

**PERBANDINGAN REPRESENTASI FITUR PADA KATEGORISASI DAN  
PREDIKSI PRIORITAS LAPORAN DALAM *BUG TRACKING SYSTEM***

**SKRIPSI**

diajukan untuk memenuhi sebagian syarat untuk memperoleh gelar Sarjana  
Komputer pada Program Studi S1 Rekayasa Perangkat Lunak



Oleh

Daffa Almer Fauzan

NIM 1904207

**PROGRAM STUDI REKAYASA PERANGKAT LUNAK**

**KAMPUS UPI DI CIBIRU**

**UNIVERSITAS PENDIDIKAN INDONESIA**

**2023**

## LEMBAR PENGESAHAN

DAFFA ALMER FAUZAN

PERBANDINGAN REPRESENTASI FITUR PADA KATEGORISASI DAN  
PREDIKSI PRIORITAS LAPORAN DALAM *BUG TRACKING SYSTEM*

disetujui dan disahkan oleh pembimbing:

Pembimbing I



**Indira Syawanodya, S.Kom, M.Kom.**

**NIP. 920190219920423201**

Pembimbing II



**Raditya Muhammad, S.T., M.T**

**NIP. 920190219920507101**

Mengetahui

Ketua Program Studi Rekayasa Perangkat Lunak



**Mochamad Iqbal Ardimansyah, S.T., M.Kom.**

**NIP. 920190219910328101**

PERNYATAAN  
KEASLIAN SKRIPSI DAN BEBAS PLAGIARISME

Dengan ini saya menyatakan bahwa skripsi dengan judul “Perbandingan Representasi Fitur pada Kategorisasi dan Prediksi Prioritas Laporan dalam *Bug Tracking System*” ini beserta seluruh isinya adalah benar-benar karya saya sendiri. Saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika ilmu yang berlaku dalam masyarakat keilmuan. Atas pernyataan ini, saya siap menanggung risiko/sanksi apabila di kemudian hari ditemukan adanya pelanggaran etika keilmuan atau ada klaim dari pihak lain terhadap keaslian karya saya ini.

Bandung, 10 Januari 2023



Daffa Almer Fauzan

NIM 1904207

## KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang telah memberikan rahmat dan karunia-Nya hingga penulis dapat menyelesaikan skripsi ini dengan judul “Perbandingan Representasi Fitur pada Kategorisasi dan Prediksi Prioritas Laporan dalam *Bug Tracking System*” untuk memperoleh gelar Sarjana Komputer pada Program Studi S1 Rekayasa Perangkat Lunak di Universitas Pendidikan Indonesia.

Penulis menyadari bahwa keberhasilannya penelitian dan penyusunan skripsi ini berkat bimbingan dan dukungan dari berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Ayah, Ibu, Kakak, dan keluarga besar penulis atas segala doa dan dukungan, baik secara materiel maupun moral selama masa perkuliahan sehingga dapat menyelesaikannya dengan baik.
2. Bapak Mochamad Iqbal Ardimansyah, S.T., M.Kom., selaku Ketua Program Studi Rekayasa Perangkat Lunak.
3. Ibu Asyifa Imanda Septiana, S.Pd., M. Eng., selaku dosen penasihat akademik selama masa perkuliahan.
4. Ibu Indira Syawanodya, M.Kom., selaku dosen pembimbing pertama dalam penelitian dan penyusunan skripsi ini.
5. Bapak Raditya Muhammad, M.T., selaku dosen pembimbing kedua dalam penelitian dan penyusunan skripsi ini.
6. Bapak Hendriyana, S.T., M.Kom., Ibu Dian Anggraini, S.ST., M.T., dan seluruh civitas akademika Universitas Pendidikan Indonesia Kampus UPI di Cibiru atas segala jasanya selama masa perkuliahan.
7. Professor Francesco Marcelloni Ph.D., Professor Pietro Ducange Ph.D., Alessandro Renda Ph.D., dan Professor Annamaria Lossi Ph.D. yang telah memberikan pengalaman akademik yang berharga selama pertukaran pelajar *Indonesian International Students Mobility Awards (IISMA) 2021*.
8. Seluruh rekan-rekan Angkatan 2019 dan 2020 yang telah memberikan semangat dan dukungan kepada penulis

Penulis menyadari bahwa skripsi ini masih memiliki kekurangan. Oleh karena itu, kritik dan saran sangat dibutuhkan guna kesempurnaan skripsi ini. Semoga skripsi ini dapat dijadikan acuan untuk penelitian ke depannya. Semoga kerja keras penulis selama ini dapat diridhoi oleh Allah SWT dan dapat bermanfaat untuk pengembangan ilmu dan masyarakat. Aamiin

Bandung, 10 Januari 2023

Daffa Almer Fauzan

## ABSTRAK

*Bug Tracking System* (BTS) merupakan suatu perangkat lunak yang digunakan dalam tahap pemeliharaan perangkat lunak dan berperan untuk menyimpan riwayat dan melacak laporan terkait permintaan terhadap perubahan, perbaikan kecacatan dan kegagalan, dan dukungan teknis dalam siklus hidup pengembangan perangkat lunak. Kategori dan prioritas suatu laporan dalam BTS dapat ditetapkan secara otomatis menggunakan model pembelajaran mesin. Algoritma yang digunakan dalam penelitian ini adalah *Logistic Regression*. Tujuan dari penelitian adalah mengidentifikasi representasi fitur yang tepat dengan memperhatikan fitur teks secara kontekstual dan karakteristik sumber himpunan data dalam menghadapi permasalahan ketidakseimbangan kelas. Permasalahan ketidakseimbangan kelas dihadapi ketika data pada label kelas, baik berdasarkan kategori maupun prioritas, memiliki jumlah yang tidak seimbang sehingga berdampak terhadap kemampuan model dalam memprediksi label kelas dengan jumlah data yang relatif sedikit. Representasi fitur yang dibandingkan mencakup TF-IDF, TF-IDF dengan SMOTE dan variasinya (ADASYN dan *BorderlineSMOTE*), TF-IDF dengan Word2Vec (CBOW dan *skip-gram*), dan TF-CHI dengan Word2Vec (CBOW dan *skip-gram*). Hasil menunjukkan bahwa model direpresentasikan dengan TF-CHI dengan Word2Vec (CBOW) dapat meningkatkan nilai *precision* paling tinggi sebesar 51%, *recall* paling tinggi sebesar 29%, *F-score* paling tinggi sebesar 35%, dan *accuracy* paling tinggi sebesar 21%. Namun, TF-IDF dengan SMOTE dan variasinya dapat menjadi alternatif solusi ketika anomali terjadi pada TF-CHI, yakni terbentuknya suatu kluster yang terdiri atas sebagian besar atau seluruh label kelas.

**Kata Kunci :** *Bug Tracking System*; Ketidakseimbangan Kelas; *TF-IDF*; *TF-CHI*; *Word2Vec*; *Logistic Regression*.

## **ABSTRACT**

*Bug Tracking System (BTS) is a software that is used in the stage of software maintenance and plays a role in keeping history and tracking reports regarding modification requests, defect fixes, and technical support in the software development life cycle. The category and priority of a report can be set automatically using a machine learning model. The algorithm that is used in this research is Logistic Regression. The objective of this research is to identify the appropriate feature representation by considering the text features contextually and the characteristic of the dataset in dealing with class imbalance problem. The class imbalance problem is faced when the data on their class label, either based on their category or priority, has an imbalance number in terms of amount which affects the capability of the model in predicting the class label with lower amount of data. The feature representation that are being compared includes TF-IDF, TF-IDF with SMOTE and its variations (ADASYN and BorderlineSMOTE), TF-IDF with Word2Vec (CBOW and skip-gram), and TF-CHI with Word2Vec (CBOW and skip-gram). The results show that the model represented by TF-CHI with Word2Vec (CBOW) can increase its precision maximum by 51%, recall maximum by 29%, F-score maximum by 35%, and accuracy maximum by 21%. However, TF-IDF with SMOTE and its variation can be an alternative solution when an anomaly occurs in TF-CHI, that is the formation of a cluster which consists of most or all class labels.*

**Keyword :** *Bug Tracking System; Class Imbalance; TF-IDF; TF-CHI; Word2Vec; Logistic Regression.*

## DAFTAR ISI

KATA PENGANTAR .....	iv
ABSTRAK .....	vi
<i>ABSTRACT</i> .....	vii
DAFTAR ISI .....	viii
DAFTAR GAMBAR .....	xiii
DAFTAR LAMPIRAN .....	xv
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	4
1.3 Tujuan Penelitian .....	4
1.4 Manfaat Penelitian .....	4
1.5 Batasan Penelitian .....	5
1.6 Struktur Organisasi Skripsi .....	5
BAB II KAJIAN PUSTAKA .....	7
2.1 Pemeliharaan Perangkat Lunak dengan <i>Bug Tracking System</i> .....	7
2.1.1 Pemeliharaan Perangkat Lunak Berdasarkan Standar ISO/IEC/IEEE 14764:2022 .....	7
2.1.2 Bug Tracking System .....	9
2.1.3 Jira .....	11
2.2 Klasifikasi Teks dengan Pemrosesan Bahasa Alami .....	12
2.2.1 Term Frequency-Inverse Document Frequency .....	14
2.2.2 Word Embeddings dengan Word2Vec .....	14
2.3 Ketidakseimbangan Kelas .....	17
2.4 Synthetic Minority Oversampling Technique (SMOTE) dan Variasinya .....	18



2.5	Uji Chi Kuadrat .....	20
2.6	Algoritma Logistic Regression.....	21
2.7	Principal Component Analysis .....	23
2.8	Model Referensi .....	23
2.9	Penelitian Terkait .....	24
BAB III METODE PENELITIAN.....		34
3.1	Desain Penelitian .....	34
3.2	Alat dan Bahan Penelitian .....	36
3.3	Instrumen Penelitian.....	38
3.4	Prosedur Penelitian.....	42
3.5	Analisis Data .....	44
BAB IV HASIL DAN PEMBAHASAN .....		45
4.1	Persiapan Data .....	45
4.2	Implementasi Tahapan Penelitian.....	55
4.2.1	Pra-Pemrosesan Teks .....	56
4.2.2	Pembagian Data Latih dan Data Uji .....	57
4.2.3	Ekstraksi dan Representasi Fitur.....	57
4.2.4	Pemodelan dan Evaluasi .....	58
4.3	Analisis Hasil Penelitian .....	76
4.4	Implikasi Hasil Terhadap Implementasi Kerangka Kerja .....	83
BAB V PENUTUP.....		84
5.1	Kesimpulan.....	84
5.2	Saran .....	85
DAFTAR PUSTAKA .....		87
LAMPIRAN.....		94

## DAFTAR TABEL

Tabel 2.1 Jenis-Jenis Pemeliharaan Perangkat Lunak Berdasarkan Standar ISO/IEC/IEEE 14764:2022.....	7
Tabel 2.2 Derajat Ketidakseimbangan Kelas Menurut Kumar, dkk. (2022) .....	17
Tabel 2.3 Tabel Kontingensi pada Uji Chi Kuadrat Antara Dua Variabel Kategorial Menurut Pearson (1904).....	20
Tabel 2.4 Pratinjau Penelitian Terkait.....	24
Tabel 2.5 Ringkasan Penelitian Terkait .....	26
Tabel 2.6 Implementasi Uji Chi Kuadrat pada Klasifikasi Teks .....	32
Tabel 3.1 Deskripsi Repositori pada the Public Jira Dataset .....	37
Tabel 3.2 Distribusi Sampel pada the Public Jira Dataset .....	38
Tabel 3.3 Library yang Digunakan .....	39
Tabel 3.4 Metrik-Metrik Evaluasi.....	40
Tabel 3.5 Confusion Matrix .....	40
Tabel 3.6 Tingkat Prioritas Cacat Perangkat Lunak pada the Public Jira Dataset	41
Tabel 4.1 Jumlah Data Sebelum dan Sesudah Pengecekan Label Kategori dan Prioritas .....	46
Tabel 4.2 Kondisi Awal Label Prioritas Laporan pada The Public Jira Dataset...	47
Tabel 4.3 Jumlah Data Sebelum dan Sesudah Pengecekan Label Prioritas yang Relevan.....	49
Tabel 4.4 Jumlah Data Sebelum dan Sesudah Pengecekan Label Kategori yang Relevan dengan Sumber Proyek .....	50
Tabel 4.5 Daftar Repositori pada Penyaringan Terakhir .....	51
Tabel 4.6 Derajat Ketidakseimbangan pada Masing-Masing Repositori.....	55
Tabel 4.7 Contoh Sebelum dan Sesudah Pra-Pemrosesan Teks .....	57
Tabel 4.8 Contoh Representasi Fitur dengan TF-IDF dan TF-CHI.....	58
Tabel 4.9 Perbandingan Model pada Repositori Mindville Berdasarkan Kategori .....	59
Tabel 4.10 Detail Perbandingan Model pada Repositori Mindville Berdasarkan Kategori.....	60
Tabel 4.11 Perbandingan Model pada Repositori Mindville Berdasarkan Prioritas .....	61

Tabel 4.12	Detail Perbandingan Model pada Repositori Mindville Berdasarkan Prioritas (Precision).....	62
Tabel 4.13	Detail Perbandingan Model pada Repositori Mindville Berdasarkan Prioritas (Recall) .....	62
Tabel 4.14	Detail Perbandingan Model pada Repositori Mindville Berdasarkan Prioritas (F-score).....	63
Tabel 4.15	Perbandingan Model pada Repositori JFrog Berdasarkan Kategori..	64
Tabel 4.16	Detail Perbandingan Model pada Repositori JFrog Berdasarkan Kategori (Precision) .....	65
Tabel 4.17	Detail Perbandingan Model pada Repositori JFrog Berdasarkan Kategori (Recall).....	65
Tabel 4.18	Detail Perbandingan Model pada Repositori JFrog Berdasarkan Kategori (F-score) .....	66
Tabel 4.19	Perbandingan Model pada Repositori JFrog Berdasarkan Prioritas ..	67
Tabel 4.20	Detail Perbandingan Model pada Repositori JFrog Berdasarkan Prioritas (Precision).....	68
Tabel 4.21	Detail Perbandingan Model pada Repositori JFrog Berdasarkan Prioritas (Recall) .....	68
Tabel 4.22	Detail Perbandingan Model pada Repositori JFrog Berdasarkan Prioritas (F-score).....	69
Tabel 4.23	Perbandingan Model pada Repositori Hyperledger Berdasarkan Kategori.....	70
Tabel 4.24	Detail Perbandingan Model pada Repositori Hyperledger Berdasarkan Kategori (Precision) .....	70
Tabel 4.25	Detail Perbandingan Model pada Repositori Hyperledger Berdasarkan Kategori (Recall).....	71
Tabel 4.26	Detail Perbandingan Model pada Repositori Hyperledger Berdasarkan Kategori (F-score).....	72
Tabel 4.27	Perbandingan Model pada Repositori Hyperledger Berdasarkan Prioritas .....	73
Tabel 4.28	Detail Perbandingan Model pada Repositori Hyperledger Berdasarkan Prioritas (Precision).....	74

Tabel 4.29 Detail Perbandingan Model pada Repositori Hyperledger Berdasarkan Prioritas (Recall) .....	74
Tabel 4.30 Detail Perbandingan Model pada Repositori Hyperledger Berdasarkan Prioritas (F-score).....	75
Tabel 4.31 Ringkasan Model Terbaik pada Setiap Repositori.....	76

## DAFTAR GAMBAR

Gambar 2.1 Klasifikasi Jenis-Jenis Pemeliharaan Perangkat Lunak Berdasarkan Standar ISO/IEC/IEEE 14765:2022.....	8
Gambar 2.2 Jenis-Jenis Kegagalan dan Kecacatan Perangkat Lunak Menurut Bath & McKay (2014).....	9
Gambar 2.3 Siklus Hidup Bug Menurut Zhang, dkk. (2015) .....	10
Gambar 2.4 Contoh Laporan dari Proyek Lucene pada Jira .....	11
Gambar 2.5 Aplikasi NLP, NLU, dan NLG Menurut Egger & Gokce (2022) .....	13
Gambar 2.6 Jenis-Jenis Pembelajaran Mesin Menurut Sarker (2021).....	13
Gambar 2.7 Arsitektur Model CBOW dan Skip-Gram Menurut Mikolov, Chen, Corrado, & Dean (2013) .....	15
Gambar 2.8 Contoh Implementasi OVA pada Logistic Regression .....	22
Gambar 2.9 Model Referensi yang Disusun .....	23
Gambar 2.10 Kerangka Kerja CaPBug oleh Ahmed dkk. (2021).....	29
Gambar 2.11 Skema Sumber Himpunan Data pada Transfer Learning oleh Du dkk. (2020) .....	30
Gambar 2.12 Skema Implementasi Transfer Learning oleh Du, dkk. (2020).....	30
Gambar 3.1 Skema Penelitian .....	34
Gambar 3.2 Model Dampak yang Disusun .....	36
Gambar 3.3 Goal Question Metric yang Digunakan.....	39
Gambar 3.4 Prosedur Penelitian.....	42
Gambar 4.1 Jumlah Laporan Repositori Mindville Berdasarkan Kategori.....	52
Gambar 4.2 Jumlah Laporan Repositori Mindville Berdasarkan Prioritas .....	52
Gambar 4.3 Jumlah Laporan Repositori JFrog Berdasarkan Kategori .....	53
Gambar 4.4 Jumlah Laporan Repositori JFrog Berdasarkan Prioritas.....	53
Gambar 4.5 Jumlah Laporan Repositori Hyperledger Berdasarkan Kategori .....	54
Gambar 4.6 Jumlah Laporan Repositori Hyperledger Berdasarkan Prioritas.....	54
Gambar 4.7 Representasi Fitur dengan TF-IDF (Kiri) dan TF-CHI (Kanan) pada Repositori Mindville Berdasarkan Kategori .....	77
Gambar 4.8 Representasi Fitur dengan TF-IDF (Kiri) dan TF-CHI (Kanan) pada Repositori Mindville Berdasarkan Prioritas.....	78

Gambar 4.9 Representasi Fitur dengan TF-IDF (Kiri) dan TF-CHI (Kanan) pada Repositori JFrog Berdasarkan Kategori .....	79
Gambar 4.10 Representasi Fitur dengan TF-IDF (Kiri) dan TF-CHI (Kanan) pada Repositori JFrog Berdasarkan Prioritas .....	80
Gambar 4.11 Representasi Fitur dengan TF-IDF (Kiri) dan TF-CHI (Kanan) pada Repositori Hyperledger Berdasarkan Kategori .....	81
Gambar 4.12 Representasi Fitur dengan TF-IDF (Kiri) dan TF-CHI (Kanan) pada Repositori Hyperledger Berdasarkan Prioritas.....	82

## DAFTAR LAMPIRAN

Lampiran 1. Distribusi pada the Public Jira Dataset .....	94
Lampiran 2. Atribut-Atribut pada the Public Jira Dataset .....	95
Lampiran 3. Contoh Implementasi Confusion Matrix dalam Memprediksi Tingkat Prioritas P1 .....	97
Lampiran 4. Notasi Performatan Teks dalam Jira.....	99
Lampiran 5. Hasil Model TF-IDF dengan Word2Vec pada Repositori Mindville Berdasarkan Kategori.....	107
Lampiran 6. Hasil Model TF-CHI dengan Word2Vec pada Repositori Mindville Berdasarkan Kategori.....	109
Lampiran 7. Representasi Fitur dengan TF-IDF pada Repositori Mindville Berdasarkan Kategori.....	111
Lampiran 8. Representasi Fitur dengan TF-CHI pada Repositori Mindville Berdasarkan Kategori.....	119
Lampiran 9. Hasil Model TF-IDF dengan Word2Vec pada Repositori Mindville Berdasarkan Prioritas .....	127
Lampiran 10. Hasil Model TF-CHI dengan Word2Vec pada Repositori Mindville Berdasarkan Prioritas .....	130
Lampiran 11. Representasi Fitur dengan TF-IDF pada Repositori Mindville Berdasarkan Prioritas .....	133
Lampiran 12. Representasi Fitur dengan TF-CHI pada Repositori Mindville Berdasarkan Prioritas .....	141
Lampiran 13. Hasil Model TF-IDF dengan Word2Vec pada Repositori JFrog Berdasarkan Kategori.....	149
Lampiran 14. Hasil Model TF-CHI dengan Word2Vec pada Repositori JFrog Berdasarkan Kategori.....	152
Lampiran 15. Representasi Fitur dengan TF-IDF pada Repositori JFrog Berdasarkan Kategori.....	155
Lampiran 16. Representasi Fitur dengan TF-CHI pada Repositori JFrog Berdasarkan Kategori.....	163

Lampiran 17. Hasil Model TF-IDF dengan Word2Vec pada Repositori JFrog Berdasarkan Prioritas .....	171
Lampiran 18. Hasil Model TF-CHI dengan Word2Vec pada Repositori JFrog Berdasarkan Prioritas .....	174
Lampiran 19. Representasi Fitur dengan TF-IDF pada Repositori JFrog Berdasarkan Prioritas .....	177
Lampiran 20. Representasi Fitur dengan TF-CHI pada Repositori JFrog Berdasarkan Prioritas .....	185
Lampiran 21. Hasil Model TF-IDF dengan Word2Vec pada Repositori Hyperledger Berdasarkan Kategori.....	193
Lampiran 22. Hasil Model TF-CHI dengan Word2Vec pada Repositori Hyperledger Berdasarkan Kategori.....	196
Lampiran 23. Representasi Fitur dengan TF-IDF pada Repositori Hyperledger Berdasarkan Kategori.....	199
Lampiran 24. Representasi Fitur dengan TF-CHI pada Repositori Hyperledger Berdasarkan Kategori.....	206
Lampiran 25. Hasil Model TF-IDF dengan Word2Vec pada Repositori Hyperledger Berdasarkan Prioritas .....	214
Lampiran 26. Hasil Model TF-CHI dengan Word2Vec pada Repositori Hyperledger Berdasarkan Prioritas .....	217
Lampiran 27. Representasi Fitur dengan TF-IDF pada Repositori Hyperledger Berdasarkan Prioritas .....	220
Lampiran 28. Representasi Fitur dengan TF-CHI pada Repositori Hyperledger Berdasarkan Prioritas .....	228



## DAFTAR PUSTAKA

- Aggarwal, C. C. (2018). *Machine Learning for Text*. Cham: Springer International Publishing.
- Agresti, A. (2019). *An Introduction to Categorical Data Analysis* (Edisi Ketiga). Florida: John Wiley & Sons.
- Ahmed, H. A., Bawany, N. Z., & Shamsi, J. A. (2021). CaPBug-A Framework for Automatic Bug Categorization and Prioritization Using NLP and Machine Learning Algorithms. *IEEE Access*, 9, 50496–50512. doi: <https://doi.org/10.1109/ACCESS.2021.3069248>
- Bath, G., & McKay, J. (2014). *The Software Test Engineer's Handbook: A Study Guide for the ISTQB Test Analyst and Technical Test Analyst Advanced Level Certificates 2012* (Edisi Kedua). San Rafael: Rocky Nook.
- Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3, 993–1022. doi: <https://doi.org/10.5555/944919.944937>
- Blessing, L. T. M., & Chakrabarti, A. (2009). *DRM, a Design Research Methodology*. doi: <https://doi.org/10.1007/978-1-84882-587-1>
- Brereton, R. G. (2018). Principal Component Analysis and Unsupervised Pattern Recognition. Dalam *Chemometrics* (hlm. 163–214). doi: <https://doi.org/10.1002/9781118904695.ch4>
- Chai, C., Wang, J., Luo, Y., Niu, Z., & Li, G. (2022). Data Management for Machine Learning: A Survey. *IEEE Transactions on Knowledge and Data Engineering*, 1–20. doi: <https://doi.org/10.1109/TKDE.2022.3148237>
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16, 321–357. doi: <https://doi.org/10.1613/jair.953>
- Chinchor, N. (1992). MUC-4 evaluation metrics. *Proceedings of the 4th Conference on Message Understanding - MUC4 '92*, 22. doi: <https://doi.org/10.3115/1072064.1072067>

- Du, X., Zhou, Z., Yin, B., & Xiao, G. (2020). Cross-project bug type prediction based on transfer learning. *Software Quality Journal*, 28(1), 39–57. doi: <https://doi.org/10.1007/s11219-019-09467-0>
- Edelstein, D. V. (1993). Report on the IEEE STD 1219–1993—standard for software maintenance. *ACM SIGSOFT Software Engineering Notes*, 18(4), 94–95. doi: <https://doi.org/10.1145/163626.163639>
- Egger, R., & Gokce, E. (2022). Natural Language Processing (NLP): An Introduction. Dalam *Applied Data Science in Tourism: Interdisciplinary Approaches, Methodologies, and Applications* (hlm. 307–334). doi: [https://doi.org/10.1007/978-3-030-88389-8\\_15](https://doi.org/10.1007/978-3-030-88389-8_15)
- Eisenstein, J. (2019). *Introduction to Natural Language Processing*. Cambridge: MIT Press.
- Fahse, T., Huber, V., & van Giffen, B. (2021). Managing Bias in Machine Learning Projects. Dalam *Innovation Through Information Systems* (hlm. 94–109). doi: [https://doi.org/10.1007/978-3-030-86797-3\\_7](https://doi.org/10.1007/978-3-030-86797-3_7)
- Gewers, F. L., Ferreira, G. R., Arruda, H. F. De, Silva, F. N., Comin, C. H., Amancio, D. R., & Costa, L. D. F. (2022). Principal Component Analysis. *ACM Computing Surveys*, 54(4), 1–34. doi: <https://doi.org/10.1145/3447755>
- Glazkova, A. (2020). *A Comparison of Synthetic Oversampling Methods for Multi-class Text Classification*. [Online]. Diakses dari <http://arxiv.org/abs/2008.04636>
- Gomes, L. A. F., Torres, R. da S., & Côrtes, M. L. (2019). Bug report severity level prediction in open source software: A survey and research opportunities. *Information and Software Technology*, 115, 58–78. doi: <https://doi.org/10.1016/j.infsof.2019.07.009>
- Grubb, P., & Takang, A. A. (2003). *Software Maintenance: Concepts and Practice* (Edisi Kedua). Singapore: World Scientific Publishing Co. Pte. Ltd.
- Haibo He, Yang Bai, Garcia, E. A., & Shutao Li. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 1322–1328. doi: <https://doi.org/10.1109/IJCNN.2008.4633969>

- Han, H., Wang, W.-Y., & Mao, B.-H. (2005). *Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning*. doi: [https://doi.org/10.1007/11538059\\_91](https://doi.org/10.1007/11538059_91)
- Hao, X., Huang, J., Qin, F., & Zheng, X. (2022). Multi-label learning with missing features and labels and its application to text categorization. *Intelligent Systems with Applications*, 14, 200086. doi: <https://doi.org/10.1016/j.iswa.2022.200086>
- Hassani, H., Beneki, C., Unger, S., Mazinani, M. T., & Yeganegi, M. R. (2020). Text Mining in Big Data Analytics. *Big Data and Cognitive Computing*, 4(1), 1. doi: <https://doi.org/10.3390/bdcc4010001>
- Huang, J., Xu, L., Qian, K., Wang, J., & Yamanishi, K. (2021). Multi-label learning with missing and completely unobserved labels. *Data Mining and Knowledge Discovery*, 35(3), 1061–1086. doi: <https://doi.org/10.1007/s10618-021-00743-x>
- IEEE. (2022). *ISO/IEC/IEEE 14764:2022(E)*. Geneva: IEEE.
- Jones, K. S. (1972). A STATISTICAL INTERPRETATION OF TERM SPECIFICITY AND ITS APPLICATION IN RETRIEVAL. *Journal of Documentation*, 28(1), 11–21. doi: <https://doi.org/10.1108/eb026526>
- Jorayeva, M., Akbulut, A., Catal, C., & Mishra, A. (2022). Machine Learning-Based Software Defect Prediction for Mobile Applications: A Systematic Literature Review. *Sensors*, 22(7), 2551. doi: <https://doi.org/10.3390/s22072551>
- Kamei, Y., Fukushima, T., McIntosh, S., Yamashita, K., Ubayashi, N., & Hassan, A. E. (2016). Studying just-in-time defect prediction using cross-project models. *Empirical Software Engineering*, 21(5), 2072–2106. doi: <https://doi.org/10.1007/s10664-015-9400-x>
- Kaur, H., Pannu, H. S., & Malhi, A. K. (2020). A Systematic Review on Imbalanced Data Challenges in Machine Learning. *ACM Computing Surveys*, 52(4), 1–36. doi: <https://doi.org/10.1145/3343440>

- Kohli, B., Choudhury, T., Sharma, S., & Kumar, P. (2018). A Platform for Human-Chatbot Interaction Using Python. *2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT)*, 439–444. doi: <https://doi.org/10.1109/ICGCIoT.2018.8753031>
- Kowsari, Jafari Meimandi, Heidarysafa, Mendu, Barnes, & Brown. (2019). Text Classification Algorithms: A Survey. *Information*, *10*(4), 150. doi: <https://doi.org/10.3390/info10040150>
- Kumar, V., Lalotra, G. S., Sasikala, P., Rajput, D. S., Kaluri, R., Lakshmana, K., ... Uddin, M. (2022). Addressing Binary Classification over Class Imbalanced Clinical Datasets Using Computationally Intelligent Techniques. *Healthcare*, *10*(7), 1293. doi: <https://doi.org/10.3390/healthcare10071293>
- Li, Y., Sun, G., & Zhu, Y. (2010). Data Imbalance Problem in Text Classification. *2010 Third International Symposium on Information Processing*, 301–305. doi: <https://doi.org/10.1109/ISIP.2010.47>
- Lobo, S. A., Siswadi, & Bakhtiar, T. (2015). Visualization of classified data with kernel principal component analysis. *Global Journal of Pure and Applied Mathematics*, *11*(4), 2347–2356.
- Lourdusamy, R., & Abraham, S. (2018). A Survey on Text Pre-processing Techniques and Tools. *International Journal of Computer Sciences and Engineering*, *06*(03), 148–157. doi: <https://doi.org/10.26438/ijcse/v6si3.148157>
- Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*, *2*(2), 159–165. doi: <https://doi.org/10.1147/rd.22.0159>
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space*. [Online]. Diakses dari <http://arxiv.org/abs/1301.3781>
- Mikolov, T., Yih, W., & Zweig, G. (2013). Linguistic Regularities in Continuous Space Word Representations. *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 746–751. Atlanta: Association for Computational Linguistics.

- Montgomery, L., Lüders, C., & Maalej, W. (2022). An Alternative Issue Tracking Dataset of Public Jira Repositories. *2022 IEEE/ACM 19th International Conference on Mining Software Repositories (MSR)*, 73–77. doi: <https://doi.org/10.1145/3524842.3528486>
- Nadkarni, P. M., Ohno-Machado, L., & Chapman, W. W. (2011). Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5), 544–551. doi: <https://doi.org/10.1136/amiajnl-2011-000464>
- Niaz, N. U., Shahariar, K. M. N., & Patwary, M. J. A. (2022). Class Imbalance Problems in Machine Learning: A Review of Methods And Future Challenges. *Proceedings of the 2nd International Conference on Computing Advancements*, 485–490. doi: <https://doi.org/10.1145/3542954.3543024>
- Noel, R., Riquelme, F., Lean, R. Mac, Merino, E., Cechinel, C., Barcelos, T. S., ... Munoz, R. (2018). Exploring Collaborative Writing of User Stories With Multimodal Learning Analytics: A Case Study on a Software Engineering Course. *IEEE Access*, 6, 67783–67798. doi: <https://doi.org/10.1109/ACCESS.2018.2876801>
- Pearson, K. (1904). On the Theory of Contingency and Its Relation to Association and Normal Correlation. Dalam *Drapers' Company Research Memoirs: Biometric Series*. London: Dulau and Co.
- Pressman, R. S. (2001). *Software Engineering: a Practitioner's Approach* (Edisi Kelima). New York: McGraw-Hill.
- Qaiser, S., & Ali, R. (2018). Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents. *International Journal of Computer Applications*, 181(1), 25–29. doi: <https://doi.org/10.5120/ijca2018917395>
- Raharjana, I. K., Siahaan, D., & Fatichah, C. (2021). User Stories and Natural Language Processing: A Systematic Literature Review. *IEEE Access*, 9, 53811–53826. doi: <https://doi.org/10.1109/ACCESS.2021.3070606>
- Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN Computer Science*, 2(3), 160. doi: <https://doi.org/10.1007/s42979-021-00592-x>

- Schapire, R. E., & Singer, Y. (1999). Improved Boosting Algorithms Using Confidence-rated Predictions. *Machine Learning*, 37, 297–336. doi: <https://doi.org/https://doi.org/10.1023/A:1007614523901>
- Schön, E.-M., Thomaschewski, J., & Escalona, M. J. (2017). Agile Requirements Engineering: A systematic literature review. *Computer Standards & Interfaces*, 49, 79–91. doi: <https://doi.org/10.1016/j.csi.2016.08.011>
- Shultz, T. R., Fahlman, S. E., Craw, S., Andritsos, P., Tsaparas, P., Silva, R., ... Mueen, A. (2011). Class Imbalance Problem. Dalam *Encyclopedia of Machine Learning* (hlm. 171–171). doi: [https://doi.org/10.1007/978-0-387-30164-8\\_110](https://doi.org/10.1007/978-0-387-30164-8_110)
- Singh, R., & Mangat, N. S. (1996). Stratified Sampling. Dalam *Elements of Survey Sampling* (hlm. 102–144). doi: [https://doi.org/10.1007/978-94-017-1404-4\\_5](https://doi.org/10.1007/978-94-017-1404-4_5)
- Singh, V. B., Misra, S., & Sharma, M. (2017). Bug Severity Assessment in Cross Project Context and Identifying Training Candidates. *Journal of Information & Knowledge Management*, 16(01), 1750005. doi: <https://doi.org/10.1142/S0219649217500058>
- Srinivasa-Desikan, B. (2018). *Natural Language Processing and Computational Linguistic*. Birmingham.
- Torrey, L., & Shavlik, J. (2010). Transfer Learning. Dalam *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques* (hlm. 242–264). Hershey: Information Science Reference.
- Tsoukas, H. (2003). Do we really understand tacit knowledge? Dalam *Handbook of Organizational Learning and Knowledge* (hlm. 410–427).
- Vajjala, S., Majumder, B., Gupta, A., & Surana, H. (2020). *Practical Natural Language Processing*. California: O'Reilly Media, Inc.
- Wasiyo, K. (2010). Using cross project learning to improve project management. *PMI® Global Congress 2010—North America, Washington, DC*. Newtown Square: Project Management Institute.
- Yang, G., Min, K., Lee, J.-W., & Lee, B. (2019). Applying Topic Modeling and Similarity for Predicting Bug Severity in Cross Projects. *KSII Transactions on Internet and Information Systems*, 13(3). doi: <https://doi.org/10.3837/tiis.2019.03.026>

- Zhai, Y., Song, W., Liu, X., Liu, L., & Zhao, X. (2018). A Chi-Square Statistics Based Feature Selection Method in Text Classification. *2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)*, 160–163. doi: <https://doi.org/10.1109/ICSESS.2018.8663882>
- Zhang, J., Wang, X., Hao, D., Xie, B., Zhang, L., & Mei, H. (2015). A survey on bug-report analysis. *Science China Information Sciences*, 58(2), 1–24. doi: <https://doi.org/10.1007/s11432-014-5241-2>
- Zhang, T., & Wang, L. (2020). Research on Text Classification Method Based on Word2vec and Improved TF-IDF. Dalam *Advances in Intelligent Systems and Interactive Applications* (hlm. 199–205). doi: [https://doi.org/10.1007/978-3-030-34387-3\\_24](https://doi.org/10.1007/978-3-030-34387-3_24)