Penerapan Metode *Sample Bootstrapping* untuk Meningkatkan Performa kNearest Neighbor pada Dataset Berdimensi Tinggi

Tri Agus Setiawan[(1)] M. Adib Al Karomi [(2)]
STMIK Widya Pratama Pekalongan
Jl. Patriot 25 Pekalongan Telp (0285) 427816
[(1)] **e***mail:tri.triagus.setiawan45@gmail.com*
[(2)] *email:* adib.comp@gmail.com

ABSTRAK

Dalam klasifikasi semakin banyak atribut yang relevan yang dipakai akan mempengaruhi hasil akurasi dari algoritma tersebut. Seleksi fitur merupakan salah satu tahapan *pre processing* klasifikasi dengan cara menghilangkan fitur yang tidak relevan dalam data. Proses ini juga dapat mengurangi dimensi data serta meningkatkan akurasi klasifikasi. Algoritma kNearest Neighbor (kNN) merupakan metode untuk melakukan klasifikasi terhadap objek baru berdasarkan k tetangga terdekatnya. Algoritma kNN memiliki kelebihan karena sederhana, efektif dan telah banyak digunakan pada banyak masalah klasifikasi. Pada penelitian ini penggunaan metode *Sample Bootstrapping* diusulkan untuk meningkatkan akurasi yang optimal pada algoritma kNN. Metode *Sample Bootstrapping* digunakan untuk mengurangi jumlah data training yang akan diproses. Dalam penelitian ini menggunakan dataset yang memiliki dataset dengan dimensi yang tinggi. Dari hasil penelitian, penggunaan *Sample Bootstrapping* dengan algoritma pada dataset *credit approval* akurasinya meningkat 5.4% (96.87%-91.52%) dibandingkan algoritma kNN standar. Dari hasil penelitian yang dilakukan, dapat disimpulkan bahwa penggunaan *Sample Bootstrapping* dengan algoritma kNN menghasilkan akurasi yang lebih baik daripada algoritma kNN standar.

Kata Kunci : algoritma kNN, *Sample Bootstrapping, Atribut*

1 Pendahuluan

1.1 Latar Belakang

Data mining merupakan suatu proses untuk mengidentifikasi pola yang memiliki potensi dan berguna untuk mengelola dataset yang besar (Witten, I. H., Frank, E., & Hall, 2011).Dalam data mining ada 10 algoritma teratas yang paling berpengaruh yang dipilih oleh peneliti dalam komunitas data mining, dimana 6 (enam) diantaranya adalah algoritma klasifikasi yaitu C4.5, Support Vector Machines (SVM), AdaBoost, k Nearest Neighbor (kNN), Naïve Bayes dan CART (Fayed & Atiya, 2009).

Performa suatu algoritma dapat dipengaruhi oleh tipe data yang digunakan (Amancio et al.,2013). Beberapa model algoritma kuat hanya pada tipe data tertentu dan lemah pada tipe data yang lain (Ragab, Noaman, Al-Ghamdi, & Madbouly, 2014) (Patel, Vala, & Pandya, 2014)(Ashari, Paryudi, & Tjoa,2013). Semakin banyak atribut yang relevan yang dipakai dalam klasifikasi akan mempengaruhi hasil akurasi dari algoritma tersebut (Han & Kamber, 2006a) (Maimoon, 2010) (Alpaydin, 2010).

Seleksi fitur merupakan salah satu tahapan *pre processing* klasifikasi dengan cara menghilangkan fitur yang tidak relevan dalam data. Proses ini juga dapat mengurangi dimensi data serta meningkatkan akurasi klasifikasi.

Proses klasifikasi dengan atribut yang terlalu banyak jelas akan memerlukan biaya komputasi yang mahal. Terlebih lagi jika beberapa atribut redundan atau juga tidak relevan yang dapat membuat performa algoritma klasifikasi menurun.

Algortima kNN merupakan sebuah metode untuk melakukan klasifikasi terhadap objek baru berdasarkan (k) tetangga terdekatnya (Witten, I. H., Frank, E., & Hall, 2011)(Amores, 2006)(Morimune & Hoshino, 2008). Tujuan dari algoritma kNN adalah untuk mengklasifikasi objek baru berdasarkan atribut dan training sample (Morimune & Hoshino, 2008)(Han, J., & Kamber, 2012), dimana hasil dari sampel uji yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada kNN.

Algoritma kNN memiliki kelebihan karena sederhana, efektif dan telah banyak digunakan pada banyak masalah klasifikasi (Wu, Xindong & Kumar, 2009).

Beberapa peneliti telah melakukan penelitian tentang pengurangan jumlah data dan waktu komputasi. Penelitian Fayed (Fayed & Atiya, 2009)menggunakan pendekatan *Novel Template Reduction* yang digunakan untuk membuang nilai yang jauh dari batasan *threshold* dan memiliki sedikit pengaruh pada klasifikasi kNN. Penelitian Wan (Wan, Lee, Rajkumar, & Isa, 2012) menggunakan Support Vector Machines-Nearest Neighbor (SVM-NN) dengan pendekatan klasifikasi *hybrid* dengan tujuan bahwa untuk meminimalkan dampak dari akurasi klasifikasi. Penelitian Koon (Neo & Ventura, 2012) menggunakan algoritma *Direct Boosting* untuk meningkatkan akurasi klasifikasi kNN dengan modifikasi pembobotan jarak terhadap data latih.

Oleh karena itu perlu adanya metode untuk mengurangi jumlah *data training* untuk diproses dan mengurangi atribut sehingga mampu meningkatkan akurasi, maka dalam penelitian ini akan dilakukan proses pemilihan atribut yang digunakan dalam proses perhitungan klasifikasi serta pembobotan atribut dalam data sehingga mampu meningkatkan performa kNN pada data berdimensi tinggi.

.

1.2 Landasan Teori

1.2.1 Klasifikasi

Algoritma klasifikasi adalah metode pembelajaran data untuk memprediksi nilai dari sekelompok atribut. Algoritma klasifikasi akan menghasilkan sekumpulan aturan yang disebut *rule* yang akan digunakan sebagai indikator untuk memprediksi kelas dari data yang ingin diprediksi (Vercellis Carlo, 2009). Menurut M. Han, J., & Kamber(Han, J., & Kamber, 2012) klasifikasi adalah proses penemuan model atau fungsi yang menggambarkan dan membedakan kelas data atau konsep yang bertujuan agar bisa digunakan untuk memprediksi kelas dari objek yang label kelasnya tidak diketahui.

1.2.2 K Nearest Neighbor (kNN)

kNN merupakan algoritma *supervised learning*, dimana hasil dari *query instance* yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada kNN. Kelas yang paling banyak muncul yang akan menjadi kelas hasil klasifikasi. kNN merupakan salah satu metode pengklasifikasian data berdasarkan similaritas dengan label data (Larose, 2006), Algortima kNN merupakan sebuah metode untuk melakukan klasifikasi terhadap objek baru berdasarkan *k* tetangga terdekatnya (Gorunescu, 2011). Menurut M. Han, J., & Kamber(Han, J., & Kamber, 2012),kNN adalah algoritma pembelajaran berbasis instan yang menggunakan jarak terdekat dalam menentukan kategori vektor baru dalam set data *training*.

PersamaanPenghitungan untuk mencari *euclidean* dengan *d* adalah jarak dan *p* adalah dimensi data dengan:

$$d_i = \sqrt{\sum_{1=1}^{p} (X_{1i} - X_{2i})^2}$$

dimana:

x1 : *sample* data uji
x2 : data uji
d : jarak
p : dimensi data

1.2.3 Sample Bootstrapping

Metode *Sample Bootstrapping* digunakan untuk mengurangi jumlah data *training* yang akan diproses (Dudani, 1976)(Amores, 2006). Untuk dapat mengatasi dataset yang besar maka perlu perlu adanya sampel data (*sampling*) secara acak agar data yang akan diproses menjadi lebih kecil (Liaw, Wu, & Leou, 2010)(Morimune & Hoshino, 2008), sedangkan untuk mengukur jarak tetangga terdekat digunakan *euclidian distance*(Han, J., & Kamber, 2012) dalam proses klasifikasi.

Menurut Efrondan Tibshirani (Efron & Tibshirani, 1993), prosedur *resampling bootstrap*dapat dituliskan sebagai sebagai berikut:

1. Mengkonstruksi distribusi empiris n dari$\hat{F}$darisuatusampel dengan memberikan probabilitas 1/n padasetiap $X_i$dimana i = 1, 2,..., n
2. Mengambil sampel *bootstrap* berukuran n secararandom dengan pengembalian dari distribusi tahap 1, distribusi empiris n dari$\hat{F}$,disebut sebagai sampel *bootstrap*X^{*1}

3. Menghitung dari tahap 1, statistik $\hat{\theta}$ yang diinginkan darisampel *bootstrap*, disebut sebagai $\hat{\theta}1^{*}$
4. Mengulangi langkah 2 dan 3 hingga B kali, 1diperoleh $\hat{\theta}_1$*, $\hat{\theta}_2$*, …,$\hat{\theta}_B$*
5. Mengkontruksisuatudistribusiprobabilitas dari$\hat{F}$denganmemberikan probabilitas 1/B padasetiap$\hat{\theta}_1$*, $\hat{\theta}_2$*, …,$\hat{\theta}_B$*.Distribusi tersebutmerupakan *estimatorbootstrap* untuk distribusisampling$\hat{\theta}$ dan$\hat{F}$dinotasikan dengan $\hat{F}^{*}$
6. Pendekatan estimasi bootstrap untukmean dari 1, distribusiyaitu:

$$\hat{\theta} = \sum_{b=1}^{B} \hat{\theta}_b^{*} \frac{1}{B}$$

2 Metode Penelitian

Dalam penelitian ini dilakukan dengan menggunakan metode *Sample Bootstrapping* untuk meningkatkan akurasi yang optimal pada algoritma kNN. Metode *Sample Bootstrapping* digunakan untuk mengurangi jumlah data *training* yang akan diproses. Untuk melakukan pengujian akurasi hasil klasikasi dilakukan menggunakan metode *confusion matrix* (Witten, I. H., Frank, E., & Hall, 2011) (Maimon Oded, 2010). Data yang digunakan dalam penelitian ini dengan data Credit Appreval Adapun alat bantu yang dilakuka dengan alat bantu yaitu Rapid Miner.

2.1 Dataset

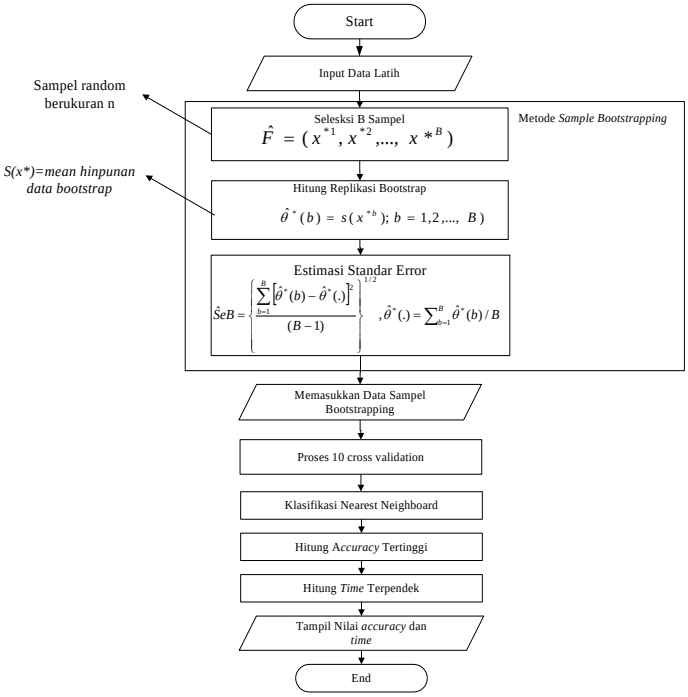
Datas yang akan digunakan dalam penelitian ini adalah data Credit Appreval dengan rincian jumlah record 766, dimensi 14 dengan class 2. Percobaan dilakukan beberapa kali dengan menggunakan jumlah atribut yang berbeda.

2.2 Pengujian

Metode yang diusulkan dalam penelitian ini yaitu dengan *Sample Bootstrapping* dalam meningkatkan akurasi pada algoritma kNN. Metode *Sample Bootstrapping* digunakan untuk mengurangi jumlah data *training* yang akan diproses Adapun tahapan eksperimen pada penelitian ini adalah:

1. Menyiapkan dataset untuk eksperimen
2. Melakukan pengujian menggunakan algoritma kNN menggunakan data Credit Appreval kemudian dari hasil pengujian tersebut dicatat hasil yang diperoleh
3. Melakukan pengujian menggunakan algoritma kNN dengan *Sample Bootstrapping* menggunakan data Credit Appreval kemudian dari hasil pengujian tersebut dicatat hasil yang diperoleh
4. Membandingkan hasil akurasi terbaik dan mengambil hasil terbaik
5. Mengintegrasikan hasil algoritma klasifikasi terbaik.

Adapun algoritma yang diusulkan dalam penelitian ini seperti pada Gambar 1, diawali dengan memasukkan dataset baik *data training* maupun *data testing*, kemudian melakukan transformasi dimana metode *Sample Bootstrapping* digunakan untuk mengurangi jumlah data *training* yang akan diproses kemudian menghitung validitas data *training*, setelah itu menghitung kuadrat jarak *euclidian* (*euclidean distance*) masing-masing objek terhadap data sampel yang diberikan. Kemudian menghitung nilai *distance weighted* yang didapat dari memasukkan nilai validitas dan nilai *eucledian*, setelah melakukan pembobotan atribut dan diperoleh klasifkasi *nearest neighbor*.



Gambar 1 Algoritma *Sample Bootstrapping* dengan kNN

3 Hasil dan Pembahasan

Dalam penelitian ini akan dilakukan komparasi antara algoritma kNN dengan algoritma kNN dan *Sample Bootstrapping*dan

yang akan digunakan untuk mengurangi jumlah data *training* yang akan diproses pada data Credit Appreval

Pada eksperimen pertama akan melakukan perhitungan menggunakan algoritma kNN dengan data Credit Appreval Adapun proses perhitungan kNN sebagai berikut:

1. Menyiapkan data Credit Appreval kita lakukan validasi dengan *cross validation* dimana dataset kita bagi menjadi data *training* dan data *testing*
2. Menentukan nilai *k*, pada penentuan *k* dilakukan input antara 1…677
3. Menghitung kuadrat jarak euclid (*query instance*) masing-masing objek terhadap sampel data yang diberikan dengan menggunakan *euclidian distance* dengan parameter *numeric* dengan rumus:

$$d_i = \sqrt{\sum_{1=i}^{p} (x_{1i} - x_{2i})^2}$$

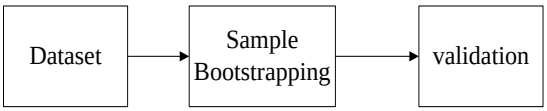
4. Mengurutkan objek-objek termasuk ke dalam kelompok yang mempunyai jarak euclid terkecil
5. Menghitung Akurasi
   Untuk menghitung nilai akurasi digunakan *confusion matrix* dengan rumus:

$$\text{akurasi} = \frac{Jumlah\ Data\ Benar}{Jumlah\ Data} \text{X } 100\%$$

   Hasil perhitungan yang dilakukan untuk algoritma kNNmendapatkan nilai akurasi terbaik pada k=1 dengan akurasi 91.52%

Pada eksperimen kedua akan melakukan perhitungan menggunakan algoritma kNN dengan *Sample Bootstrapping* pada dataset Pinjaman. Adapun proses yang dilakukan adalah:

1. Melakukan preprocessing menggunakan metode *sampling*. Algoritma yang dipakai yaitu *Sample Bootstrapping*, kemudian memilih parameter *sample* yaitu *relative* (sampel dibuat sebagai sebagian kecil dari jumlah total contoh dalam sampel data) dan nilai sampel rasio yang diinput antara 0-1. Setelah dilakukan *sampling* maka data *bootstrap* tersebut divalidasi dengan *cross validation* sebagaimana ditunjukkan dalam Gambar 2.



Gambar 2 Pengujian Performa Algoritma kNN dengan *Sample Bootstrapping*

2. Langkah berikutnya yaitu melakukan normalisasi terhadap *attribute class* pada dataset dan melakukan pembobotan terhadap *attribute class* dengan *weighted relation*. *Weighted relation* mencerminkan relevansi bobot atribut dengan nilai attribut class 0 sampai 1.0, pada penelitian ini bobot atribut diisi, kemudian menentukan k. Dalam hal ini bobot atribut dan nilai k sangat berperan dalam memdapatkan akurasi yang baik. Nilai akurasi dari *confusion matrix* tersebut seperti dalam Tabel 1 dengan rumus:

$$\text{akurasi} = \frac{Jumlah\ Data\ Benar}{Jumlah\ Data} \text{X } 100\%$$

Dari hasil eksperimen tentang nilai akurasi yang dilakukan antara algoritma kNN dengan kNN dengan *Sample Bootstrapping* dapat meiningkatkan akurasi untuk dataset *credit approval* pada algoritma kNN data yang digunakan sejumlah data secara keseluruhan tidak ada proses *filtering* maupun sampel data yang digunakan sehingga tingkat akurasi menjadi rendah, sedangkan pada algoritma kNNdan *Sample Boostrapping* data yang digunakan tidak keseluruhan tetapi dilakukan menggunakan data *sampling* (Witten, I. H., Frank, E., & Hall, 2011) untuk melakukan *filtering* agar mengurangi jumlah data sampel (Champagne, Mcnairn, Daneshfar, & Shang, 2014)(McRoberts, Magnussen, Tomppo, & Chirici, 2011)(Chen & Samson, 2015).

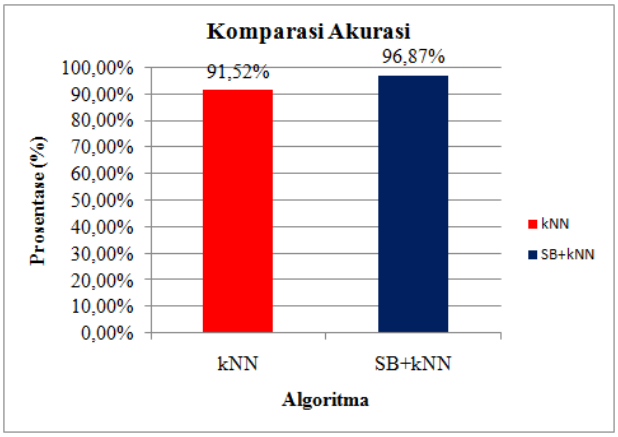
Tabel 1 Komparasi Akurasi Algoritma kNN

Data	Nilai Akurasi (dalam %)		Kenaikan Akurasi (dalam %)
Credit Appreval	kNN	SB+kNN	
	91.52%	96.87%	5,4%

Dengan *Sample Bootstrapping*

Dalam metode *Sample Bootstrapping* terdapat rasio parameter *sample* yang berfungsi memberikan nilai jumlah data sample yang digunakan dari seluruh data yang ada dengan nilai 0-1. Dengan metode ini jumlah data yang diproses tidak secara keseluruhan melainkan

beberapa data tetapi tidak mengurangi jumlah data yang ada karena setelah data tersebut digunakan maka akan dikembalikan lagi (Tian, Song, Li, & Wilde, 2014). Dari hasil perhitungan dapat dilihat perbandingan berdasarkan akurasi didapat hasil dimana algoritma kNN dengan *Sample Bootstrapping* mempunyai nilai akurasi yang lebih baik yaitu 5.4% (96.87%-91.52%) pada k=1 dan t=0.10.



Gambar 3 Komparasi Waktu Komputasi Algoritma kNN dengan *Sample Bootstrapping* dengan kNN

4 Kesimpulan

Dari hasil penelitian tersebut dapat disimpulkan bahwa penerapan *Sample Bootstrapping* dengan algoritma kNN meningkatkan akurasi dibandingkan dengan algoritma kNN standar sebesar 5.4%.

5 Saran dan Penelitian Selanjutnya

Untuk penelitian selanjutnya dapat menggunakan algoritma lain selin kNN seperti *decission tree*, *naive bayes*, *neural network* serta menambahkan algoritma *searching* seperti *Dynamic Programming*ataupun *Djikstra* dalam menentukan nilai *sample* pada *data training* dan bobot atribut sehingga tidak perlu melakukan pengiputan secara manual.

Referensi

Amores, J. (2006). Boosting the distance estimation Application to the K -Nearest Neighbor Classifier. *Pattern Recognition Letters*, *27*(d), 201–209. http://doi.org/10.1016/j.patrec.2005.08.019

Champagne, C., Mcnairn, H., Daneshfar, B., & Shang, J. (2014). A bootstrap method for assessing classification accuracy and confidence for agricultural land use mapping in Canada. *International Journal of Applied Earth Observations and Geoinformation*, *29*, 44–52. http://doi.org/10.1016/j.jag.2013.12.016

Chen, X., & Samson, E. (2015). Environmental assessment of trout farming in France by life cycle assessment : using bootstrapped principal component analysis to better de fi ne system classification. *Journal of Cleaner Production*, *87*, 87–95. http://doi.org/10.1016/j.jclepro.2014.09.021

Dudani, S. a. (1976). The Distance-Weighted k-Nearest-Neighbor Rule. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC-6*(4), 325–327. http://doi.org/10.1109/TSMC.1976.5408784

Efron, B., & Tibshirani, R. J. (1993). A n Introduction to the Bootstrap. *Chapman and Hall, New York*, 33.

Fayed, H. A., & Atiya, A. F. (2009). A Novel Template Reduction Approach for the -Nearest Neighbor Method. *IEEE Transactions on Neural Networks / a Publication of the IEEE Neural Networks Council*, *20*(5), 890–896.

Gorunescu, F. (2011). *Data Mining*. (F. Gorunescu, Ed.). Scientific Publishing Service Pvt.Chennai India. http://doi.org/10.1007/978-3-642-19721-5

Han, J., & Kamber, M. (2012). *Data Mining Concepts and Techniques*. (M. Han, J., & Kamber, Ed.) (Third Edit). USA: Morgan Kaufmann Publishers.

Larose, D. T. (2006). *Data Mining Methodes And Model*. (D. T. Larose, Ed.). USA: John Wiley & Sons, Inc. New York, NY, USA.

Liaw, Y.-C., Wu, C.-M., & Leou, M.-L. (2010). Fast k-nearest neighbors search using modified principal axis search tree. *Digital Signal Processing*, *20*(5), 1494–1501. http://doi.org/10.1016/j.dsp.2010.01.009

Maimon Oded, R. L. (2010). *Data Mining And Knowledge Discovery Handbook*. (R. L. Maimon Oded, Ed.) (Second Edi). Israel: Springer.

McRoberts, R. E., Magnussen, S., Tomppo, E. O., & Chirici, G. (2011). Parametric, bootstrap, and jackknife variance estimators for the k-Nearest Neighbors technique with illustrations using forest inventory and satellite image data. *Remote Sensing of Environment*, *115*(12), 3165–3174. http://doi.org/10.1016/j.rse.2011.07.002

Morimune, K., & Hoshino, Y. (2008). Testing homogeneity of a large data set by bootstrapping. *Mathematics And Computers In Simulation*, *78*, 292–302. http://doi.org/10.1016/j.matcom.2008.01.021

Neo, T. K. C., & Ventura, D. (2012). A direct boosting algorithm for the k-nearest neighbor classifier via local warping of the distance metric. *Pattern Recognition Letters*, *33*(1), 92–102. http://doi.org/10.1016/j.patrec.2011.09.028

Tian, W., Song, J., Li, Z., & Wilde, P. De. (2014). Bootstrap techniques for sensitivity analysis and model selection in building thermal performance analysis. *Applied Energy*, *135*, 320–328. http://doi.org/10.1016/j.apenergy.2014.08.110

Vercellis Carlo. (2009). *Business Intelligence* (First). Italy: John Wiley & Sons, Inc. New York, NY, USA.

Wan, C. H., Lee, L. H., Rajkumar, R., & Isa, D. (2012). A hybrid text classification approach with low dependency on parameter by integrating K-nearest neighbor and support vector machine. *Expert Systems with Applications*, *39*(15), 11880–11888. http://doi.org/10.1016/j.eswa.2012.02.068

Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data mining*. (M. A. Witten, I. H., Frank, E., & Hall, Ed.) (Third Edit). USA: Morgan Kaufmann Publishers.

Wu, Xindong & Kumar, V. (2009). *The Top Ten Algorithms in Data Mining*. (V. Wu, Xindong & Kumar, Ed.). USA: Taylor & Francis Group.