Terbit online pada laman web jurnal : **http://jurnal.iaii.or.id**

JURNAL RESTI

(Rekayasa Sistem dan Teknologi Informasi)

Vol. 2 No. 4 (2018) 633 – 638 ISSN : 2580-0760 (media online)

Deteksi Bot Spammer Pada Twitter Menggunakan Smith Waterman Similarity Dan Time Interval Entropy

Imam Safi'i[a], Arief Setyanto[b], Suwanto Raharjo[c]

[a]Magister Teknik Informatika, Universitas Amikom Yogyakarta, nangimam75@gmail.com

[b]Magister Teknik Informatika, Universitas Amikom Yogyakarta, arief_s@amikom.ac.id

[c]Teknik Informatika, Fakultas Teknologi Industri, Institut Sains & Teknologi AKPRIND Yogyakarta, wa2n@akprind.ac.id

Abstract

Twitter is a social media that interacts through 140-character text-based tweet posts including photos, videos and hyperlinks. Spam tweets contain harmful messages sent continuously. Besides disturbing it is also dangerous for the recipient, exacerbated by the use of bots that automatically and quickly spread spam messages that can cause data damage. This study aims to detect spam bots by utilizing the similarity of tweets using Smith Waterman and the posting time interval. Data tweets are collected using scrap libraries in python in the form of id, text, time, link, based on datasets labeled as available. The data is carried out by text preprocessing steps to clean the text and then do the calculations. The calculation results of both the similarity method and the post time interval are then classified with k-Neaset Neighbor with the previous dataset that has been labeled to get the spam or legitimate bot prediction results. The results of classification experiments with several combinations of k to detect spam bots with similarity criteria and entropy interval obtained the best results k = 3 Neirest Neighbor and 10 fold Cross Validation with a predictive value of detection accuracy of 80%, 84% precission and 84% recall.

Keywords: Detect spam bots on Twitter, with waterman similarity and time interval entropy, k-NN classification for spammer predictions

Abstrak

Twitter merupakan media sosial yang berinteraksi melalui postingan *tweet* yang berbasis teks 140 karakter termasuk foto, video dan *hyperlink*. *Tweet spam* berisi pesan membahayakan yang dikirim secara terus-menerus. Selain mengganggu juga membahayakan bagi yang menerima, diperburuk dengan penggunaan *bot* yang secara otomatis dan cepat menyebarkan pesan *spam* yang dapat menyebabkan kerusakan data. Penelitian ini bertujuan mendeteksi *bot spam* dengan memanfaatkan kemiripan *tweets* menggunakan *Smith Waterman* dan *Interval* waktu *posting*. Data *tweets* dikumpulkan menggunakan *library* scrap di python berupa id, text, time, link, berdasarkan dataset berlabel yang telah tersedia. Data tersebut dilakukan tahapan *text preprocessing* untuk membersihkan teks kemudian dilakukan perhitungan. Hasil perhitungan dari kedua metode *similarity* dan *interval* waktu *posting* kemudian diklasifikasi dengan *k-Neaset Neighbour* dengan dataset sebelumnya yang telah berlabel untuk mendapatkan hasil prediksi *bot spam* atau *legitimate*. Hasil percobaan klasifikasi dengan beberapa kombinasi *k* untuk mendeteksi *bot spam* dengan kriteria *similarity* dan *interval entropy* diperoleh hasil terbaik *k=3 Neirest Neighbour* dan 10 *fold Cross Validation* dengan nilai prediksi deteksi *accuracy* sebesar 80%, *precission* 84% dan *recall* 84%.

Kata kunci : Deteksi *bot spam* di twitter, smith waterman similarity dan time interval entropy, klasifikasi k-NN untuk prediksi spammer

1. Pendahuluan

Masyarakat Indonesia merupakan pengguna terbesar ke 5 setelah USA, Brazil, Jepang dan Inggris, pada penggunaan platform media sosial twitter[1]. Twitter dengan pengguna lebih dari 500 juta dan 400 juta tweet perharinya, memungkinkan pengguna untuk berbagi pesan[1]. Pengguna Twitter menulis tentang berbagai opini, isu-isu yang sedang terjadi atau berbagi suatu produk menyebabkan para spammer mulai menyebarkan sejumlah besar pesan spam dengan tujuan komersialnya[2]. Tweet spam berisi pesan singkat atau link yang dikirimkan secara terus-menerus dan mengganggu pengguna yang menerima. Karakteristik tweet spam yaitu seringkali di posting secara otomatis dan teratur dalam waktu yang dekat dan tweet spam seringkali tidak memiliki ungkapan/ekspresi berbeda dengan pengguna asli yang mem-posting tweet yang memiliki ungkapan ekspresi. Tweet spam di perburuk dengan penggunaan program otomatis (bot)[3]. Bot spammer berbahaya bagi pengguna media sosial, tidak

Diterima Redaksi : 17-08-2017 | Selesai Revisi : 01-10-2017 | Diterbitkan Online : 02-11-2017

hanya berpotensi merusak tweet, juga dapat menyebabkan kerusakan data[4]. Twitter memiliki mekanisme untuk penanganan bot spammer dengan melaporkan, namun memiliki kelemahan apabila laporan pengguna Twitter yang dikumpulkan ternyata laporan palsu[5].

Penelitian yang akan dilakukan dalam deteksi bot spam ini menggunakan parameter tweets similarity dengan Smith Waterman karena belum ada yang menggunakan metode ini untuk deteksi kemiripan tweets. Adapun penelitan yang terdahulu menggunakan cosine similarity untuk kemiripan tweets dan membuang url didalam tweets[6]. Penelitan ini akan memanfaatkan *URL* pada tweets sebagai parameter dan metode similarity lain dalam deteksi bot. Selain kemiripan tweets peneliti memanfaatkan interval waktu *posting tweets* menggunakan *interval entropy*. Hasil perhitungan dari kedua metode tersebut kemudian diklasifikasi dengan metode k-Nearest Neighbour untuk memprediksi akun *bot spam* atau *legitimate* dengan dataset yang telah berlabel. Hasil klasifikasi di validasi dengan *k-fold cross validation* untuk mendapatkan nilai *accuracy*, *precission* dan *recall*.

Penelitian ini bertujuan untuk menunjukkan hasil proporsi akun bot spam atau legitimate user pada twitter menggunakan pendekatan Tweets similarity dan time interval antar tweets. Performa klasfikasi k-NN untuk memprediksi akun bot spam atau legitimate dengan menggabungkan metode Smith Waterman dan Time Interval Entropy.

2. Tinjauan Pustaka

Deteksi antara bot spammer dan legitimate user menggunakan kombinasi metode kemiripan dengan cosine similarity dan waktu antar posting tweet, untuk *url* didalam tweets tidak digunakan, penelitian tersebut mendapatkan tingkat accuracy 85%, precission 94% dan recall 90%[7]. Dimana time stamp digunakan untuk menghitung interval antar tweet dan kemiripan tweet menggunakan unigram matching-based. Data tweet yang digunakan terdiri atas kumpulan akun normal dan akun yang terindikasi sebagai bot spammer yang sudah di kategorikan sebelumnya yang dihasilkan dari penelitianya lebih baik daripada penelitian yang menggunakan satu metode. Dalam penelitianya ada beberapa parameter seperti *URL* yang dibuang saat pre-processing, penelitian[6] menggunakan *URL* sebagai salah satu parameter untuk deteksi spam *campaigns*. Penelitian deteksi bot spammer dengan memanfaatkan fitur waktu dilakukan[8] fitur waktu posting dalam penelitian ini belum bisa mengidentifikasi tweet yang dilakukan berulang kali oleh *legitimate user*, sehingga *legitimate* dapat terinditifikasi sebagai *spammer*. Smith Waterman merupakan algoritma yang digunakan untuk menghitung kemiripan dua buah teks atau dokumen berdasarkan urutan. Algoritma ini mempunyai efek yang baik dalam pencocokan, menggunakan sub matriks yang berisi semua kemungkinan kesamaan, membandingkan nilai-nilai dari sub matriks hingga didapatkan nilai yang optimal[9][10]. Implementasi algoritma Smith Waterman dan Cosine Similarity untuk menghitung kemiripan teks berdasarkan urutan dan kemunculan kata[11]. *Preprocessing* merupakan pengubahan bentuk text yang terstruktur secara acak menjadi terstruktur sesuai kebutuhan, *Preprocessing* terdiri dari *case folding, tokenizing, removing punctuation, removing stop words, removing Link/URL*, dan *stemming*.

3. Metodologi Penelitian

Penelitian ini menggunakan metode eksperimen dimana peneliti mengkaji kemampuan *Natural Language Processing* dalam melakukan deteksi terhadap akun *spam*, sehingga dapat dikategorikan sebagai penelitian inovasi. Dataset yang digunakan berasal dari *Trend Micro's Web Reputation Technology* telah berlabel *spam* dan *legitimate*, kemudian dikumpulkan kembali sebanyak 2000 *Tweets* dengan proses *scrapping* dan pencarian *search API* mengalami pemrosesan text standar.

3.1 Pengumpulan Data

Peneliti menggunakan dataset yang telah disediakan sebanyak 40 akun, dan masing-masing akun diambil 50 Tweets dengan proses *scrapping*, kemudian peneliti membagi menjadi dua data, yaitu data *spammer* sebanyak 25 akun dan data *legitimate* sebanyak 15 akun, terkumpul 2000 tweets dari keseluruhan akun kemudian dilakukan pemrosesan teks standar Tweets.

3.2 Analisis Data

Data tweets dilakukan *standard text preprocessing* untuk membersihkan teks agar meningkatkan akurasi dalam deteksi *spammer* dan *legitimate*.

```
regex = re.sub('(?=RT)[^\s]+', '', regex)
regex = re.sub('(?=rt)[^\s]+', '', regex)
regex = re.sub('(?=https)[^\s]+', '', regex)
regex = re.sub('(?=http)[^\s]+', '', regex)
regex = re.sub('(?=www)[^\s]+', '', regex)
regex = re.sub('(?=WWW)[^\s]+', '', regex)
regex = re.sub('(?=PIC)[^\s]+', '', regex)
regex = re.sub('(?=pic)[^\s]+', '', regex)
regex = re.sub('(?=url)[^\s]+', '', regex)
regex = re.sub('(?=URL)[^\s]+', '', regex)
regex = re.sub('(?=bitly)[^\s]+', '', regex)
proses_regex = regex
from nltk.tokenize import word_tokenize
tokens = word_tokenize(proses_regex)
tokens = [w.lower() for w in tokens]
words = [word for word in tokens if word.isalpha()]
from nltk.corpus import stopwords
stop_words = set(stopwords.words('english'))
words = [w for w in words if not w in stop_words]
from nltk.stem.porter import PorterStemmer
porter = PorterStemmer()
stemmed = [porter.stem(word) for word in tokens]
```

Gambar 1. Preprocessing Data Tweets

Tahapan analisis data dengan *standart text processing* dengan bahasa python dan *library* dari nltk sebagai berikut :

a. Masukan Tweets asli dari setiap Tweets yang diproses seringkali bersifat noisy karena berupa URL atau HTML link, simbol, kode angka ASCII, tanda baca selain koma, titik, tanda seru dan tanda tanya, singkatan kata tidak baku, dan kata dalam bahasa asing. Pada penelitian ini, bahasa asing yang ditemukan tidak memiliki arti karena berfokus pada bahasa Inggris saja.

b. Tahapan menghilangkan url, www, http/s, pic, bitly, Simbol (#,RT,@), kode angka ASCII, dan Tanda Baca koma, titik, tanda seru, dan tanda tanya, kemudian Lower Case Folding melalui proses ini dari library nltk;

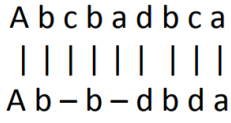
c. Tokenizing tahapan pemotongan berupa kata untuk setiap kalimat yang ada kemudian dipisahkan menjadi kata token dengan cara mendeteksi spasi yang ditemukan.

d. Stop Words Removal menghilangkan kata umum yang tidak memiliki pengaruh signifikan pada sebuah kalimat. Hal ini diselesaikan dengan melakukan proses import daftar stop word dari library nltk.

e. Stemming masukan teks yang sudah dipisahkan menjadi kata token kemudian akan mudah untuk diproses. Salah satunya adalah stemming yang berusaha mengembalikan setiap kata yang ditemukan kembali ke dalam bentuk baku.

3.3 Similarity Smith Waterman

Algoritma Smith Waterman merupakan algoritma klasik yang telah dikenal luas dalam bidang bioinformatika sebagai metode yang dapat mengidentifikasi local similarities (penyejajaran sequence) yaitu proses penyusunan dua *local sequences* (rangkaian/susunan atau rentetan) protein sequences sehingga kemiripan antara dua sequence tersebut akan terlihat. Berdasarkan fungsi proses penyejajaran sekuens tersebut, maka algoritma ini dapat digunakan dalam proses pendeteksian kemiripan tweets dari yang dianggap sebagai tweets spammer dengan cara melihat kemiripan antar tweets yang dipostingkan. Algoritma Smith Waterman sendiri banyak digunakan untuk menghitung penyelarasan lokal yang optimal[12][13].



Gambar 2. Optimal alignment dua substring

Dua urutan urutan kueri dan urutan basis data akan dibandingkan, didefinisikan sebagai A= $a_1a_2...a_n$ dan B= $b_1b_2...\ b_n$ jadilah urutan yang harus disesuaikan, dimana n dan m adalah panjang dari masing-masing A dan B.

1. Tentukan matriks substitusi dan skema penalti gap
   a. *s(a,b)* Nilai kesamaan elemen yang merupakan dua urutan
   b. W_k hukum dari celah yang memiliki panjang *k*
2. Buatlah matriks penilaian *H* dan inisialisasi baris pertama dan kolom pertama. Ukuran dari matriks penilaian adalah (n+1)*(m+1). Perhatikan pengindeksan berbasis 0

$$H_{k0} = H_{0l} = 0\ for\ 0 \le k \le n\ and\ 0 \le 1 \le m \quad (1)$$

3.4 Time Interval Entropy

Time interval entropy digunakan untuk menangkap pola keteraturan waktu posting tweets yang menunjukkan otomatisasi, TIE (H) dihitung dengan menggunakan persamaan (1) dan persamaan (2).

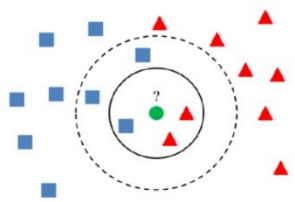
$$H_{\Delta T}(Ti) = -\sum_{i=1}^{nT} P\Delta T(\Delta t_i)\log(P\Delta T(\Delta t_i))$$

$$P\Delta T(\Delta t_i) = \frac{n\Delta t_i}{\sum_{k=1}^{nT} n\Delta t_k} \quad (2)$$

Dimana ΔT merepresentasikan interval waktu antar tweets, dimana $P\Delta T(\Delta t_i)$ menunjukkan probabilitas interval waktu ΔT_i. Komponen *entropy* dapat mendeteksi waktu periodik yang merupakan indikasi kuat terjadinya otomatisasi. Penggunaan Twitter yang memiliki *entropy* lebih rendah dari *threshold* akan diklasifikasikan sebagai *bot spammer* karena nilai *entropy* rendah dibawah *threshold* menunjukkan perilaku yang teratur[2].

3.5 K-Nearest Neighbour

Klasifikasi *k-Nearest Neighbour* mencari sejumlah *k* objek data atau pola (dari semua pola latih yang ada) yang paling dekat dengan pola masukan, kemudian memilih kelas dengan sejumlah pola terbanyak diantara *k* pola tersebut. Penentuan *k* pola terdekat dilakukan berdasarkan ukuran jarak, similarity atau dissimilarity, bergantung jenis atributnya. Pada proses pengklasifikasian, algoritma *k-Neirest Neighbour* menggunakan keterangan sebagai nilai prediksi dari sampel uji yang baru, Jarak yang digunakan adalah jarak *Euclidean Distance*. Klasifikasi dua kelas menggunakan *k-Nearest Neighbour,* adapaun tahapan algoritma *ini adalah* :



Gambar 3. Klasifikasi *K-Nearest Neighbour*

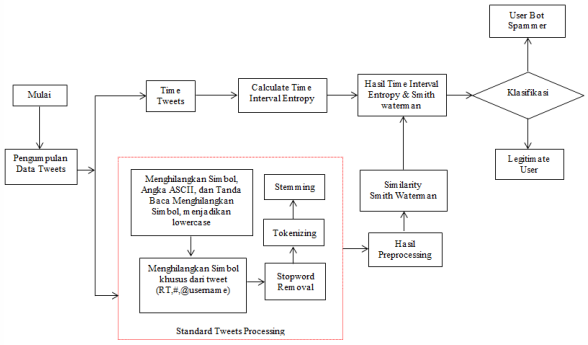
1. Untuk setiap pola latih *<x,f(x)>*, tambahkan pola tersebut ke dalam daftar pola latih
2. Untuk sebuah pola masukan x_q
   a. Misalkan $x_1,x_2...$, x_k adalah *k* pola yang memiliki jarak terdekat (tetangga) dengan x_q
   b. Kembalikan kelas yang memiliki jumlah pola paling banyak diantara *k* pola tersebut sebagai kelas keputusan.

3.6 K-Fold Cross Validation

Pada penelitian ini data berjumlah 40 *user* kemudian dibagi menjadi data *training* dan *testing* dengan perbandingan 80:20. *K-Fold Cross Validation* pada penelitian ini ditentukan jumlah fold 10, untuk memperkirakan tingkat kesalahan yang terjadi, sebab data *training* pada setiap fold cukup berbeda dengan data *training* yang asli.

3.7 Alur Penelitian

Tahapan penelitian yang dilakukan dalam deteksi *bot spammer* dan *legitimate*.



Gambar 4. Alur Penelitian

4. Hasil dan Pembahasan

Dari beberapa hasil percobaan yang telah dilakukan dengan menentukan jumlah tetangga 1 sampai 5 pada klasifikasi dengan metode *Nearest Neighbour* maka peneliti mengambil kesimpulan bahwa *k=3* memiliki *accuracy* deteksi lebih tinggi. Peneliti menggunakan validasi dari beberapa percobaan klasifikasi dengan *K-Fold Cross Validation* sebanyak 10 dan 5 *fold*. Hasil percobaan tersebut menghasilkan tingkat *accuracy* pada *k=5* dan 10 *fold* sebesar 78%. Percobaan *k=5* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 73%. Percobaan *k=4* dan 10 *fold* mendapatkan tingkat *accuracy* sebesar 78%. Percobaan *k=4* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 73%. Percobaan *k=3* dan 10 *fold* mendapatkan tingkat *accuracy* sebesar 80%. Percobaan *k=3* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 75%. Percobaan *k=2* dan 10 *fold* mendapatkan tingkat *accuracy* sebesar 70%. Percobaan *k=2* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 73%. Percobaan *k=1* dan 10 *fold* mendapatkan tingkat *accuracy* sebesar 70%. Percobaan *k=1* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 73%.

Dari beberapa percobaan kombinasi *k* tetangga pada *Nearest Neighbour* dan *Fold Cross Validation* dapat diketahuin *k=3* dan 10 *fold* mendapatkan nilai *accuracy* 80% lebih besar pada deteksi *bot spammer* dengan parameter *similarity tweets* dan *interval entropy* antar *posting*.

4.1 Hasil Klasifikasi

Deteksi *bot spammer* dengan dengan parameter *similarity* dan *time interval entropy* dengan tetangga terdekat sebesar *k*=3 *Nearest Neighbour* dan 10 *Fold Cross Validation* menghasilkan prediksi

Tabel 1. Hasil k=3 *Nearest Neighbour* dan 10 *Fold Cross Validation*

User Id	Tweets Similarity	TIE Tweets	Label Asli	Label Klasifikasi
1	62	1.69	Spam	Spam
2	34	1.05	Spam	Spam
3	16	1.65	Spam	Spam
4	21	1.67	Human	Spam
5	34	1.69	Spam	Spam
6	23	1.67	Spam	Human
7	11	1.68	Human	Human
8	0	1.69	Human	Human
9	76	1.69	Spam	Spam
10	42	1.37	Spam	Spam
11	7	1.69	Human	Human
12	14	1.68	Human	Human
13	17	1.67	Spam	Spam
14	78	1.66	Spam	Spam
15	0	1.68	Human	Human
16	25	1.69	Human	Spam
17	15	1.67	Spam	Human
18	75	1.54	Spam	Spam
19	10	1.69	Human	Human
20	37	1.69	Human	Spam
21	29	1.69	Spam	Spam
22	26	1.68	Spam	Spam
23	14	1.69	Human	Human
24	28	1.68	Human	Spam
25	20	1.69	Spam	Spam
26	80	1.67	Spam	Spam
27	43	1.4	Spam	Spam
28	6	1.69	Human	Human
29	36	1.49	Spam	Spam
30	51	1.69	Spam	Spam
31	30	1.28	Spam	Spam
32	14	1.69	Human	Human
33	65	1.68	Spam	Spam
34	14	1.61	Spam	Human
35	69	1.53	Spam	Spam
36	12	1.69	Human	Human
37	68	1.68	Spam	Spam
38	14	1.67	Spam	Human
39	79	1.57	Spam	Spam
40	15	1.69	Human	Human

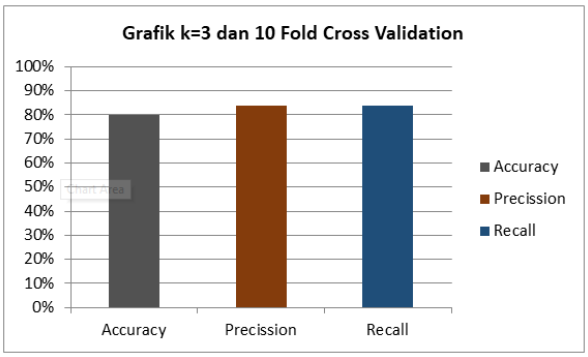
Hasil klasifikasi *K-Nearest Neighbour* dengan parameter *tweets similarity* menggunakan smith waterman dan waktru interval antar *posting tweets* menunjukkan tingkat *accuracy* deteksi *bot spammer* dan *legitimate* sebesar 80%, *precission* 84%, dan *recall* 84%. Sedangkan penelitian sebelumnya yang menggunaan cosine similarity dalam deteksi *similarity tweets* menunjukkan tingkat *accuracy* 85%, *precission* 94% dan *recall* 90%

Tabel 2. Accuracy, Precission dan Recall

Accuracy: 80.00%	True Spam	True Human	Precision
Pred. Spam	21	4	84.00%
Pred. Human	4	11	73.33%
Recall	84.00%	73.33%	

4.2 Grafik

Dibawah ini merupakan grafik hasil prediksi deteksi *bot spammer* dengan menggunakan k-Nearest Neighbour dan Cross Validation, lihat Gambar 5.



Gambar 5. Grafik Hasil Accuracy, Precission dan Recall

Gambar 5. menunjukkan tingkat *accuracy* terbaik dalam memprediksi akun *bot spam* dan *legitimate* sebesar 80%, *precission* 84%, dan *recall* 84%.

5. Kesimpulan

5.1 Simpulan

Berdasarkan hasil penelitian yang telah dilakukan dengan memberdayakan ketersediaan pokok pemikiran, dokumentasi, dan alat bantu dapat disimpulkan hasil deteksi *bot spammer* dengan *tweets similarity smith waterman* menghasilkan nilai yang hampir mirip antara pengguna keduannya. Akun *legitimate* memiliki rata-rata *similarity* kurang dari 28 kemiripan dan lebih dari sama dengan 1.68 pada interval waktu *posting tweets*. Sedangkan untuk akun *bot spam* memiliki rata-rata *similarity* lebih dari 28 kemiripan dan kurang dari 1.68 pada interval waktu posting. Sedangkan untuk prediksi *Bot Spammer* dan *legitimate user* menggunakan klasifikasi k-Nearest Neighbour menunjukkan tingkat *accurcy* sebesar 80%, *precission* 84%, dan *recall* 84% pada klasifikasi *k=3* dan *10 Fold Cross Validation*. Hasil yang didapatkan mempunyai selisih 5% dengan similarity yang menggunakan metode cosine simalirity, untuk penelitian selanjutnya bisa dikembangkan dengan menambahkan jumlah dataset dan menambahkan parameter lain dalam mendeteksi *bot spammer* seperti umur akun, jumlah pengikut, jumlah *tweet* dan *retweet*.

5.2 Saran

Suatu penelitian yang telah dilakukan merupakan awal dari penelitian selanjutnya. Tingkat hasil capain penelitian ini tentunya belum bisa dikatakan ideal untuk menjadi tolak ukur penelitian dengan pendekatan sejenis, namun dapat dijadikan sebagai dasar perbandingan untuk mendapatkan hasil yang lebih baik. Banyaknya alat bantu yang beragam dapat menarik peneliti untuk melakukan analisa dan perbandingan lebih mendalam agar menghasilkan sebuah kesimpulan tentang kesesuain alat bantu dengan kasus maupun skenario tertentu. Koleksi jumlah data yang lebih banyak perlu digunakan untuk meningkatkan akurasi dalam deteksi *bot spam*. Algoritma Smith Waterman yang digunakan untuk mendeteksi kesamaan dalam *Tweets* masih kurang cocok karena urutan huruf yang dihitung.

Daftar Rujukan

[1] W. Hidayat, "Kementrian Komunikasi dan Informatika Republik Indonesia," 2017. [Online]. Available: https://kominfo.go.id/content/detail/4286/pengguna-internet-indonesia-nomor-enam-dunia/0/sorotan_media.

[2] D. P. Christian Sri Kusuma Aditya., Mamluatul Hani'ah., Alif Akbar Fitrawan., Agus Zainal Arifin., "Deteksi Bot Spammer pada Twitter Berbasis Sentiment Analysis dan Time Interval Entropy," *J. Buana Inform.*, vol. 7, 2016.

[3] S. J. Zi Chu, Steven Gianvecchio, Haining Wang, "Who is Tweeting on Twitter: Human, Bot, or Cyborg?," in *Proceedings of the 26th Annual Computer Security Applications Conference*, 2010, pp. 21–30.

[4] H. L. Fred Morstatter., Liang Wu., Tahora H. Nazer., Kathleen M. Carley., "A New Approach to Bot Detection: Striking the Balance Between Precision and Recall," *IEEE*, 2016.

[5] Twitter, "Twitter," *Twitter*. .

[6] Hongyu Gao., Jun Hu., Christo Wilson., Zhichun Li., Yan Chen., Ben Y. Zhao., "Detecting and Characterizing Social Spam Campaigns," *ACM*, 2010.

[7] R. S. Perdana, T. H. Muliawati, and R. Alexandro, "Bot Spammer Detection in Twitter Using Tweet Similarity and Time Interval Entropy," *J. Ilmu Komput. dan Inf.*, vol. 8, no. 1, p. 19, 2015.

[8] Mahdi Washha., Aziz Qaroush., Florence Sedes., "Leveraging Time for Spammers Detection on Twitter," *ACM*, 2016.

[9] Vincentius Riandaru Prasetyo., Edi Winarko., "Rating Of Indonesian Sinetron Based On Public Opinion In Twitter Using Cosine Similarity," *IEEE*, 2016.

[10] R. I. Abdul Munif., Rizky Januar Akbar., Ruchi Intan Tantra., "Rancang Bangun Sistem E-Learning Pemrograman Pada Modul Deteksi Plagiarisme Kode Program Dan Student Feedback System," *J. Ilm. Teknol. Inf.*, vol. 15, 2017.

[11] A. R. Radiant Victor Imbar., Adelia., Mewati Ayub., "Implementasi Cosine Similarity dan Algoritma Smith-Waterman untuk Mendeteksi Kemiripan Teks," *J. Inform.*, vol. 10, 2015.

[12] Gotoh O, "An Improved Algorithm For Matching Biological Sequences," *J. Mol. Biol*, vol. 162, 1982.

[13] Smith T.F., Waterman M.S., "Identification Of Common Molecular Subsequencees," *J. Mol. Biol*, vol. 147, 1981.

Judul : Deteksi Bot Spammer Pada Twitter Menggunakan Smith Waterman Similarity Dan Time Interval Entropy

Penulis : Imam Safi'i, Arief Setyanto, Suwanto Raharjo

Jurnal : Jurnal Resti (Rekayasa Sistem dan Teknologi Informasi)

Penerbit : Badan Penelitian dan Pengembangan SDM dari KOMINFO

Web Jurnal : http://jurnal.iaii.or.id/

Web Paper : http://jurnal.iaii.or.id/index.php/RESTI/article/view/549

DOI : https://doi.org/10.29207/resti.v2i3.549

2019_Resti_Bot_spam_SINTA2_2.pdf

by

Submission date: 26-Aug-2021 02:56PM (UTC+0700)
Submission ID: 1636172811
File name: 2019_Resti_Bot_spam_SINTA2_2.pdf (618.09K)
Word count: 3337
Character count: 19654

Terbit online pada laman web jurnal : **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 2 No. 4 (2018) 633 – 638 ISSN : 2580-0760 (media online)

Deteksi Bot Spammer Pada Twitter Menggunakan Smith Waterman Similarity Dan Time Interval Entropy

Imam Safi'i[a], Arief Setyanto[b], Suwanto Raharjo[c]
[a]Magister Teknik Informatika, Universitas Amikom Yogyakarta, nangimam75@gmail.com
[b]Magister Teknik Informatika, Universitas Amikom Yogyakarta, arief_s@amikom.ac.id
[c]Teknik Informatika, Fakultas Teknologi Industri, Institut Sains & Teknologi AKPRIND Yogyakarta, wa2n@akprind.ac.id

Abstract

Twitter is a social media that interacts through 140-character text-based tweet posts including photos, videos and hyperlinks. Spam tweets contain harmful messages sent continuously. Besides disturbing it is also dangerous for the recipient, exacerbated by the use of bots that automatically and quickly spread spam messages that can cause data damage. This study aims to detect spam bots by utilizing the similarity of tweets using Smith Waterman and the posting time interval. Data tweets are collected using scrap libraries in python in the form of id, text, time, link, based on datasets labeled as available. The data is carried out by text preprocessing steps to clean the text and then do the calculations. The calculation results of both the similarity method and the post time interval are then classified with k-Neaset Neighbor with the previous dataset that has been labeled to get the spam or legitimate bot prediction results. The results of classification experiments with several combinations of k to detect spam bots with similarity criteria and entropy interval obtained the best results k = 3 Neirest Neighbor and 10 fold Cross Validation with a predictive value of detection accuracy of 80%, 84% precission and 84% recall.

Keywords: Detect spam bots on Twitter, with waterman similarity and time interval entropy, k-NN classification for spammer predictions

Abstrak

Twitter merupakan media sosial yang berinteraksi melalui postingan *tweet* yang berbasis teks 140 karakter termasuk foto, video dan *hyperlink*. *Tweet spam* berisi pesan membahayakan yang dikirim secara terus-menerus. Selain mengganggu juga membahayakan bagi yang menerima, diperburuk dengan penggunaan *bot* yang secara otomatis dan cepat menyebarkan pesan *spam* yang dapat menyebabkan kerusakan data. Penelitian ini bertujuan mendeteksi *bot spam* dengan memanfaatkan kemiripan *tweets* menggunakan *Smith Waterman* dan *Interval* waktu *posting*. Data *tweets* dikumpulkan menggunakan *library* scrap di python berupa id, text, time, link, berdasarkan dataset berlabel yang telah tersedia. Data tersebut dilakukan tahapan *text preprocessing* untuk membersihkan teks kemudian dilakukan perhitungan. Hasil perhitungan dari kedua metode *similarity* dan *interval* waktu *posting* kemudian diklasifikasi dengan *k-Neaset Neighbour* dengan dataset sebelumnya yang telah berlabel untuk mendapatkan hasil prediksi *bot spam* atau *legitimate*. Hasil percobaan klasifikasi dengan beberapa kombinasi *k* untuk mendeteksi *bot spam* dengan kriteria *similarity* dan *interval entropy* diperoleh hasil terbaik *k=3 Neirest Neighbour* dan 10 *fold Cross Validation* dengan nilai prediksi deteksi *accuracy* sebesar 80%, *precission* 84% dan *recall* 84%.

Kata kunci : Deteksi *bot spam* di twitter, smith waterman similarity dan time interval entropy, klasifikasi k-NN untuk prediksi spammer

1. Pendahuluan

Masyarakat Indonesia merupakan pengguna terbesar ke 5 setelah USA, Brazil, Jepang dan Inggris, pada penggunaan platform media sosial twitter[1]. Twitter dengan pengguna lebih dari 500 juta dan 400 juta tweet perharinya, memungkinkan pengguna untuk berbagi pesan[1]. Pengguna Twitter menulis tentang berbagai opini, isu-isu yang sedang terjadi atau berbagi suatu produk menyebabkan para spammer mulai menyebarkan sejumlah besar pesan spam dengan tujuan komersialnya[2]. Tweet spam berisi pesan singkat atau link yang dikirimkan secara terus-menerus dan mengganggu pengguna yang menerima. Karakteristik tweet spam yaitu seringkali di posting secara otomatis dan teratur dalam waktu yang dekat dan tweet spam seringkali tidak memiliki ungkapan/ekspresi berbeda dengan pengguna asli yang mem-posting tweet yang memiliki ungkapan ekspresi. Tweet spam di perburuk dengan penggunaan program otomatis (bot)[3]. Bot spammer berbahaya bagi pengguna media sosial, tidak

Diterima Redaksi : 17-08-2017 | Selesai Revisi : 01-10-2017 | Diterbitkan Online : 02-11-2017

hanya berpotensi merusak tweet, juga dapat menyebabkan kerusakan data[4]. Twitter memiliki mekanisme untuk penanganan bot spammer dengan melaporkan, namun memiliki kelemahan apabila laporan pengguna Twitter yang dikumpulkan ternyata laporan palsu[5].

Penelitian yang akan dilakukan dalam deteksi bot spam ini menggunakan parameter tweets similarity dengan Smith Waterman karena belum ada yang menggunakan metode ini untuk deteksi kemiripan tweets. Adapun penelitan yang terdahulu menggunakan cosine similarity untuk kemiripan tweets dan membuang url didalam tweets[6]. Penelitan ini akan memanfaatkan *URL* pada tweets sebagai parameter dan metode similarity lain dalam deteksi bot. Selain kemiripan tweets peneliti memanfaatkan interval waktu *posting tweets* menggunakan *interval entropy*. Hasil perhitungan dari kedua metode tersebut kemudian diklasifikasi dengan metode k-Nearest Neighbour untuk memprediksi akun *bot spam* atau *legitimate* dengan dataset yang telah berlabel. Hasil klasifikasi di validasi dengan *k-fold cross validation* untuk mendapatkan nilai *accuracy*, *precission* dan *recall*.

Penelitian ini bertujuan untuk menunjukkan hasil proporsi akun bot spam atau legitimate user pada twitter menggunakan pendekatan Tweets similarity dan time interval antar tweets. Performa klasfikasi k-NN untuk memprediksi akun bot spam atau legitimate dengan menggabungkan metode Smith Waterman dan Time Interval Entropy.

2. Tinjauan Pustaka

Deteksi antara bot spammer dan legitimate user menggunakan kombinasi metode kemiripan dengan cosine similarity dan waktu antar posting tweet, untuk *url* didalam tweets tidak digunakan, penelitian tersebut mendapatkan tingkat accuracy 85%, precission 94% dan recall 90%[7]. Dimana time stamp digunakan untuk menghitung interval antar tweet dan kemiripan tweet menggunakan unigram matching-based. Data set yang digunakan terdiri atas kumpulan akun normal dan akun yang terindikasi sebagai bot spammer yang sudah di kategorikan sebelumnya yang dihasilkan dari penelitianya lebih baik daripada penelitian yang menggunakan satu metode. Dalam penelitianya ada beberapa parameter seperti *URL* yang dibuang saat pre-processing, penelitian[6] menggunakan *URL* sebagai salah satu parameter untuk deteksi spam *campaigns*. Penelitian deteksi bot spammer dengan memanfaatkan fitur waktu dilakukan[8] fitur waktu posting dalam penelitian ini belum bisa mengidentifikasi tweet yang dilakukan berulang kali oleh *legitimate user*, sehingga *legitimate* dapat terinditifikasi sebagai *spammer*. Smith Waterman merupakan algoritma yang digunakan untuk menghitung kemiripan dua buah teks atau dokumen berdasarkan urutan. Algoritma ini mempunyai efek yang baik dalam pencocokan, menggunakan sub matriks yang berisi semua kemungkinan kesamaan, membandingkan nilai-nilai dari sub matriks hingga didapatkan nilai yang optimal[9][10]. Implementasi algoritma Smith Waterman dan Cosine Similarity untuk menghitung kemiripan teks berdasarkan urutan dan kemunculan kata[11]. *Preprocessing* merupakan pengubahan bentuk text yang terstruktur secara acak menjadi terstruktur sesuai kebutuhan, *Preprocessing* terdiri dari *case folding, tokenizing, removing punctuation, removing stop words, removing Link/URL*, dan *stemming*.

3. Metodologi Penelitian

Penelitian ini menggunakan metode eksperimen dimana peneliti mengkaji kemampuan *Natural Language Processing* dalam melakukan deteksi terhadap akun *spam*, sehingga dapat dikategorikan sebagai penelitian inovasi. Dataset yang digunakan berasal dari *Trend Micro's Web Reputation Technology* telah berlabel *spam* dan *legitimate*, kemudian dikumpulkan kembali sebanyak 2000 *Tweets* dengan proses *scrapping* dan pencarian *search API* mengalami pemrosesan text standar.

3.1 Pengumpulan Data

Peneliti menggunakan dataset yang telah disediakan sebanyak 40 akun, dan masing-masing akun diambil 50 Tweets dengan proses *scrapping*, kemudian peneliti membagi menjadi dua data, yaitu data *spammer* sebanyak 25 akun dan data *legitimate* sebanyak 15 akun, terkumpul 2000 tweets dari keseluruhan akun kemudian dilakukan pemrosesan teks standar Tweets.

3.2 Analisis Data

Data tweets dilakukan *standard text preprocessing* untuk membersihkan teks agar meningkatkan akurasi dalam deteksi *spammer* dan *legitimate*.

```
regex = re.sub('(?=RT)[^\s]+', '', regex)
regex = re.sub('(?=rt)[^\s]+', '', regex)
regex = re.sub('(?=https)[^\s]+', '', regex)
regex = re.sub('(?=http)[^\s]+', '', regex)
regex = re.sub('(?=www)[^\s]+', '', regex)
regex = re.sub('(?=WWW)[^\s]+', '', regex)
regex = re.sub('(?=PIC)[^\s]+', '', regex)
regex = re.sub('(?=pic)[^\s]+', '', regex)
regex = re.sub('(?=url)[^\s]+', '', regex)
regex = re.sub('(?=URL)[^\s]+', '', regex)
regex = re.sub('(?=bitly)[^\s]+', '', regex)
proses_regex = regex
from nltk.tokenize import word_tokenize
tokens = word_tokenize(proses_regex)
tokens = [w.lower() for w in tokens]
words = [word for word in tokens if word.isalpha()]
from nltk.corpus import stopwords
stop_words = set(stopwords.words('english'))
words = [w for w in words if not w in stop_words]
from nltk.stem.porter import PorterStemmer
porter = PorterStemmer()
stemmed = [porter.stem(word) for word in tokens]
```

Gambar 1. Preprocessing Data Tweets

Tahapan analisis data dengan *standart text processing* dengan bahasa python dan *library* dari nltk sebagai berikut :

a. Masukan Tweets asli dari setiap Tweets yang diproses seringkali bersifat noisy karena berupa URL atau HTML link, simbol, kode angka ASCII, tanda baca selain koma, titik, tanda seru dan tanda tanya, singkatan kata tidak baku, dan kata dalam bahasa asing. Pada penelitian ini, bahasa asing yang ditemukan tidak memiliki arti karena berfokus pada bahasa Inggris saja.

b. Tahapan menghilangkan url, www, http/s, pic.ly, Simbol (#,RT,@), kode angka ASCII, dan Tanda Baca koma, titik, tanda seru, dan tanda tanya, kemudian Lower Case Folding melalui proses ini dari library nltk;

c. Tokenizing tahapan pemotongan berupa kata untuk setiap kalimat yang ada kemudian dipisahkan menjadi kata token dengan cara mendeteksi spasi yang ditemukan.

d. Stop Words Removal menghilangkan kata umum yang tidak memiliki pengaruh signifikan pada sebuah kalimat. Hal ini diselesaikan dengan melakukan proses import daftar stop word dari library nltk.

e. Stemming masukan teks yang sudah dipisahkan menjadi kata token kemudian akan mudah untuk diproses. Salah satunya adalah stemming yang berusaha mengembalikan setiap kata yang ditemukan kembali ke dalam bentuk baku.

3.3 Similarity Smith Waterman

Algoritma Smith Waterman merupakan algoritma klasik yang telah dikenal luas dalam bidang informatika sebagai metode yang dapat mengidentifikasi local similarities (penyejajaran sequence) yaitu proses penyusunan dua *local sequences* (rangkaian/susunan atau rentetan) protein sequences sehingga kemiripan antara dua sequence tersebut akan terlihat. Berdasarkan fungsi proses penyejajaran sekuens tersebut, maka algoritma ini dapat digunakan dalam proses pendeteksian kemiripan tweets dari yang dianggap sebagai tweets spammer dengan cara melihat kemiripan antar tweets yang dipostingkan. Algoritma Smith Waterman sendiri banyak digunakan untuk menghitung penyelarasan lokal yang optimal[12][13].

```
A b c b a d b c a
| | | | | | | | |
A b – b – d b d a
```

Gambar 2. Optimal alignment dua substring

Dua urutan urutan kueri dan urutan basis data akan dibandingkan, didefinisikan sebagai A= $a_1a_2...a_n$ dan B= $b_1b_2...\ b_n$ jadilah urutan yang harus disesuaikan, dimana n dan m adalah panjang dari masing-masing A dan B.

1. Tentukan matriks substitusi dan skema penalti gap
   a. *s(a,b)* Nilai kesamaan elemen yang merupakan dua urutan
   b. W_k hukum dari celah yang memiliki panjang *k*
2. Buatlah matriks penilaian *H* dan inisialisasi baris pertama dan kolom pertama. Ukuran dari matriks penilaian adalah (n+1)*(m+1). Perhatikan pengindeksan berbasis 0

$$H_{k0} = H_{0l} = 0 \; for \; 0 \le k \le n \; and \; 0 \le 1 \le m \qquad (1)$$

3.4 Time Interval Entropy

Time interval entropy digunakan untuk menangkap pola keteraturan waktu posting tweets yang menunjukkan otomatisasi, TIE (H) dihitung dengan menggunakan persamaan (1) dan persamaan (2).

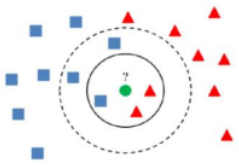
$$H_{\Delta T}(Ti) = -\sum_{i=1}^{nT} P\Delta T(\Delta t_i)\log(P\Delta T(\Delta t_i))$$

$$P\Delta T(\Delta t_i) = \frac{n\Delta t_i}{\sum_{k=1}^{nT} n\Delta t_k} \qquad (2)$$

Dimana ΔT merepresentasikan interval waktu antar tweets, dimana $P\Delta T(\Delta t_i)$ menunjukkan probabilitas interval waktu ΔT_i. Komponen *entropy* dapat mendeteksi waktu periodik yang merupakan indikasi kuat terjadinya otomatisasi. Penggunaan Twitter yang memiliki *entropy* lebih rendah dari *threshold* akan diklasifikasikan sebagai *bot spammer* karena nilai *entropy* rendah dibawah *threshold* menunjukkan perilaku yang teratur[2].

3.5 K-Nearest Neighbour

Klasifikasi *k-Nearest Neighbour* mencari sejumlah *k* obyek data atau pola (dari semua pola latih yang ada) yang paling dekat dengan pola masukan, kemudian memilih kelas dengan sejumlah pola terbanyak diantara *k* pola tersebut. Penentuan *k* pola terdekat dilakukan berdasarkan ukuran jarak, similarity atau dissimilarity, bergantung jenis atributnya. Pada proses pengklasifikasian, algoritma *k-Neirest Neighbour* menggunakan keterangan sebagai nilai prediksi dari sampel uji yang baru, Jarak yang digunakan adalah jarak *Euclidean Distance*. Klasifikasi dua kelas menggunakan *k-Nearest Neighbour,* adapaun tahapan algoritma *ini adalah* :



Gambar 3. Klasifikasi *K-Nearest Neighbour*

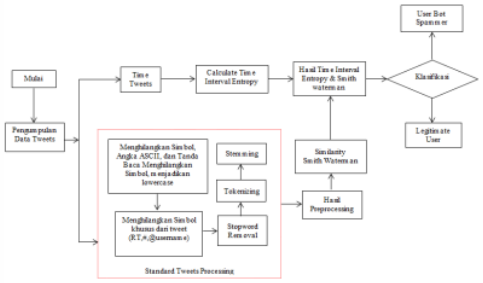
1. Untuk setiap pola latih <$x_f f(x)$>, tambahkan pola tersebut ke dalam daftar pola latih
2. Untuk sebuah pola masukan x_q
   a. Misalkan $x_1, x_2 ..., x_k$ adalah *k* pola yang memiliki jarak terdekat (tetangga) dengan x_q
   b. Kembalikan kelas yang memiliki jumlah pola paling banyak diantara *k* pola tersebut sebagai kelas keputusan.

3.6 K-Fold Cross Validation

Pada penelitian ini data berjumlah 40 *user* kemudian dibagi menjadi data *training* dan *testing* dengan perbandingan 80:20. *K-Fold Cross Validation* pada penelitian ini ditentukan jumlah fold 10, untuk memperkirakan tingkat kesalahan yang terjadi, sebab data *training* pada setiap fold cukup berbeda dengan data *training* yang asli.

3.7 Alur Penelitian

Tahapan penelitian yang dilakukan dalam deteksi *bot spammer* dan *legitimate*.



Gambar 4. Alur Penelitian

4. Hasil dan Pembahasan

Dari beberapa hasil percobaan yang telah dilakukan dengan menentukan jumlah tetangga 1 sampai 5 pada klasifikasi dengan metode *Nearest Neighbour* maka peneliti mengambil kesimpulan bahwa *k=3* memiliki *accuracy* deteksi lebih tinggi. Peneliti menggunakan validasi dari beberapa percobaan klasifikasi dengan *K-Fold Cross Validation* sebanyak 10 dan 5 *fold*. Hasil percobaan tersebut menghasilkan tingkat *accuracy* pada *k=5* dan 10 *fold* sebesar 78%. Percobaan *k=5* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 73%. Percobaan *k=4* dan 10 *fold* mendapatkan tingkat *accuracy* sebesar 78%. Percobaan *k=4* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 73%. Percobaan *k=3* dan 10 *fold* mendapatkan tingkat *accuracy* sebesar 80%. Percobaan *k=3* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 75%. Percobaan *k=2* dan 10 *fold* mendapatkan tingkat *accuracy* sebesar 70%. Percobaan *k=2* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 73%. Percobaan *k=1* dan 10 *fold* mendapatkan tingkat *accuracy* sebesar 70%. Percobaan *k=1* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 73%.

Dari beberapa percobaan kombinasi *k* tetangga pada *Nearest Neighbour* dan *Fold Cross Validation* dapat diketahuin *k=3* dan 10 *fold* mendapatkan nilai *accuracy* 80% lebih besar pada deteksi *bot spammer* dengan parameter *similarity tweets* dan *interval entropy* antar *posting*.

4.1 Hasil Klasifikasi

Deteksi *bot spammer* dengan dengan parameter *similarity* dan *time interval entropy* dengan tetangga terdekat sebesar *k*=3 *Nearest Neighbour* dan 10 *Fold Cross Validation* menghasilkan prediksi

Tabel 1. Hasil k=3 *Nearest Neighbour* dan 10 *Fold Cross Validation*

User Id	Tweets Similarity	TIE Tweets	Label Asli	Label Klasifikasi
1	62	1.69	Spam	Spam
2	34	1.05	Spam	Spam
3	16	1.65	Spam	Spam
4	21	1.67	Human	Spam
5	34	1.69	Spam	Spam
6	23	1.67	Spam	Human
7	11	1.68	Human	Human
8	0	1.69	Human	Human
9	76	1.69	Spam	Spam
10	42	1.37	Spam	Spam
11	7	1.69	Human	Human
12	14	1.68	Human	Human
13	17	1.67	Spam	Spam
14	78	1.66	Spam	Spam
15	0	1.68	Human	Human
16	25	1.69	Human	Spam
17	15	1.67	Spam	Human
18	75	1.54	Spam	Spam
19	10	1.69	Human	Human
20	37	1.69	Human	Spam
21	29	1.69	Spam	Spam
22	26	1.68	Spam	Spam
23	14	1.69	Human	Human
24	28	1.68	Human	Spam
25	20	1.69	Spam	Spam
26	80	1.67	Spam	Spam
27	43	1.4	Spam	Spam
28	6	1.69	Human	Human
29	36	1.49	Spam	Spam
30	51	1.69	Spam	Spam
31	30	1.28	Spam	Spam
32	14	1.69	Human	Human
33	65	1.68	Spam	Spam
34	14	1.61	Spam	Human
35	69	1.53	Spam	Spam
36	12	1.69	Human	Human
37	68	1.68	Spam	Spam
38	14	1.67	Spam	Human
39	79	1.57	Spam	Spam
40	15	1.69	Human	Human

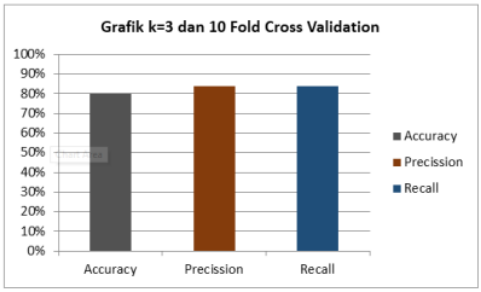
Hasil klasifikasi *K-Nearest Neighbour* dengan parameter *tweets similarity* menggunakan smith waterman dan waktru interval antar *posting tweets* menunjukkan tingkat *accuracy* deteksi *bot spammer* dan *legitimate* sebesar 80%, *precission* 84%, dan *recall* 84%. Sedangkan penelitian sebelumnya yang menggunaan cosine similarity dalam deteksi *similarity tweets* menunjukkan tingkat *accuracy* 85%, *precission* 94% dan *recall* 90%

Tabel 2. Accuracy, Precission dan Recall

Accuracy: 80.00%			
	True Spam	True Human	Precision
Pred. Spam	21	4	84.00%
Pred. Human	4	11	73.33%
Recall	84.00%	73.33%	

4.2 Grafik

Dibawah ini merupakan grafik hasil prediksi deteksi *bot spammer* dengan menggunakan k-Nearest Neighbour dan Cross Validation, lihat Gambar 5.



Gambar 5. Grafik Hasil Accuracy, Precission dan Recall

Gambar 5. menunjukkan tingkat *accuracy* terbaik dalam memprediksi akun *bot spam* dan *legitimate* sebesar 80%, *precission* 84%, dan *recall* 84%.

5. Kesimpulan

5.1 Simpulan

Berdasarkan hasil penelitian yang telah dilakukan dengan memberdayakan ketersediaan pokok pemikiran, dokumentasi, dan alat bantu dapat disimpulkan hasil deteksi *bot spammer* dengan *tweets similarity smith waterman* menghasilkan nilai yang hampir mirip antara pengguna keduannya. Akun *legitimate* memiliki rata-rata *similarity* kurang dari 28 kemiripan dan lebih dari sama dengan 1.68 pada interval waktu *posting tweets*. Sedangkan untuk akun *bot spam* memiliki rata-rata *similarity* lebih dari 28 kemiripan dan kurang dari 1.68 pada interval waktu posting. Sedangkan untuk prediksi *Bot Spammer* dan *legitimate user* menggunakan klasifikasi k-Nearest Neighbour menunjukkan tingkat *accurcy* sebesar 80%, *precission* 84%, dan *recall* 84% pada klasifikasi *k=3* dan *10 Fold Cross Validation*. Hasil yang didapatkan mempunyai selisih 5% dengan similarity yang menggunakan metode cosine simalirity, untuk penelitian selanjutnya bisa dikembangkan dengan menambahkan jumlah dataset dan menambahkan parameter lain dalam mendeteksi *bot spammer* seperti umur akun, jumlah pengikut, jumlah *tweet* dan *retweet*.

5.2 Saran

Suatu penelitian yang telah dilakukan merupakan awal dari penelitian selanjutnya. Tingkat hasil capain penelitian ini tentunya belum bisa dikatakan ideal untuk menjadi tolak ukur penelitian dengan pendekatan sejenis, namun dapat dijadikan sebagai dasar perbandingan untuk mendapatkan hasil yang lebih baik. Banyaknya alat bantu yang beragam dapat menarik peneliti untuk melakukan analisa dan perbandingan lebih mendalam agar menghasilkan sebuah kesimpulan tentang kesesuain alat bantu dengan kasus maupun skenario tertentu. Koleksi jumlah data yang lebih banyak perlu digunakan untuk meningkatkan akurasi dalam deteksi *bot spam*. Algoritma Smith Waterman yang digunakan untuk mendeteksi kesamaan dalam *Tweets* masih kurang cocok karena urutan huruf yang dihitung.

Daftar Rujukan

[1] W. Hidayat, "Kementrian Komunikasi dan Informatika Republik Indonesia," 2017. [Online]. Available: https://kominfo.go.id/content/detail/4286/pengguna-internet-indonesia-nomor-enam-dunia/0/sorotan_media.

[2] D. P. Christian Sri Kusuma Aditya., Ma'rifatul Hani'ah., Alif Akbar Fitrawan., Agus Zainal Arifin., "Deteksi Bot Spammer pada Twitter Berbasis Sentiment Analysis dan Time Interval Entropy," *J. Buana Inform.*, vol. 7, 2016.

[3] S. J. Zi Chu, Steven Gianvecchio, Haining Wang, "Who is Tweeting on Twitter: Human, Bot, or Cyborg?," in *Proceedings of the 26th Annual Computer Security Applications Conference*, 2010, pp. 21–30.

[4] H. L. Fred Morstatter., Liang Wu., Tahora H. Nazer., Kathleen M. Carley., "A New Approach to Bot Detection: Striking the Balance Between Precision and Recall," *IEEE*, 2016.

[5] Twitter, "Twitter," *Twitter*. .

[6] Hongyu Gao., Jun Hu., Christo Wilson., Zhichun Li., Yan Chen., Ben Y. Zhao., "Detecting and Characterizing Social Spam Campaigns," *ACM*, 2010.

[7] R. S. Perdana, T. H. Muliawati, and R. Alexandro, "Bot Spammer Detection in Twitter Using Tweet Similarity and Time Interval Entropy," *J. Ilmu Komput. dan Inf.*, vol. 8, no. 1, p. 19, 2015.

[8] Mahdi Washha., Aziz Qaroush., Florence Sedes., "Leveraging Time for Spammers Detection on Twitter," *ACM*, 2016.

[9] Vincentius Riandaru Prasetyo., Edi Winarko., "Rating Of Indonesian Sinetron Based On Public Opinion In Twitter Using Cosine Similarity," *IEEE*, 2016.

[10] Abdul Munif., Rizky Januar Akbar., Ruchi Intan Tantra., "Rancang Bangun Sistem E-Learning Pemrograman Pada Modul Deteksi Plagiarisme Kode Program Dan Student Feedback System," *J. Ilm. Teknol. Inf.*, vol. 15, 2017.

[11] A. R. Radiant Victor Imbar., Adelia., Mewati Ayub., "Implementasi Cosine Similarity dan Algoritma Smith-Waterman untuk Mendeteksi Kemiripan Teks," *J. Inform.*, vol. 2015.

[12] Gotoh O, "An Improved Algorithm For Matching Biological Sequences," *J. Mol. Biol*, vol. 162, 1982.

[13] Smith T.F., Waterman M.S., "Identification Of Common Molecular Subsequencees," *J. Mol. Biol*, vol. 147, 1981.

2019_Resti_Bot_spam_SINTA2_2.pdf

ORIGINALITY REPORT

20%	18%	12%	12%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Universitas Esa Unggul Student Paper	2%
2	sisfotek.iaii.or.id Internet Source	2%
3	media.neliti.com Internet Source	2%
4	dblp.dagstuhl.de Internet Source	1%
5	www.scribd.com Internet Source	1%
6	www.neliti.com Internet Source	1%
7	Latifah Alhaura, Indra Budi. "Malicious Account Detection on Indonesian Twitter Account", 2020 3rd International Conference on Computer and Informatics Engineering (IC2IE), 2020 Publication	1%
8	juti.if.its.ac.id Internet Source	

		1 %
9	Wang Xiang, Zhang Zhilin, Yu Xiang, Jia Yan, Zhou Bin, Li Shasha. "Finding the hidden hands: a case study of detecting organized posters and promoters in SINA weibo", China Communications, 2015 Publication	1 %
10	mymisterplanner.com Internet Source	1 %
11	Submitted to School of Business and Management ITB Student Paper	1 %
12	Saikin Saikin, Kusrini Kusrini. "MODEL DATA MINING UNTUK KAREKTERISTIK DATA TRAVELLER PADA PERUSAHAAN TOUR AND TRAVEL", Jurnal Manajemen Informatika dan Sistem Informasi, 2019 Publication	1 %
13	docobook.com Internet Source	1 %
14	edoc.pub Internet Source	1 %
15	www.slideshare.net Internet Source	1 %

16	Submitted to University of London External System Student Paper	1 %
17	sciencepubco.com Internet Source	1 %
18	agustrihandayani.wordpress.com Internet Source	<1 %
19	Submitted to University of Melbourne Student Paper	<1 %
20	Submitted to Sriwijaya University Student Paper	<1 %
21	Submitted to University of Hertfordshire Student Paper	<1 %
22	E S Arbintarso, M Muslim, T Rusianto. "Simulation and Failure Analysis of Car Bumper Made of Pineapple Leaf Fiber Reinforced Composite", IOP Conference Series: Materials Science and Engineering, 2018 Publication	<1 %
23	repository.ub.ac.id Internet Source	<1 %
24	link.springer.com Internet Source	<1 %

25	Antonius Rachmat Chrismanto, Willy Sudiarto Raharjo, Yuan Lukito. "Firefox Extension untuk Klasifikasi Komentar Spam pada Instagram Berbasis REST Services", Jurnal Edukasi dan Penelitian Informatika (JEPIN), 2019 Publication	<1%
26	ejournal.undip.ac.id Internet Source	<1%
27	Submitted to UIN Sultan Syarif Kasim Riau Student Paper	<1%
28	jiki.cs.ui.ac.id Internet Source	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		

2019_Resti_Bot_spam_SINTA2_2.pdf

by

Submission date: 26-Aug-2021 02:56PM (UTC+0700)
Submission ID: 1636172811
File name: 2019_Resti_Bot_spam_SINTA2_2.pdf (618.09K)
Word count: 3337
Character count: 19654

Terbit online pada laman web jurnal : **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 2 No. 4 (2018) 633 – 638 ISSN : 2580-0760 (media online)

Deteksi Bot Spammer Pada Twitter Menggunakan Smith Waterman Similarity Dan Time Interval Entropy

Imam Safi'i[a], Arief Setyanto[b], Suwanto Raharjo[c]
[a]Magister Teknik Informatika, Universitas Amikom Yogyakarta, nangimam75@gmail.com
[b]Magister Teknik Informatika, Universitas Amikom Yogyakarta, arief_s@amikom.ac.id
[c]Teknik Informatika, Fakultas Teknologi Industri, Institut Sains & Teknologi AKPRIND Yogyakarta, wa2n@akprind.ac.id

Abstract

Twitter is a social media that interacts through 140-character text-based tweet posts including photos, videos and hyperlinks. Spam tweets contain harmful messages sent continuously. Besides disturbing it is also dangerous for the recipient, exacerbated by the use of bots that automatically and quickly spread spam messages that can cause data damage. This study aims to detect spam bots by utilizing the similarity of tweets using Smith Waterman and the posting time interval. Data tweets are collected using scrap libraries in python in the form of id, text, time, link, based on datasets labeled as available. The data is carried out by text preprocessing steps to clean the text and then do the calculations. The calculation results of both the similarity method and the post time interval are then classified with k-Neaset Neighbor with the previous dataset that has been labeled to get the spam or legitimate bot prediction results. The results of classification experiments with several combinations of k to detect spam bots with similarity criteria and entropy interval obtained the best results k = 3 Neirest Neighbor and 10 fold Cross Validation with a predictive value of detection accuracy of 80%, 84% precission and 84% recall.

Keywords: Detect spam bots on Twitter, with waterman similarity and time interval entropy, k-NN classification for spammer predictions

Abstrak

Twitter merupakan media sosial yang berinteraksi melalui postingan *tweet* yang berbasis teks 140 karakter termasuk foto, video dan *hyperlink*. *Tweet spam* berisi pesan membahayakan yang dikirim secara terus-menerus. Selain mengganggu juga membahayakan bagi yang menerima, diperburuk dengan penggunaan *bot* yang secara otomatis dan cepat menyebarkan pesan *spam* yang dapat menyebabkan kerusakan data. Penelitian ini bertujuan mendeteksi *bot spam* dengan memanfaatkan kemiripan *tweets* menggunakan *Smith Waterman* dan *Interval* waktu *posting*. Data *tweets* dikumpulkan menggunakan *library* scrap di python berupa id, text, time, link, berdasarkan dataset berlabel yang telah tersedia. Data tersebut dilakukan tahapan *text preprocessing* untuk membersihkan teks kemudian dilakukan perhitungan. Hasil perhitungan dari kedua metode *similarity* dan *interval* waktu *posting* kemudian diklasifikasi dengan *k-Neaset Neighbour* dengan dataset sebelumnya yang telah berlabel untuk mendapatkan hasil prediksi *bot spam* atau *legitimate*. Hasil percobaan klasifikasi dengan beberapa kombinasi *k* untuk mendeteksi *bot spam* dengan kriteria *similarity* dan *interval entropy* diperoleh hasil terbaik *k=3 Neirest Neighbour* dan 10 *fold Cross Validation* dengan nilai prediksi deteksi *accuracy* sebesar 80%, *precission* 84% dan *recall* 84%.

Kata kunci : Deteksi *bot spam* di twitter, smith waterman similarity dan time interval entropy, klasifikasi k-NN untuk prediksi spammer

1. Pendahuluan

Masyarakat Indonesia merupakan pengguna terbesar ke 5 setelah USA, Brazil, Jepang dan Inggris, pada penggunaan platform media sosial twitter[1]. Twitter dengan pengguna lebih dari 500 juta dan 400 juta tweet perharinya, memungkinkan pengguna untuk berbagi pesan[1]. Pengguna Twitter menulis tentang berbagai opini, isu-isu yang sedang terjadi atau berbagi suatu produk menyebabkan para spammer mulai menyebarkan sejumlah besar pesan spam dengan tujuan komersialnya[2]. Tweet spam berisi pesan singkat atau link yang dikirimkan secara terus-menerus dan mengganggu pengguna yang menerima. Karakteristik tweet spam yaitu seringkali di posting secara otomatis dan teratur dalam waktu yang dekat dan tweet spam seringkali tidak memiliki ungkapan/ekspresi berbeda dengan pengguna asli yang mem-posting tweet yang memiliki ungkapan ekspresi. Tweet spam di perburuk dengan penggunaan program otomatis (bot)[3]. Bot spammer berbahaya bagi pengguna media sosial, tidak

Diterima Redaksi : 17-08-2017 | Selesai Revisi : 01-10-2017 | Diterbitkan Online : 02-11-2017

hanya berpotensi merusak tweet, juga dapat menyebabkan kerusakan data[4]. Twitter memiliki mekanisme untuk penanganan bot spammer dengan melaporkan, namun memiliki kelemahan apabila laporan pengguna Twitter yang dikumpulkan ternyata laporan palsu[5].

Penelitian yang akan dilakukan dalam deteksi bot spam ini menggunakan parameter tweets similarity dengan Smith Waterman karena belum ada yang menggunakan metode ini untuk deteksi kemiripan tweets. Adapun penelitan yang terdahulu menggunakan cosine similarity untuk kemiripan tweets dan membuang url didalam tweets[6]. Penelitan ini akan memanfaatkan *URL* pada tweets sebagai parameter dan metode similarity lain dalam deteksi bot. Selain kemiripan tweets peneliti memanfaatkan interval waktu *posting tweets* menggunakan *interval entropy*. Hasil perhitungan dari kedua metode tersebut kemudian diklasifikasi dengan metode k-Nearest Neighbour untuk memprediksi akun *bot spam* atau *legitimate* dengan dataset yang telah berlabel. Hasil klasifikasi di validasi dengan *k-fold cross validation* untuk mendapatkan nilai *accuracy*, *precission* dan *recall*.

Penelitian ini bertujuan untuk menunjukkan hasil proporsi akun bot spam atau legitimate user pada Twitter menggunakan pendekatan Tweets similarity dan time interval antar tweets. Performa klasfikasi k-NN untuk memprediksi akun bot spam atau legitimate dengan menggabungkan metode Smith Waterman dan Time Interval Entropy.

2. Tinjauan Pustaka

Deteksi antara bot spammer dan legitimate user menggunakan kombinasi metode kemiripan dengan cosine similarity dan waktu antar posting tweet, untuk *url* didalam tweets tidak digunakan, penelitian tersebut mendapatkan tingkat accuracy 85%, precission 94% dan recall 90%[7]. Dimana time stamp digunakan untuk menghitung interval antar tweet dan kemiripan tweet menggunakan unigram matching-based. Data tweet yang digunakan terdiri atas kumpulan akun normal dan akun yang terindikasi sebagai bot spammer yang sudah di kategorikan sebelumnya yang dihasilkan dari penelitianya lebih baik daripada penelitian yang menggunakan satu metode. Dalam penelitianya ada beberapa parameter seperti *URL* yang dibuang saat pre-processing, penelitian[6] menggunakan *URL* sebagai salah satu parameter untuk deteksi spam *campaigns*. Penelitian deteksi bot spammer dengan memanfaatkan fitur waktu dilakukan[8] fitur waktu posting dalam penelitian ini belum bisa mengidentifikasi tweet yang dilakukan berulang kali oleh *legitimate user*, sehingga *legitimate* dapat terinditifikasi sebagai *spammer*. Smith Waterman merupakan algoritma yang digunakan untuk menghitung kemiripan dua buah teks atau dokumen berdasarkan urutan. Algoritma ini mempunyai efek yang baik dalam pencocokan, menggunakan sub matriks yang berisi semua kemungkinan kesamaan, membandingkan nilai-nilai dari sub matriks hingga didapatkan nilai yang optimal[9][10]. Implementasi algoritma Smith Waterman dan Cosine Similarity untuk menghitung kemiripan teks berdasarkan urutan dan kemunculan kata[11]. *Preprocessing* merupakan pengubahan bentuk text yang terstruktur secara acak menjadi terstruktur sesuai kebutuhan, *Preprocessing* terdiri dari *case folding, tokenizing, removing punctuation, removing stop words, removing Link/URL*, dan *stemming*.

3. Metodologi Penelitian

Penelitian ini menggunakan metode eksperimen dimana peneliti mengkaji kemampuan *Natural Language Processing* dalam melakukan deteksi terhadap akun *spam*, sehingga dapat dikategorikan sebagai penelitian inovasi. Dataset yang digunakan berasal dari *Trend Micro's Web Reputation Technology* telah berlabel *spam* dan *legitimate*, kemudian dikumpulkan kembali sebanyak 2000 *Tweets* dengan proses *scrapping* dan pencarian *search API* mengalami pemrosesan text standar.

3.1 Pengumpulan Data

Peneliti menggunakan dataset yang telah disediakan sebanyak 40 akun, dan masing-masing akun diambil 50 Tweets dengan proses *scrapping*, kemudian peneliti membagi menjadi dua data, yaitu data *spammer* sebanyak 25 akun dan data *legitimate* sebanyak 15 akun, terkumpul 2000 tweets dari keseluruhan akun kemudian dilakukan pemrosesan teks standar Tweets.

3.2 Analisis Data

Data tweets dilakukan *standard text preprocessing* untuk membersihkan teks agar meningkatkan akurasi dalam deteksi *spammer* dan *legitimate*.

```
regex = re.sub('(?=RT)[^\s]+', '', regex)
regex = re.sub('(?=rt)[^\s]+', '', regex)
regex = re.sub('(?=https)[^\s]+', '', regex)
regex = re.sub('(?=http)[^\s]+', '', regex)
regex = re.sub('(?=www)[^\s]+', '', regex)
regex = re.sub('(?=WWW)[^\s]+', '', regex)
regex = re.sub('(?=PIC)[^\s]+', '', regex)
regex = re.sub('(?=pic)[^\s]+', '', regex)
regex = re.sub('(?=url)[^\s]+', '', regex)
regex = re.sub('(?=URL)[^\s]+', '', regex)
regex = re.sub('(?=bitly)[^\s]+', '', regex)
proses_regex = regex
from nltk.tokenize import word_tokenize
tokens = word_tokenize(proses_regex)
tokens = [w.lower() for w in tokens]
words = [word for word in tokens if word.isalpha()]
from nltk.corpus import stopwords
stop_words = set(stopwords.words('english'))
words = [w for w in words if not w in stop_words]
from nltk.stem.porter import PorterStemmer
porter = PorterStemmer()
stemmed = [porter.stem(word) for word in tokens]
```

Gambar 1. Preprocessing Data Tweets

Tahapan analisis data dengan *standart text processing* dengan bahasa python dan *library* dari nltk sebagai berikut :

a. Masukan Tweets asli dari setiap Tweets yang diproses seringkali bersifat noisy karena berupa URL atau HTML link, simbol, kode angka ASCII, tanda baca selain koma, titik, tanda seru dan tanda tanya, singkatan kata tidak baku, dan kata dalam bahasa asing. Pada penelitian ini, bahasa asing yang ditemukan tidak memiliki arti karena berfokus pada bahasa Inggris saja.

b. Tahapan menghilangkan url, www, http/s, pic, bitly, Simbol (#,RT,@), kode angka ASCII, dan Tanda Baca koma, titik, tanda seru, dan tanda tanya, kemudian Lower Case Folding melalui proses ini dari library nltk;

c. Tokenizing tahapan pemotongan berupa kata untuk setiap kalimat yang ada kemudian dipisahkan menjadi kata token dengan cara mendeteksi spasi yang ditemukan.

d. Stop Words Removal menghilangkan kata umum yang tidak memiliki pengaruh signifikan pada sebuah kalimat. Hal ini diselesaikan dengan melakukan proses import daftar stop word dari library nltk.

e. Stemming masukan teks yang sudah dipisahkan menjadi kata token kemudian akan mudah untuk diproses. Salah satunya adalah stemming yang berusaha mengembalikan setiap kata yang ditemukan kembali ke dalam bentuk baku.

3.3 Similarity Smith Waterman

Algoritma Smith Waterman merupakan algoritma klasik yang telah dikenal luas dalam bidang bioinformatika sebagai metode yang dapat mengidentifikasi local similarities (penyejajaran sequence) yaitu proses penyusunan dua *local sequences* (rangkaian/susunan atau rentetan) protein sequences sehingga kemiripan antara dua sequence tersebut akan terlihat. Berdasarkan fungsi proses penyejajaran sekuens tersebut, maka algoritma ini dapat digunakan dalam proses pendeteksian kemiripan tweets dari yang dianggap sebagai tweets spammer dengan cara melihat kemiripan antar tweets yang dipostingkan. Algoritma Smith Waterman sendiri banyak digunakan untuk menghitung penyelarasan lokal yang optimal[12][13].

```
A b c b a d b c a
| | | | | | | | |
A b – b – d b d a
```

Gambar 2. Optimal alignment dua substring

Dua urutan urutan kueri dan urutan basis data akan dibandingkan, didefinisikan sebagai A= $a_1a_2...a_n$ dan B= $b_1b_2...\ b_n$ jadilah urutan yang harus disesuaikan, dimana n dan m adalah panjang dari masing-masing A dan B.

1. Tentukan matriks substitusi dan skema penalti gap
   a. *s(a,b)* Nilai kesamaan elemen yang merupakan dua urutan
   b. W_k hukum dari celah yang memiliki panjang *k*
2. Buatlah matriks penilaian *H* dan inisialisasi baris pertama dan kolom pertama. Ukuran dari matriks penilaian adalah (n+1)*(m+1). Perhatikan pengindeksan berbasis 0

$$H_{k0} = H_{0l} = 0 \; for \; 0 \le k \le n \; and \; 0 \le 1 \le m \qquad (1)$$

3.4 Time Interval Entropy

Time interval entropy digunakan untuk menangkap pola keteraturan waktu posting tweets yang menunjukkan otomatisasi, TIE (H) dihitung dengan menggunakan persamaan (1) dan persamaan (2).

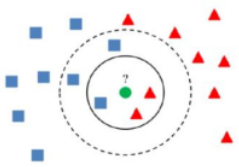
$$H_{\Delta T}(Ti) = -\sum_{i=1}^{nT} P\Delta T(\Delta t_i)\log(P\Delta T(\Delta t_i))$$

$$P\Delta T(\Delta t_i) = \frac{n\Delta t_i}{\sum_{k=1}^{nT} n\Delta t_k} \qquad (2)$$

Dimana ΔT merepresentasikan interval waktu antar tweets, dimana $P\Delta T(\Delta t_i)$ menunjukkan probabilitas interval waktu ΔT_i. Komponen *entropy* dapat mendeteksi waktu periodik yang merupakan indikasi kuat terjadinya otomatisasi. Penggunaan Twitter yang memiliki *entropy* lebih rendah dari *threshold* akan diklasifikasikan sebagai *bot spammer* karena nilai *entropy* rendah dibawah *threshold* menunjukkan perilaku yang teratur[2].

3.5 K-Nearest Neighbour

Klasifikasi *k-Nearest Neighbour* mencari sejumlah *k* objek data atau pola (dari semua pola latih yang ada) yang paling dekat dengan pola masukan, kemudian memilih kelas dengan sejumlah pola terbanyak diantara *k* pola tersebut. Penentuan *k* pola terdekat dilakukan berdasarkan ukuran jarak, similarity atau dissimilarity, bergantung jenis atributnya. Pada proses pengklasifikasian, algoritma *k-Neirest Neighbour* menggunakan keterangan sebagai nilai prediksi dari sampel uji yang baru, Jarak yang digunakan adalah jarak *Euclidean Distance*. Klasifikasi dua kelas menggunakan *k-Nearest Neighbour,* adapaun tahapan algoritma *ini adalah* :



Gambar 3. Klasifikasi *K-Nearest Neighbour*

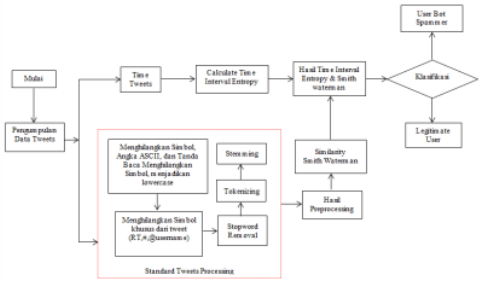
1. Untuk setiap pola latih <$x_f(x)$>, tambahkan pola tersebut ke dalam daftar pola latih
2. Untuk sebuah pola masukan x_q
   a. Misalkan $x_1, x_2 ..., x_k$ adalah *k* pola yang memiliki jarak terdekat (tetangga) dengan x_q
   b. Kembalikan kelas yang memiliki jumlah pola paling banyak diantara *k* pola tersebut sebagai kelas keputusan.

3.6 K-Fold Cross Validation

Pada penelitian ini data berjumlah 40 *user* kemudian dibagi menjadi data *training* dan *testing* dengan perbandingan 80:20. *K-Fold Cross Validation* pada penelitian ini ditentukan jumlah fold 10, untuk memperkirakan tingkat kesalahan yang terjadi, sebab data *training* pada setiap fold cukup berbeda dengan data *training* yang asli.

3.7 Alur Penelitian

Tahapan penelitian yang dilakukan dalam deteksi *bot spammer* dan *legitimate*.



Gambar 4. Alur Penelitian

4. Hasil dan Pembahasan

Dari beberapa hasil percobaan yang telah dilakukan dengan menentukan jumlah tetangga 1 sampai 5 pada klasifikasi dengan metode *Nearest Neighbour* maka peneliti mengambil kesimpulan bahwa *k=3* memiliki *accuracy* deteksi lebih tinggi. Peneliti menggunakan validasi dari beberapa percobaan klasifikasi dengan *K-Fold Cross Validation* sebanyak 10 dan 5 *fold*. Hasil percobaan tersebut menghasilkan tingkat *accuracy* pada *k=5* dan 10 *fold* sebesar 78%. Percobaan *k=5* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 73%. Percobaan *k=4* dan 10 *fold* mendapatkan tingkat *accuracy* sebesar 78%. Percobaan *k=4* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 73%. Percobaan *k=3* dan 10 *fold* mendapatkan tingkat *accuracy* sebesar 80%. Percobaan *k=3* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 75%. Percobaan *k=2* dan 10 *fold* mendapatkan tingkat *accuracy* sebesar 70%. Percobaan *k=2* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 73%. Percobaan *k=1* dan 10 *fold* mendapatkan tingkat *accuracy* sebesar 70%. Percobaan *k=1* dan 5 *fold* mendapatkan tingkat *accuracy* sebesar 73%.

Dari beberapa percobaan kombinasi *k* tetangga pada *Nearest Neighbour* dan *Fold Cross Validation* dapat diketahuin *k=3* dan 10 *fold* mendapatkan nilai *accuracy* 80% lebih besar pada deteksi *bot spammer* dengan parameter *similarity tweets* dan *interval entropy* antar *posting*.

4.1 Hasil Klasifikasi

Deteksi *bot spammer* dengan dengan parameter *similarity* dan *time interval entropy* dengan tetangga terdekat sebesar *k*=3 *Nearest Neighbour* dan 10 *Fold Cross Validation* menghasilkan prediksi

Tabel 1. Hasil k=3 *Nearest Neighbour* dan 10 *Fold Cross Validation*

User Id	Tweets Similarity	TIE Tweets	Label Asli	Label Klasifikasi
1	62	1.69	Spam	Spam
2	1	1.05	Spam	Spam
3	16	1.65	Spam	Spam
4	21	1.67	Human	Spam
5	34	1.69	Spam	Spam
6	1	1.67	Spam	Human
7	11	1.68	Human	Human
8	0	1.69	Human	Human
9	76	1.69	Spam	Spam
10	42	1.37	Spam	Spam
11	7	1.69	Human	Human
12	14	1.68	Human	Human
13	17	1.67	Spam	Spam
14	78	1.66	Spam	Spam
15	0	1.68	Human	Human
16	25	1.69	Human	Spam
17	15	1.67	Spam	Human
18	75	1.54	Spam	Spam
19	10	1.69	Human	Human
20	37	1.69	Human	Spam
21	29	1.69	Spam	Spam
22	1	1.68	Spam	Spam
23	14	1.69	Human	Human
24	28	1.68	Human	Spam
25	20	1.69	Spam	Spam
26	1	1.67	Spam	Spam
27	43	1.4	Spam	Spam
28	6	1.69	Human	Human
29	36	1.49	Spam	Spam
30	51	1.69	Spam	Spam
31	1	1.28	Spam	Spam
32	14	1.69	Human	Human
33	65	1.68	Spam	Spam
34	14	1.61	Spam	Human
35	69	1.53	Spam	Spam
36	1	1.69	Human	Human
37	68	1.68	Spam	Spam
38	1	1.67	Spam	Human
39	79	1.57	Spam	Spam
40	15	1.69	Human	Human

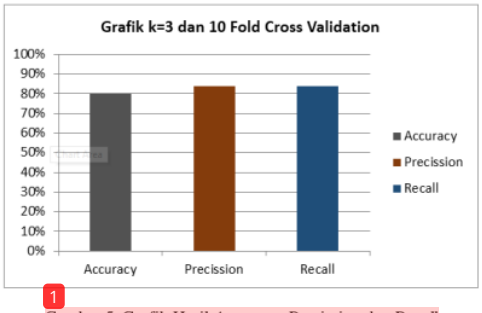
Hasil klasifikasi *K-Nearest Neighbour* dengan parameter *tweets similarity* menggunakan smith waterman dan waktru interval antar *posting tweets* menunjukkan tingkat *accuracy* deteksi *bot spammer* dan *legitimate* sebesar 80%, *precission* 84%, dan *recall* 84%. Sedangkan penelitian sebelumnya yang menggunaan cosine similarity dalam deteksi *similarity tweets* menunjukkan tingkat *accuracy* 85%, *precission* 94% dan *recall* 90%

Tabel 2. Accuracy, Precission dan Recall

Accuracy: 80.00%			
	True Spam	True Human	Precision
Pred. Spam	21	4	84.00%
Pred. Human	4	11	73.33%
Recall	84.00%	73.33%	

4.2 Grafik

Dibawah ini merupakan grafik hasil prediksi deteksi *bot spammer* dengan menggunakan k-Nearest Neighbour dan Cross Validation, lihat Gambar 5.



Gambar 5. Grafik Hasil Accuracy, Precission dan Recall

Gambar 5. menunjukkan tingkat *accuracy* terbaik dalam memprediksi akun *bot spam* dan *legitimate* sebesar 80%, *precission* 84%, dan *recall* 84%.

5. Kesimpulan

5.1 Simpulan

Berdasarkan hasil penelitian yang telah dilakukan dengan memberdayakan ketersediaan pokok pemikiran, dokumentasi, dan alat bantu dapat disimpulkan hasil deteksi *bot spammer* dengan *tweets similarity smith waterman* menghasilkan nilai yang hampir mirip antara pengguna keduannya. Akun *legitimate* memiliki rata-rata *similarity* kurang dari 28 kemiripan dan lebih dari sama dengan 1.68 pada interval waktu *posting tweets*. Sedangkan untuk akun *bot spam* memiliki rata-rata *similarity* lebih dari 28 kemiripan dan kurang dari 1.68 pada interval waktu posting. Sedangkan untuk prediksi *Bot Spammer* dan *legitimate user* menggunakan klasifikasi k-Nearest Neighbour menunjukkan tingkat *accurcy* sebesar 80%, *precission* 84%, dan *recall* 84% pada klasifikasi *k=3* dan *10 Fold Cross Validation*. Hasil yang didapatkan mempunyai selisih 5% dengan similarity yang menggunakan metode cosine simalirity, untuk penelitian selanjutnya bisa dikembangkan dengan menambahkan jumlah dataset dan menambahkan parameter lain dalam mendeteksi *bot spammer* seperti umur akun, jumlah pengikut, jumlah *tweet* dan *retweet*.

5.2 Saran

Suatu penelitian yang telah dilakukan merupakan awal dari penelitian selanjutnya. Tingkat hasil capain penelitian ini tentunya belum bisa dikatakan ideal untuk menjadi tolak ukur penelitian dengan pendekatan sejenis, namun dapat dijadikan sebagai dasar perbandingan untuk mendapatkan hasil yang lebih baik. Banyaknya alat bantu yang beragam dapat menarik peneliti untuk melakukan analisa dan perbandingan lebih mendalam agar menghasilkan sebuah kesimpulan tentang kesesuain alat bantu dengan kasus maupun skenario tertentu. Koleksi jumlah data yang lebih banyak perlu digunakan untuk meningkatkan akurasi dalam deteksi *bot spam*. Algoritma Smith Waterman yang digunakan untuk mendeteksi kesamaan dalam *Tweets* masih kurang cocok karena urutan huruf yang dihitung.

Daftar Rujukan

[1] W. Hidayat, "Kementrian Komunikasi dan Informatika Republik Indonesia," 2017. [Online]. Available: https://kominfo.go.id/content/detail/4286/pengguna-internet-indonesia-nomor-enam-dunia/0/sorotan_media.

[2] D. P. Christian Sri Kusuma Aditya., Mamluatul Hani'ah., Alif Akbar Fitrawan., Agus Zainal Arifin., "Deteksi Bot Spammer pada Twitter Berbasis Sentiment Analysis dan Time Interval Entropy," *J. Buana Inform.*, vol. 7, 2016.

[3] S. J. Zi Chu, Steven Gianvecchio, Haining Wang, "Who is Tweeting on Twitter: Human, Bot, or Cyborg?," in *Proceedings of the 26th Annual Computer Security Applications Conference*, 2010, pp. 21–30.

[4] H. L. Fred Morstatter., Liang Wu., Tahora H. Nazer., Kathleen M. Carley., "A New Approach to Bot Detection: Striking the Balance Between Precision and Recall," *IEEE*, 2016.

[5] Twitter, "Twitter," *Twitter*. .

[6] Hongyu Gao., Jun Hu., Christo Wilson., Zhichun Li., Yan Chen., Ben Y. Zhao., "Detecting and Characterizing Social Spam Campaigns," *ACM*, 2010.

[7] R. S. Perdana, T. H. Muliawati, and R. Alexandro, "Bot Spammer Detection in Twitter Using Tweet Similarity and Time Interval Entropy," *J. Ilmu Komput. dan Inf.*, vol. 8, no. 1, p. 19, 2015.

[8] Fahdi Washha., Aziz Qaroush., Florence Sedes., "Leveraging Time for Spammers Detection on Twitter," *ACM*, 2016.

[9] Vincentius Riandaru Prasetyo., Edi Winarko., "Rating Of Indonesian Sinetron Based On Public Opinion In Twitter Using Cosine Similarity," *IEEE*, 2016.

[10] R. I. Abdul Munif., Rizky Januar Akbar., Ruchi Intan Tantra., "Rancang Bangun Sistem E-Learning Pemrograman Pada Modul Deteksi Plagiarisme Kode Program Dan Student Feedback System," *J. Ilm. Teknol. Inf.*, vol. 15, 2017.

[11] A. R. Radiant Victor Imbar., Adelia., Mewati Ayub., "Implementasi Cosine Similarity dan Algoritma Smith-Waterman untuk Mendeteksi Kemiripan Teks," *J. Inform.*, vol. 10, 2015.

[12] Gotoh O, "An Improved Algorithm For Matching Biological Sequences," *J. Mol. Biol*, vol. 162, 1982.

[13] Smith T.F., Waterman M.S., "Identification Of Common Molecular Subsequencees," *J. Mol. Biol*, vol. 147, 1981.

2019_Resti_Bot_spam_SINTA2_2.pdf

ORIGINALITY REPORT

66%	66%	12%	12%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	jurnal.iaii.or.id Internet Source	63%
2	Submitted to Universitas Esa Unggul Student Paper	2%
3	juti.if.its.ac.id Internet Source	<1%
4	Saikin Saikin, Kusrini Kusrini. "MODEL DATA MINING UNTUK KAREKTERISTIK DATA TRAVELLER PADA PERUSAHAAN TOUR AND TRAVEL", Jurnal Manajemen Informatika dan Sistem Informasi, 2019 Publication	<1%
5	repository.ub.ac.id Internet Source	<1%
6	www.slideshare.net Internet Source	<1%

Exclude quotes	Off	Exclude matches	Off

Exclude bibliography On