Jurnal ICT : Information Communication & Technology
Vol. 20, No.2, Desember 2021, pp. 254-259
p-ISSN: 2302-0261, e-ISSN: 2303-3363, DOI:

Implementasi Support Vector Machine (SVM) dan Naive Bayes untuk Analisis Sentimen Aplikasi KAI Access

Muhammad Riky Sudrajat[1*], Prima Dina Atika[2*], Herlawati[3*]

[1]Program Studi Informatika, Universitas Bhayangkara Jakarta Raya, Indonesia
Email: [1]holdstrong21@gmail.com, [2]prima.dina@dsn.ubharajaya.ac.id, [3]herlawati@ dsn.ubharajaya.ac.id

INFORMASI ARTIKEL

Histori artikel:

Naskah masuk, 15 November 2021
Direvisi, 17 November 2021
Diiterima, 17 November 2021

Kata Kunci:
Google Play, KAI Access, Support Vector Machine, Naïve Bayes

ABSTRAK

Abstract- *Google Play Store is one of the platforms on Android to download an application, Google Play also provides a feature for the public to be able to provide comments/reviews of the downloaded application. Reviews of the application are in the form of perception, both positive and negative, a review of one of the applications on google play, namely the KAI access application, can be used as research material to find information. The technique that can be used for this research is sentiment analysis, the classification method that will be used for this sentiment analysis is the support vector machine and naive Bayes as a comparison to find better accuracy of the two algorithms, this research can help developers to find out the shortcomings and advantages that must be improved on the application. The results of the study using the Support Vector Machine (SVM) classification obtained an accuracy rate of 93% while using the Naïve Bayes method that was 89%. So, the Support Vector Machine method provides a higher level of accuracy than the Naïve Bayes method.*

Abstrak- Google Play Store merupakan salah satu platform di android untuk mengunduh suatu aplikasi, Google Play juga menyediakan fitur untuk masyarakat dapat memberikan komentar / ulasan terhadap aplikasi yang di unduh nya. Ulasan mengenai aplikasi tersebut berupa sebuah persepsi, baik itu positif maupun negatif, ulasan dari salah satu aplikasi yang ada di google play yaitu aplikasi KAI access dapat dijadikan sebagai bahan penelitian untuk mencari sebuah informasi. Teknik yang dapat digunakan untuk penelitian ini adalah analisis sentimen, metode klasifikasi yang akan digunakan untuk analisis sentimen ini adalah support vector machine serta naive baiyes sebagai pembanding untuk mencari akurasi yang lebih baik dari kedua algoritma tersebut, penelitian ini dapat membantu para developer untuk mengetahui kekurangan maupun kelebihan yang harus di tingkatkan terhadap aplikasinya. Hasil penelitian dengan klasifikasi Support Vector Machine (SVM) diperoleh tingkat akurasi sebesar 93%, sedangkan dengan menggunakan metode Naïve Bayes yaitu sebesar 89%. Sehingga metode Support Vector Machine memberikan tingkat akurasi yang lebih tinggi daripada metode Naïve Bayes.

Penulis Korespondensi:

Prima Dina Atika
Program Studi Informatika,
Fakultas Ilmu Komputer
Universitas Bhayangkara Jakarta Raya
Jl. Raya Perjuangan Bekasi Utara, Kota Bekasi, Jawa Barat 17121, Indonesia
Email: prima.dina@dsn.ubharajaya.ac.id

1. Pendahuluan

Transportasi memudahkan masyarakat untuk mencapai tempat tujuan. Salah satu moda trasnportasi yang dipilih adalah kereta api yang dianggap bisa mengurangi kemacetan [1]

Kereta api merupakan salah satu moda transportasi tertua di dunia (Kereta Api Indonesia, 2018) memiliki berbagai keunggulan komparatif dan kompetitif, dapat menghemat lahan dan energi, memiliki lebih sedikit polusi, memiliki sifat yang baik, dan beradaptasi dengan perubahan teknologi. Industri perkeretaapian saat ini sepenuhnya dikelola oleh negara sebagai perusahaan layanan publik atas nama PT. Kereta Api Indonesia disingkat PT. KAI (*Undang - Undang Republik Indonesia Nomor 13*, 1992)

PT. Kereta Api Indonesia juga melakukan inovasi dengan meluncurkan aplikasi e-ticketing yang diberi nama "KAI Access", untuk pemesanan tiket secara online dan mendapatkan info-info terbaru terkait kereta api. Aplikasi ini dapat diunduh di *Google Playstore*, *Appstore*, *Windows Market*, dan *Blackberry App* (KAI, 2016). Peluncuran aplikasi KAI Access tentunya didasarkan pada data jumlah pengguna kereta api di Indonesia yang kian melonjak.

Berdasarkan penelitian sebelumnya tentang analisis sentimen terhadap Kereta Api Indonesia Seperti Metode *Naïve Bayes Classifier NBC)* yang mencapai akurasi sebesar 84% [3]. Analisis sentimen juga di aplikasikan pada pelayanan ojek online menggunakan algoritma *Support Vector Machine (SVM)* dengan nilai akurasi 76% [4]. Algoritma Regresi Logistik juga berhasil digunakan untuk analisis sentimen Nasabah Pada Layanan Perbankan dengan akurasi mencapai 99,3% [5].

Salah satu algoritma klasifikasi yang sering digunakan dan mendapat banyak perhatian para peneliti adalah *naïve bayes classifier*, kesederhanaan pada algoritma *naïve bayes* yang membuat algoritma tersebut mempunyai daya tarik untuk di implementasikan dalam berbagai aplikasi, tetapi kelemahan yang di hadapi algoritma ini adalah lamanya waktu dan tingkat akurasi prediksi yang digunakan untuk melakukan prediksi [6].

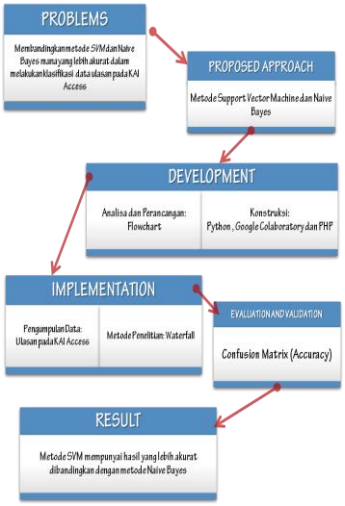
Algoritma *Support Vector Machine* memiliki kelebihan dalam menunjukan performa yang sangat baik untuk prediksi *time series* [7].

Metode *SVM* tidak menghasilkan hasil yang akurat ketika banyak fitur yang tidak relevan, tidak semua fitur diperlukan dalam proses [8]. Regresi Logistik merupakan klasifikasi linier yang telah terbukti menghasilkan klasifikasi yang powerfull dengan statisttik probabilitas dan menangani masalah klasifikasi multikelas [9]. Masalah besar yang dialami oleh algoritma Regresi Logistik adalah ketidak seimbangan kelas *(Class Imbalance)* pada dataset berdimensi tinggi [10].

Oleh karena itu penelitian ini menerapkan metode pengklasifikasian *Support Vector Machine (SVM)* dan Naïve Bayes untuk perbandingan pada penganalisaan sentimen yang terdapat pada aplikasi KAI Access di *platform Google Playstore*.

2. Metode Penelitian

Bagan Kerangka Pemikiran Penelitian dapat dilihat pada Gambar 2.1 berikut ini:



Gambar 1. Bagan Kerangka Pemikiran Penelitian

3. Hasil dan Pembahasan

3.1 Pengumpulan Data

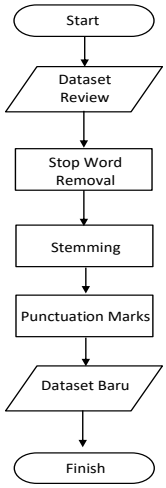
Pengumpulan Data di ambil dari aplikasi KAI Access yang ada di situs Google playstore dengan cara *crawling*. *Crawling* data dilakukan secara realtime sehingga data yang diambil adalah data yang terbaru.

3.2 Preprocessing Data

Text preprocessing berfungsi untuk mengubah data teks yang tidak terstruktur atau sembarang menjadi data yang terstruktur. Secara umum proses yang dilakukan dalam tahapan *preprocessing* adalah sebagai berikut [11]. Data ulasan yang diperoleh belum sepenuhnya siap digunakan untuk proses klasifikasi secara langsung karena data masih tidak terstruktur dengan baik dan terdapat banyak *noise*. Data masih memuat angka, tanda baca, *emoticon*, serta kata-kata lain yang kurang bermakna untuk dijadikan fitur. Maka dari itu, perlu dilakukan *preprocessing* yang bertujuan untuk menyeragamkan bentuk kata, menghilangkan karakter-karakter selain huruf, dan mengurangi

volume kosakata sehingga data akan lebih terstruktur.

Pada tahap *preprocessing*, beberapa tahapan yang dapat dilihat pada gambar 2 berikut ini:



Gambar 2. Tahap Preprocesing

a. *Stop Word Removal*

Tahap *stop word* merupakan tahap untuk menghilangkan kata-kata yang tidak berpengaruh/tidak informatif namun seringkali muncul dalam dokumen. Beberapa kata yang akan dihilangkan adalah kata penghubung, kata ganti, preposisi, kata-kata yang tidak diinginkan.

b. *Stemming*

Stemming berarti menghapus kata awalan dan akhiran untuk menghasilkan kata dasar. Proses ini juga dikatakan sebagai *conflation*. Proses *stemming* secara luas sudah digunakan di dalam *Information retrieval* (pencarian informasi) untuk meningkatkan kualitas informasi yang bisa didapatkan.

C. Punctuation Marks

Setelah tahap *stemming* selanjutnya tahap *punctuation marks* atau menghilangkan tanda baca, *symbol*, maupun *emoticon.*

d. Pembobotan TF-IDF

Hal yang perlu diperhatikan dalam pencarian informasi dari koleksi dokumen yang heterogen adalah pembobotan *term*. Term dapat berupa kata, frase atau unit hasil indexing lainnya dalam suatu dokumen yang dapat digunakan untuk mengetahui konteks dari dokumen tersebut. Karena setiap kata memiliki tingkat kepentingan yang berbeda dalam dokumen, maka untuk setiap kata tersebut diberikan sebuah indikator, yaitu *term weight* [11].

3.3 Representasi Model

Setelah data melalui tahap *preprocessing*, data perlu dibuat model agar data yang masih berupa kata-kata data diolah dan dihitung. Akan dilakukan perhitungan skor berdasarkan kamus kata untuk kemudian diberi label kelas sentimen. Perhitungan skor sentimen hanya berdasarkan pada jumlah kata yang terdeteksi di kamus lexicon dan belum mendeteksi pada tingkatan frasa maupun kalimat.

A. Perhitungan Skor Sentimen

Berdasarkan *review* pengguna "*mudah proses beli tiket lihat jadwal berangkat cetak struk tunjuk qrcode scan beres ada ganggu sedikit saat proses daftar moda transportasi favorit*" Rumus perhitungan skor sentimen yang digunakan pada proses ini adalah jumlah kata positif dikurangkan dengan jumlah kata negatif.

$$Skor = (jumlah\ kata\ positif) - (jumlah\ kata\ negatif)$$

Tabel 1. Perhitungan Skor Sentimen

Review	Kata Positif	Kata Negatif
mudah proses beli tiket lihat jadwal berangkat cetak struk tunjuk qrcode scan beres ada **ganggu** sedikit saat proses daftar moda transportasi **favorit**	mudah favorit	ganggu
Jumlah	2	1
Perhitungan	Skor = 2-1 Skor = 1	

B. Perhitungan Label Sentimen

Analisis sentimen diawali dengan pelabelan data yang dilakukan secara otomatis dengan menghitung skor *sentiment*. Pada tahapan pelabelan akan dilakukan kedalam dua kelas sentimen yaitu sentiment positif dan sentimen negatif. Data akan masuk pada kelas positif jika skor yang didapat > 0, akan masuk pada kelas negatif jika skor yang di dapat < 0. Berikut adalah hasil pelabelan kelas sentimen pada review aplikasi KAI Acces.

Tabel 2. Hasil Label Kelas Sentimen

Sentimen	KAI Access
Positif	731
Negatif	269

Hasil pelabelan kelas sentimen pada aplikasi KAI Access jumlah *review* positif memiliki jumlah yang lebih tinggi dibandingkan dengan jumlah *review* negatif.

C. Klasifikasi Sentimen

Setelah kata-kata tersebut diubah menjadi vektor kemudian diberi nilai dan pembobotan untuk setiap kata untuk kemudian

diolah menggunakan algoritma prediksi, data tersebut akan dibagi menjadi data latih dan data uji.

D. Data Latih dan Data Uji

Algoritma klasifikasi menggunakan data latih untuk membentuk model *classifier*, model yang terbentuk adalah hasil reperesentasi pengetahuan yang akan digunakan prediksi kelas data baru yang belum pernah ada sebelumnya. Banyaknya data latih yang digunakan akan menentukan baik tidaknya machine/mesin dalam memahami pola data sedangkan data uji digunakan untuk mengukur sejauh mana classifier melakukan klasifikasi secara benar. Data latih dan data uji yang digunakan adalah data yang telah memiliki label kelas, dengan perbandingan data uji dan data latih adalah 10% : 90% [12] menyatakan bahwa meskipun penelitian ekstensif belum dilakukan dalam pemilihan rasio yang optimal antara kumpulan data ini, ada beberapa praktik umum dalam memilih ukuran kumpulan data ini. Berdasarkan *Paretto Principle*, rasio yang umum digunakan adalah 90:10 untuk *data sets training* dan *testing*. Perbandingan jumlah data latih dan data uji dapat dilihat pada Tabel 3 berikut:

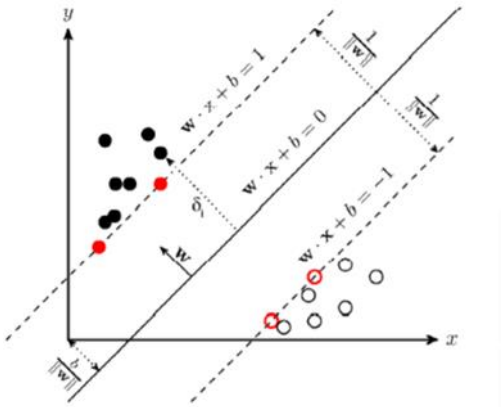
Tabel 3. Perbandingan data latih dan data uji

Klasifikasi	Jumlah	Data Latih (90%)	Data Uji (10%)
Positif	731	658	74
Negatif	269	242	26
Total	**1000**	**900**	**100**

Jumlah data latih pada aplikasi KAI Access sebanyak 900 *review* sedangkan data uji sebanyak 100 *review* sehingga jumlah data *review* adalah sebanyak 1000 *review*. Pembuatan data latih dan data uji dilakukan dengan pengacakan dengan asumsi bahwa setiap data *review* mempunyai peluang yang sama untuk dapat digunakan sebagai data uji maupun data latih. Pengacakan data dilakukan dengan mempertimbangkan kelas data pada setiap data *review* aplikasi karena proporsi data yang tidak seimbang antara kelas positif dan kelas negatif.

E. **Klasifikasi *Support Vector Machine* dan *Naïve Bayes***

Support Vector Machine (*SVM*) merupakan metode pembelajaran terbimbing yang dapat menghasilkan pemetaan fungsi *input-output* dari sekumpulan data pelatihan. *SVM* menggunakan ruang hipotesis berupa fungsi linier pada ruang fitur. Dengan mendukung bias pembelajaran yang diperoleh dari teori pembelajaran statistika, *SVM* dilatih untuk memungkinkannya menggunakan algoritma pembelajaran berdasarkan teori optimasi [13]



Gambar 3. Visualisasi SVM
Sumber: [14]

Pada gambar di atas, dapat dilihat dua kelas data yang dipisahkan secara linier (kelas lingkaran hitam dan kelas lingkaran putih). Kedua tipe ini dipisahkan oleh sebuah garis yang disebut *hyperplane* persamaan untuk garis *hyperplane* adalah w.x + b = 0 (w adalah bidang normal, dan b adalah *offset* atau posisi bidang relatif terhadap koordinat pusat). Pada saat yang sama, vektor yang paling dekat dengan bidang-hiper disebut vektor pendukung. *SVM* akan menggunakan *hyperplane* dengan batas terbesar antar kelas untuk memisahkan data. Oleh karena itu, vektor pendukung yang sejajar dengan semua kategori membentuk garis pemisah. Persamaan yang dibentuk oleh dua jenis garis pemisah tersebut adalah:

$$\begin{aligned} &untuk\ y = +1, maka\ w \cdot x + b = 1 \\ &untuk\ y = -1, maka\ w \cdot c + b = -1 \end{aligned} \quad (1)$$

Dapat dilihat dari persamaan tersebut bahwa data yang termasuk pada kategori pertama adalah data dengan nilai persamaan lebih besar dari atau sama dengan 1 (w.x + b≥1), sedangkan data pada kategori kedua memiliki persamaan sebagai berikut: nilai persamaan yang lebih kecil atau sama dengan -1 (w.x + b≤-1). Mengetahui fakta ini, ketimpangan bisa dibentuk oleh dua garis pemisah

$$y_i\ (x_i \cdot w + b) - 1\ \geq 0 \quad (2)$$

Pencarian *hyperplane* dengan batas terbesar dapat dirumuskan sebagai masalah optimasi terkendala, yaitu

$$\frac{1}{2}\ |w|^2 \quad (3)$$
$$dengan\ y_i\ (x_i \cdot w + b) - 1\ \geq 0$$

Kemudian metode *Lagrangian* akan digunakan untuk mengoptimalkan untuk mengatasi kumpulan data yang besar. Metode *Lagrangian* akan mengubah rumus (4) menjadi

$$L_p(w,b,a) = \frac{1}{2}|w|^2 - \sum_{i=1}^{n} a_i(y_i(x_i.w+b)-1) \quad (4)$$

α merupakan nilai koefisien *Lagrangian* dengan nilai $a_i \geq 0$. Selain itu, persamaan di atas akan diminimalkan untuk w dan b sehingga $\frac{\partial}{\partial b} L_p(w,b,a) = 0 \; dan \; \frac{\partial}{\partial w} L_p(w,b,a) = 0$. Jadi bisa didapatkan

$$w = \sum_{i=1}^{n} a_i y_i x_i \quad (5)$$

$$\sum_{i=1}^{n} a_i y_i = 0 \quad (6)$$

Namun, karena kemungkinan vektor w tidak terbatas, dengan mengganti persamaan (4) dengan (6), rumus *Lagrangian* asli menjadi bentuk masalah ganda.

$$L_D(a) = \sum_{i=1}^{n} a_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j y_i y_j x_i.x_j \quad (7)$$

Dengan diketahui persamaan berikut:

$$\min_{w,b} \; Lp= \; \max_{w,b} \; L_D= \quad (8)$$

Maka rumus untuk mencari separator atau *hyperplane* terbaik adalah

$$(\sum_{i=1}^{n} a_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j y_i y_j x_i.x_j) \quad (9)$$

dengan

$$\sum_{i=1}^{n} a_i y_i = 0 \, , a_i \geq 0, i = 1, \dots, n$$

Rumus di atas menghasilkan vektor α yang akan menghitung nilai w menggunakan rumus (8). Rumus (9) adalah masalah pemrograman sekunder, menyebabkan a_i selalu memiliki nilai. Karena nilai a_i selalu ada / ditemukan, klasifikasi data uji x dapat ditentukan dengan rumus berikut:

$$f(x) = sgn(\sum_{i=1}^{n} a_i y_i x_i.x - b) \quad (10)$$

Naive Bayes Classification merupakan teknik klasifikasi berdasarkan Teorema *Bayes* dengan asumsi independensi di antara para prediktor
Rumus *Bayes* secara umum dapat diberikan sebagai berikut:

$$P(H \mid X) = \frac{P(H|X)\,P(H)}{P(X)} \quad (10)$$

Dari proses pelatihan data tersebut akan didapatkan model klasifikasi selanjutnya akan diuji untuk mengetahui tingkat akurasi model atau sejauh mana model tersebut dapat mengklasifikasikan data uji, Data latih positif dan data latih negatif digunakan oleh algoritma *SVM* dan algoritma *Naïve Bayes* dalam mempelajari pola data berdasarkan ciri-ciri data pada masing-masing kelas. Berikut ini adalah hasil dari algoritma *Support Vector Machine*:

Tabel 4. Akurasi SVM

Metode	Akurasi
SVM	93%
Naïve Bayes	89%

F. Evaluasi model

Evaluasi model untuk mengetahui hasil akurasi klasifikasi yang terbentuk menggunakan *confusion matrix* [15] seperti pada table 5 di bawah ini:

Tabel 5 Hasil Confusion Matrix

Prediksi	SVM		Naïve Bayes	
	Positif	Negatif	Positif	Negatif
Positif	68	1	65	2
Negatif	6	25	9	24
Akurasi 93%			Akurasi 89%	

4. Kesimpulan

Berdasarkan rating dapat diketahui bahwa mayoritas pengguna KAI Access mempunyai penilaian ataupun presepsi yang baik terhadap aplikasi tersebut. Terlihat dari pelabelan sentimen jumlah ulasan positif lebih banyak dibandingkan dengan jumlah ulasan negatif. Adapun jumlah ulasan positif yaitu sebanyak 731 ulasan atau sebesar 73% sedangkan sisanya merupakan ulasan negatif dari total 1.000 ulasan.

Dengan menggunakan perbandingan data latih dan data uji sebesar 90% : 10% diperoleh hasil klasifikasi sentimen menggunakan metode *Support Vector Machine (SVM)* diperoleh tingkat akurasi sebesar 93% artinya dari 100 data ulasan yang diujikan, terdapat 93 ulasan yang benar pengklasifikasiannya oleh metode *SVM*. Sedangkan dengan menggunakan metode *Naïve Bayes* yaitu sebesar 89% artinya dari 100 data ulasan yang diujikan, terdapat 89 ulasan yang benar pengklasifikasiannya oleh metode *Naïve Bayes*. Sehingga metode *Support Vector Machine* memberikan tingkat akurasi yang lebih tinggi daripada metode *Naïve Bayes*.

Daftar Pustaka

[1] S. Mishra, T. F. . Welch, and M. K. Jha,

"Performance indicators for public transit connectivity in multi-modal transportation networks," *Transp. Res. Part A Policy Pract.*, vol. 46, no. 7, pp. 1066–1085, 2012, doi: 10.1016/j.tra.2012.04.006.

[2] "Undang - Undang Republik Indonesia Nomor 13 Tahun 1992," 1992.

[3] S. Nuryahyuni, "Analisis Sentimen Aplikasi Transportasi Online Krl Access Menggunakan Metode Naive Bayes.," *J. Swabumi*, vol. 17, no. 1, pp. 31–38, 2019.

[4] S. Raja, H.;Magdhalena, "Twitter Sentimen Gojek Indonesia Dan Grab Indonesia.," in *rosiding Seminar Nasional Matematika, Statistika, dan Aplikasinya*, 2019, pp. 256–261.

[5] E. D. . Sari and Irhamah, "Analisis Sentimen Nasabah pada Layanan Perbankan Menggunakan Metode Regresi Logistik Biner, Naïve Bayes Classifier (NBC), dan Support Vector Machine (SVM)," *J. Sains dan Seni ITS*, vol. 8, no. 2, pp. D177–D184, 2019.

[6] T. Rosandy, "Perbandingan Metode Naive Bayes Classifier Dengan Metode Decision Tree (C4.5) Untuk Menganalisa Kelancaran Pembiayaan (Studi Kasus : Kspps / Bmt Al-Fadhila)," *J. TIM Darmajaya*, vol. 2, no. 1, pp. 52–62, 2016.

[7] R. F. Widodo, A.B;Aji, "Prediksi Topik Penelitian Menggunakan Kombinasi Antar Support Vector Regression Dan Kurva Logistik.," *Semin. Nas. Apl. Teknol. Inf.*, 2012.

[8] I. C. Drajana, "Metode Support Vector Machine Dan Forward Selection Prediksi Pembayaran Pembelian Bahan Baku Kopra.," *Ilk. J. Ilm.*, vol. 9, no. 2, pp. 116–123, 2017.

[9] A. Canu, S.; Smola, "Kernel methods and the exponential family.," *Neurocomputing.*, 2006.

[10] S. S. Lin, C.W.; Keerthi, "rust Region Newton Method for Large-Scale Logistic Regression.," *J. Mach. Learn. Res.*, 2008.

[11] C. C. Aggarwal, *Data Mining*. Cham: Springer International Publishing, 2015.

[12] S. Suthaharan, *Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning*, 36th ed. Springer US, 2015.

[13] S. T. Christianini, N.; John, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000.

[14] M. Haltuf, "Support Vector Machines for Credit Scoring," University of Economics in Prague, 2014.

[15] F. Gorunescu, *Data Mining Concept, Models and Techniques*. 2011.