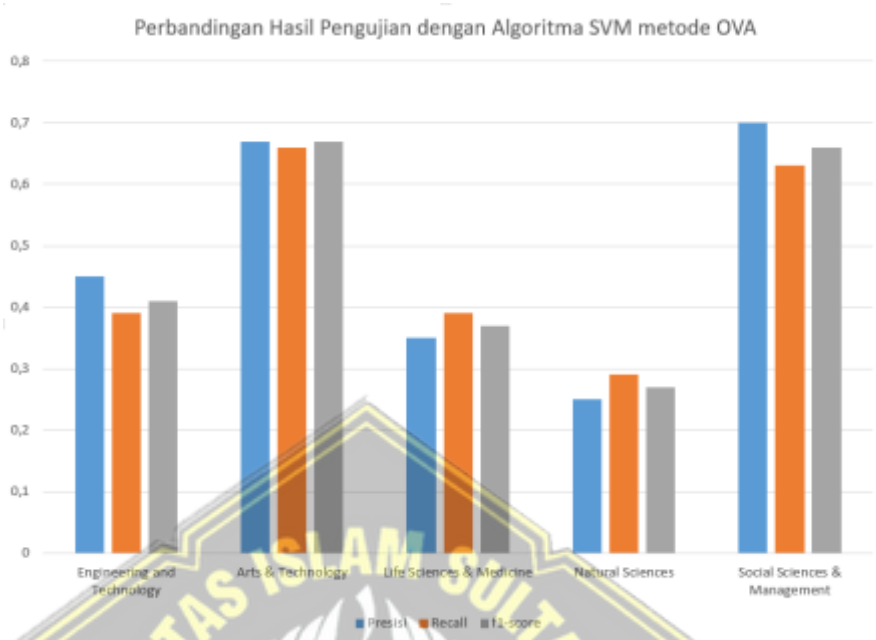
Untuk mencari nilai akurasi dapat mengimplementasikan rumus persamaan 5 berikut aritmatika dari hasil *confusion matrix* tersebut.

$$Accuracy = \frac{595}{750\ (Jml\ Data)}$$

$$Accuracy = \frac{595}{750} = 0{,}793333333\ (79\ \%)$$

Dihasilkan nilai akurasi dari kelas *Social Sciences & Management (SM)* sebesar 79%

4.4.3 Hasil Pengujian Klasifikasi *Multiclass* SVM metode OVA



Gambar 4. 11 Perbandingan Hasil pengujian

Pada Gambar 4.11, dapat kita lihat hasil perolehan nilai dari masing-masing kelas kategori dengan menggunakan metode SVM OVA implementasi dari *library* Sklearn Python *OneVsRestClassifier*(). Untuk perolehan hasil nilai presisi tertinggi terletak pada kelas "*Social Sciences & Management*" dengan perolehan persentase sebesar 70% dan untuk nilai presisi terkecil terletak pada kelas "*Natural Sciences*" dengan perolehan presentase sebesar 25%.

Untuk persentase pada nilai recall yang tertinggi diperoleh pada kelas "Art & Humanities" dengan persentase sebanyak 66% dan untuk perolehan nilai persentase paling sedikit terdapat pada kelas "Natural Sciences" dengan perolehan nilai persentase sebanyak 29%.

Dan terakhir untuk perolehan persentase f1-score. Hasil nilai persentase pengujian terbesar terletak pada kelas "Art & Humanities" dengan perolehan nilai sebanyak 67% selisih sedikit dengan kelas "*Social Sciences & Management*" dengan perolehan nilai persentase sebanyak 66%. Dan untuk hasil nilai f1-score terkecil terletak pada kelas "Natural Sciences" dengan perolehan nilai sebanyak 27%.

Selanjutnya untuk perolehan nilai Untuk nilai hasil perolehan lebih lengkapnya ditampilkan pada tabel 4.12.

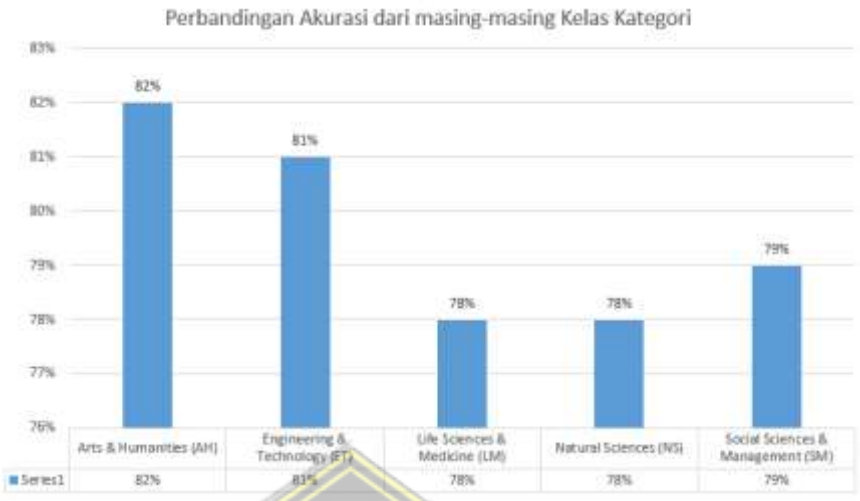
Tabel 4. 21 Hasil pengujian SVM OVA

Bidang Fokus Penelitian	**Presisi**	**Recall**	**f1-score**	**Support**
Engineering and Technology	0,45	0,39	0,41	166
Arts & Technology	0,67	0,66	0,67	135
Life Sciences & Medicine	0,35	0,39	0,37	147
Natural Sciences	0,25	0,29	0,27	129
Social Sciences & Management	0,7	0,63	0,66	173

Perolehan nilai akurasi dari pengujian metode SVM OVA dengan implementasi dari *library* Sklearn Python *OneVsRestClassifier*() berikut. Hasil perolehan hanya mencakup nilai akurasi sebanyak 48%. Hal ini terjadi karena dapat penggunaan dataset berupa tittle atau judul jurnal yang kurang akurat dan masih terbilang ambigu.

4.4.4 Hasil Perbandingan Akurasi dari Setiap Kelas

Untuk lebih jelasnya perhal perolehan nilai akurasi dari masing-masing kelas dengan mencoba pengujian dengan SVM kondisi *biner* berdasarkan *binary clasification* seperti pada Tabel 4.7. Misalkan untuk kelas utama yaitu kelas bidang fokus "*Engineering and Technology*" perluya pemberian label berupa angka 1 (positif) dan untuk kelas yang lain selain dari kelas "*Engineering and Technology*" seperti "*Arts & Humanities, Life Sciences & Medicine, Natural Sciences, Social Sciences & Management*" akan diberi label dengan 0 (negatif) Dengan pengujian tersebut berdasarkan masing-masing kelas dari berikut hasil perolehan nilai terpapar pada Gambar 4.12.



Gambar 4. 12 perbandingan akurasi per kategori kelas

Pada gambar 4.12 diatas, hasil dari pengujian dengan menggunakan metode SVM biner dengan pembagian kelas kategori berdasarkan pada kondisi Tabel 4.7 dimana dari hasil pengujian tersebut pada kita lihat, hasil akurasi tertinggi yaitu pada kelas "*Art & Humanities*" dengan perolehan nilai akurasi sebesar 82%, diikuti dengan kelas "*Engineering & Technology*" dengan perolehan nilai akurasi sebesar 81%, dan seterusnya sampai pada perolehan akurasi terkecil terdapat pada kelas "*Life Sciences & Medicine*" serta kelas "*Natural Sciences"* dengan perolehan nilai akurasi sebanyak 78%.

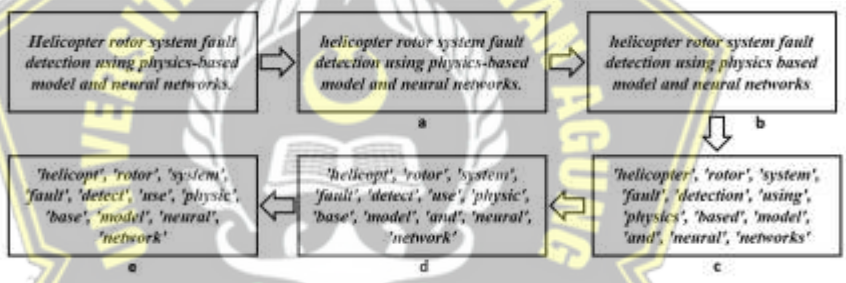
4.4.5 Analisis Prediksi Hasil Klasifikasi

Dari hasil pengujian klasifikasi data Jurnal Penelitian derdasarkan kelas Bidang Fokus penelitian diatas. Telah diperoleh modul klasifikasi yang dapat memprediksi klasifikasi jurnal-jurnal penelitian yang baru (*Up to Date*) yang sekiranya belum terklasifikasi. Maka dari itu modul berikut dapat berperan membantu dalam proses pengklasifikasian jurnal. Dalam menentukan perdiksi dari data jurnal baru dilakukan dengan fungsi *code program* berikut :

```
def prediction(text):
      text = text_preprocessing(text)
      full_text = " ".join(text)
```

```
        tfidf_vector = tfidf_vectorizer.transform([full_text])
        pred = joblib_model.predict(tfidf_vector)
return pred[0]
```

berikut adalah deklarasi *code program* fungsi `prediction()` yang digunakan dalam memprediksi jurnal baru yang nantinya klasifikasi termasuk dalam bidang fokus penelitian yang sesuai. Pertama untuk *data input* merupakan judul Jurnal baru yang akan diprediksi dimana *data input* disimpan pada *variable* `text`. Selanjutnya *data input* diproses pada *text preprocessing* dengan memanggil fungsi `text_preprocessing()`. Pada proses ini *data input* akan dibersihkan dari karakter, *noise*, ataupun teks yang tidak perlu dengan melalui beberapa tahapan seperti yang dibahas pada proses *preprocessing* sebelumnya. Berikut hasil dari *text preprocessing* terlihat pada Gambar 4.13.



Gambar 4. 13 Hasil tahapan *text processing* teks

Tahap selanjutnya yaitu pemberian nilai bobot dari data teks atau token dengan metode TF IDF melalui eksekusi pada *code program* `tfidf_vectorizer.transform([full_text])` dimana Fungsi pada `tfidf_vectorizer` adalah pendeklarasian dari fungsi sebelumnya yaitu `TfidfVectorizer()` untuk menentukan nilai TF IDF dari setiap token. Dan untuk *code* `transform` berfungsi untuk proses normalisasi nilai pembobotan dari hasil perhitungan TF IDF. Sehingga diperoleh hasil seperti pada Tabel 4.22.

Tabel 4. 22 Hasil tahapan bobot kata teks

Token	Bobot
base	0.301511
detect	0.301511

fault	0.301511
helicopt	0.301511
model	0.301511
network	0.301511
neural	0.301511
physic	0.301511
rotor	0.301511
system	0.301511
use	0.301511

Setelah didapatkan nilai dari masing-masing token, data langsung diuji dengan model klasifikasi dimana dataset menjadi *data testing* yang disimpan pada *variable* `tfidf_vector`. Pada code program `joblib_model.predict(tfidf_vector)` adalah eksekusi dari proses klasifikasi dimana *code* `joblib_model` merupakan model hasil dari proses klasifikasi *data training* sebelumnya dan untuk `predict(tfidf_vector)` berfungsi untuk mencari ataupun uji prediksi dengan mencocokan klasifikasi pada model dengan memasukan parameter *variable* `tfidf_vector` *data input* hasil dari proses pembobotan term sebelumnya. Sehingga dari hasil prediksi klasifikasi input jurnal didapatkan hasil seperti pada Gambar 4.14.

```
text = "Helicopter rotor system fault detection using physics-based model"

prediction(text)
```

```
array(['Engineering & Technology'], dtype='<U28')
```

Gambar 4. 14 *code program* panggil fugsi *prediction*

Pada Gambar 4.14 diatas, merupakan hasil dari prediksi dimana *input data* ditampung pada *variable text* dan *data input* tersebut sebagai parameter dalam eksekusi pada fungsi `prdiction()` untuk proses uji mencari hasil prediksi model klasifikasi tersebut dengan hasil *ouput* berupa *array* dari jenis bidang fokus penelitian berdasarkan prediksi klasifikasi model penelitian ini.

Adapun pengujian yang dilakukan dengan mengecek jurnal-jurnal dengan membandingkan pada modul penelitian ini. Yang nantinya *output*

hasil uji prediksi keakuratan bidang fokus jurnal. Hasil dari pengujian dapat dilihat pada tabel 4.23.

Tabel 4. 23 Data Hasil Prediksi

Judul Penelitian	**Label Bidang Fokus**	**Kesimpulan**
A paradigm of Islamic money and banking	Social Sciences & Management	Sesuai
Pricing credit risk of asset-backed securitization bonds in Singapore	Social Sciences & Management	Sesuai
Customers adoption of banking channels in Hong Kong	Social Sciences & Management	Sesuai
Variable structure control of rigid-flexible closed-chain systems	*Engineering and Technology*	Sesuai
A fuzzy clustering approach to manufacturing cell formation	*Engineering and Technology*	Sesuai
Electro- and thermophysical properties of polyaniline	*Engineering and Technology*	Sesuai
Synthesis and Structures of Titanaoxacyclobutanes	Natural Sciences	Sesuai
Particle scattering in nonassociative quantum field theory	Natural Sciences	Sesuai
etailed balance and entropy for atoms in squeezed vacuum	Natural Sciences	Sesuai
The Perils of Prosperity Approach in Papua	Arts & Humanities	Sesuai
Muslim cultural identity and attitude change among tolakinese comunity in Kendari	Arts & Humanities	Sesuai
Apple shrinkage upon drying	Life Sciences & Medicine	Sesuai
Metabolic activation of styrene by erythrocytes detected as increased sister chromatid exchanges in	Life Sciences & Medicine	Sesuai

Pada tabel 4.23 merupakan data-data jurnal yang sudah melalui uji diprediksi dari model pada penelitian ini. Data jurnal yang dipakai dipilih

secara acak dengan pembuktian hasil prediksi akurat tidaknya disertai dengan label bidang fokus penelitian. Hasil dari pengujian dirasa cukup tepat dalam memprediksi kelas bidang fokus penelitian. Namun disamping itu kemungkinan munculnya hasil yang tidak sesuai juga bisa terjadi karena tingkat akurasi perolahan dalam uji klasifikasi juga termasuk cukup rendah.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah melakukan analisa, pembuatan model, serta implementasi pada klasifikasi bidang fokus publikasi penelitian dengan metode *Multiclass Support Vector Machines* (SVM), maka dapat diambil kesimpulan sebagai berikut :

1. Dengan membangun suatu model klasifikasi yang mengimplementasi *Text Mining* dengan Algoritma *Multiclass Support Vector Machines (SVM)* dapat melakukan pengujian terhadap klasifikasi bidang fokus penelitian terutama Jurnal Penelitian yang bekerja dengan baik.
2. Hasil dari penelitian ini menunjukan bahwa penelitian ini berhasil dan cukup dalam melakukan klasifikasi jurnal penelitian berdasarkan kelas kategori bidang fokus yang terdiri dari 5 bidang dengan capaian akurasi prediksi sebesar 48%.
3. Dengan model penelitian ini dapat membantu dalam pengkategorian Jurnal-jurnal penelitian terbaru (*up to date*) berdasarkan bidang fokus penelitian pada portal Garuda ataupun Sinta yang terindeks Scopus yang bekerja secara otomatis.

5.2 Saran

Saran yang dapat diterapkan guna mengembangkan penelitian ini untuk penelitian berikutnya adalah pelaksanaan penelitian topik yang sama dengan penambahan *dataset* lebih banyak lagi baik untuk kategori ataupun *dataset* yang digunakan. Serta pengambilan data yang lebih akurat dari pada *dataset* penelitian ini (bisa berupa abstrak penelitian), sehingga diharapkan hasil dari peroleh nilai *akurasi*, *recall*, *precision*, dan F1-*Score* akan meningkat serta tingkat prediksi pada saat klasifikasi jurnal lebih baik dan tepat.

DAFTAR PUSTAKA

Alita, D. (2021) 'Multiclass SVM Algorithm for Sarcasm Text in Twitter', *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, 8(1), pp. 118–128. Available at: https://doi.org/10.35957/jatisi.v8i1.646.

Alita, D., Fernando, Y. and Sulistiani, H. (2020) 'Implementasi Algoritma Multiclass Svm Pada Opini Publik Berbahasa Indonesia Di Twitter', *Jurnal Tekno Kompak*, 14(2), p. 86. Available at: https://doi.org/10.33365/jtk.v14i2.792.

Arifidin, S. (1991) 'Bab 3 landasan teori 3.1', pp. 11–27.

Delimayanti, M.K. *et al.* (2021) 'Pemanfaatan Metode Multiclass-SVM pada Model Klasifikasi Pesan Bencana Banjir di Twitter', *Edu Komputika Journal*, 8(1), pp. 39–47. Available at: https://doi.org/10.15294/edukomputika.v8i1.47858.

Feldman, R. and Sanger, J. (2006) *The Text Mining Handbook*, *The Text Mining Handbook*. Available at: https://doi.org/10.1017/cbo9780511546914.

Fikriani, A., Asror, I. and Murti, Y.R. (2019) 'Klasifikasi Kepribadian Berdasarkan Data Twitter dengan Menggunakan Metode Support Vector Machine', *e-Proceeding of Engineering*, 6(3), pp. 10436–10450.

Fitriana, D.N. and Sibaroni, Y. (2020) 'Klasifikasi Data Tweet dengan Menggunakan Metode Klasifikasi Multi-Class Support Vector Machine (SVM) (Studi Kasus : PT.KAI)', *e-Proceeding of Engineering*, 7(2), pp. 8493–8505. Available at: https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/12746.

Irmanda, H.N. and Astriratma, R. (2020) 'Klasifikasi Jenis Pantun Dengan Metode Support Vector Machines (SVM)', *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 4(5), pp. 915–922. Available at: https://doi.org/10.29207/resti.v4i5.2313.

Jumeilah, F.S. (2017) 'Penerapan Support Vector Machine (SVM) untuk Pengkategorian Penelitian', *Jurnal RESTI (Rekayasa Sistem dan*

Teknologi Informasi), 1(1), pp. 19–25. Available at: https://doi.org/10.29207/resti.v1i1.11.

Kurniati, I.D. *et al.* (2015) *Buku Ajar*.

Latif, S. (2018) *TEXT MINING UNTUK KLASIFIKASI KONTEN ABSTRAK JURNAL BAHASA INGGRIS MENGGUNAKAN METODE REDUKSI DIMENSI DAN NAIVE BAYES*, *Universitas HAsanuddin*. Available at: http://dx.doi.org/10.1186/s13662-017-1121-6%0Ahttps://doi.org/10.1007/s41980-018-0101-2%0Ahttps://doi.org/10.1016/j.cnsns.2018.04.019%0Ahttps://doi.org/10.1016/j.cam.2017.10.014%0Ahttp://dx.doi.org/10.1016/j.apm.2011.07.041%0Ahttp://arxiv.org/abs/1502.020.

Maulana, A. (2018) 'Konsep Dasar Data Mining', *Konsep Data Mining*, 1, pp. 1–16.

O'Callaghan, C. (2010) *Peringkat universitas dunia QS - Wikipedia bahasa Indonesia, ensiklopedia bebas*, *Quacquarelli Symonds Limited*. Available at: https://id.wikipedia.org/wiki/Peringkat_universitas_dunia_QS (Accessed: 12 January 2023).

Ramadhan, R., Sari, Y.A. and Adikara, P.P. (2021) 'Perbandingan Pembobotan Term Frequency-Inverse Document Frequency dan Term Frequency-Relevance Frequency terhadap Fitur N-Gram pada Analisis Sentimen', *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 5(11), pp. 5075–5079. Available at: http://j-ptiik.ub.ac.id.

Sastypratiwi, H., Muhardi, H. and Noveanto, M. (2022) 'Klasifikasi Emosi Pada Lirik Lagu Menggunakan Algoritma Multiclass SVM dengan Tuning Hyperparameter PSO', 6, pp. 2279–2286. Available at: https://doi.org/10.30865/mib.v6i4.4609.

Setiawan, K.D. *et al.* (2022) 'Analisis Respon Pelanggan Terhadap Suatu Produk Dengan Metode Support Vector Machine (SVM)', 3(2).