SITEKIN: Jurnal Sains, Teknologi dan Industri, Vol. 20, No. 1, Desember 2022, pp.326 - 332
ISSN 2407-0939 print/ISSN 2721-2041 online

Model Sistem Monitoring URL Menggunakan Plugin Browser dengan Pendekatan NLP dan SDMIL untuk Perlindungan Anak

Esa Fauzi[1], Feri Sulianta[2], Yenie Syukriyah[3], Sy Yuliani[4]
Jurusan Teknik Inforamtika, Fakultas Teknik
Universitas Widyatama, Bandung, Indonesia
Email: esa.fauzi@widyatama.ac.id[1], feri.sulianta@widyatama.ac.id[2], yenie.syukriyah@widyatama.ac.id[3], sy.yuliani@widyatama.ac.id[4]

ABSTRAK

Internet adalah salah satu kemajuan teknologi yang berdampak ke banyak orang termasuk anak-anak. Dengan internet anak-anak dapat mengakses halaman web yang memiliki banyak informasi termasuk untuk mendukung pendidikan anak. Namun di internet banyak pula halaman web yang berisikan konten tidak pantas untuk dilihat anak-anak seperti pornografi. Bebasnya anak dalam mengakses konten halaman web ini menyulitkan orang tua untuk memantau perilaku anak di internet. Oleh karena itu pada penelitian ini diajukan sebuah model sistem monitoring untuk perlindungan anak dari konten negatif internet. Model sistem ini dikembangkan dengan arsitektur microservice dengan masing-masing *service* memiliki fungsi untuk mendeteksi konten negatif. *Service* pertama didukung dengan google API content filtering untuk menyaring konten website untuk orang dewasa. *Service* kedua dibuat dengan pendekatan NLP (Natural Language Processing) untuk menyaring tulisan-tulisan negatif. *Service* ketiga dibuat dengan pendekatan SDMIL atau Strongly Supervised Deep MIL (Multiple Instant Learning) untuk mendeteksi gambar dan video yang tidak cocok untuk anak-anak. *Service* keempat menyediakan layanan untuk kustomisasi URL negatif yang bisa diatur sendiri. Terakhir *service* kelima untuk log yang dapat membantu developer memantau kesalahan sistem. Model ini juga diintegrasikan dengan perangkat mobile sehingga orang tua dapat menerima laporan pengaksesan internet anak secara *real-time.*

Kata Kunci: API, microservice, Google API, NLP, SDMIL

ABSTRACT

The internet is one of the technological advances that has an impact on many people, including children. With the internet, children can access web pages that have a lot of information, including to support children's education. However, on the internet there are also many web pages that contain inappropriate content for children to view, such as pornography. The freedom of children to access the content of this web page challenges parents to monitor their children's behavior on the internet. Therefore, this research proposes a monitoring system model to protect children from negative internet content. This system model was developed with a microservice architecture where each service has a function to detect negative content. The first service is powered by Google API content filtering to filter website content for adults. The second service is made with the NLP (Natural Language Processing) approach to filter out negative writings. The third service is created using the SDMIL or Strongly Supervised Deep MIL (Multiple Instant Learning) approach to detect images and videos that are not suitable for children. The fourth service provides services for self-adjustable negative URL customization. Finally, the fifth service is for logs that can help developers deal with system errors. This model is also integrated with mobile devices so that parents can receive real-time reports on their child's internet access.

Keywords: *API, microservice, Google API, NLP, SDMIL*

Pendahuluan

Perkembangan internet memiliki banyak keuntungan salah satunya adalah untuk anak-anak. Dari internet, anak-anak bisa mendapatkan kesenangan, pendidikan dan hiburan [1]. Selain itu, akses untuk mendapatkan pengetahuan juga menjadi lebih mudah karena dapat ditemukan melalui URL web pada jaringan internet. Lebih jauhnya, internet juga memberikan tempat belajar alternatif untuk anak-anak selain di sekolah. Contohnya dalam hal ini adalah pembelajaran jarak jauh seperti e-learning. Pada saat pandemi e-learning menjadi salah satu alternatif belajar yang sangat membantu anak-anak.

Namun meningkatnya penggunaan internet oleh anak-anak, meningkat pula resiko yang akan dihadapi oleh anak-anak. Bahkan menurut Byron resiko internet pada anak sangat banyak dan bisa dibagi menjadi 4 jenis resiko, yiatu: resiko komersial, resiko gresif, resiko seksual, dan resiko nilai [2]. Semua resiko tersebut disebabkan karena anak-anak bisa mengakses konten internet dengan bebas. Oleh karenanya dibutuhkan solusi agar anak dapat terjaga dari konten internet yang beresiko.

Penyaringan Internet adalah salah satu solusi mengurangi resiko buruk internet terhadap anak-anak. Banyak negara mengimplementasikan penyaringan internet sebagai alat untuk mengatur konten-konten internet yang berbahaya dan ilegal[3]. Negara dalam hal ini biasanya bekerja sama dengan layanan penyedia internet untuk mengatur konten yang dapat diakses melalui internet. Namun peran untuk mengurangi resiko internet bukan hanya tugas negara, tetapi juga orang tua sebagai pendidik utama anak.

Orang tua memiliki peran utama dalam menjaga anak-anak dari resiko buruk internet. Dalam studi penelitian yang dilakukan oleh Samir N.Hamade [4], orang tua biasanya melakukan beberapa cara mediasi terhadap anak-anaknya. Beberapa cara umumnya adalah dengan adalah mendiskusikan penggunaan internet yang baik, berbicara tentang keamanan internet, memberitahu untuk menjauhi situs-situs berbahaya dan memantau perilaku anak di internet.

Salah satu cara memantau perilaku anak di internet yang dilakukan orang tua adalah dengan mengecek *history browser* URL. Pengecekan ini dilakukan untuk melihat kemungkinan ada URL yang mengandung konten tidak baik telah terlihat oleh anak. Namun cara ini sebenarnya tidak terlalu efektif karena anak bisa saja menghapus *history* URL dari *browser*-nya. Oleh karena itu dalam penelitian ini diajukan sebuah model yang dapat memberitahukan *history* URL browser secara *real-time* kepada orang tua tanpa perlu membuka browser yang dipakai anak. Pada model yang diajukan akan menggabungkan komunikasi antara plugin dan API.

Kemudian dari API akan bisa diintegrasikan dengan perangkat lain seperti perangkat android atau iphone, sehingga orang tua dapat memantau *history browser* anak secara *real-time*.

Proses penyaringan terhadap konten internet umumnya dapat dibagi menjadi 3 konten yaitu penyaringan terhadap konten teks, konten foto, dan konten video. Untuk konten teks umumnya penelitian menggunakan metode SVM (*Support Vector Machine*). Salah satu diantaranya adalah penelitian untuk *filtering* text terhadap konten dewasa dengan menggunakan SVM *machine learning*, algoritma pemilihan fitur IG dan log-TF, dan algoritma pengindeksan TFIDF sebagai metodenya [5]. Namun hasil dari penelitian ini masih dirasa kurang karena masih menghasilkan akurasi sekitar 80%. Lalu terdapat juga penelitian dengan metode SVM yang ditingkatkan. Penelitian oleh Jing Ouyang ini meningkatkan parameter data ketika eksperimen yang relevan dilakukan, sehingga efisiensi klasifikasi dan akurasi pengklasifikasi dapat meningkat secara signifikan [6]. Selain itu terdapat juga penelitian mengenai algoritma yang bisa dipakai untuk melakukan penyaringan teks terhadap konten dewasa, diantaranya adalah dengan menggunakan algoritma *No Semantic Accidental Injury Filter*(NSAIF). Algoritma ini dipakai sebagai pengganti algoritma Aho-2Corasick (AC) karena lebih memakan waktu dan ruang dalam proses semantik [7].

Dalam penyaringan terhadap konten foto, terdapat penelitian yang menggunakan metode *deep learning* yaitu *neural network* [8]. Metode ini ditingkatkan juga dengan metode transfer learning dengan mengukur efek dari metode pemrosesan gambar yang berbeda dan menentukan membuat pengklasifikasi yang ditingkatkan untuk domain konten dewasa. Lalu terdapat juga penyaringan konten foto dengan metode statistical [9]. Metode statistical berdasarkan model multi-color skin (MCSM) menghasilkan tingkat pengecekan sampai 89% dengan mengkombinasikan warna *simple visual cue* dan karakteristik geometrik badan manusia.

Penelitian dengan SVM ternyata juga dapat dipakai dalam penyaringan foto. Penelitian oleh Zhicheng Zhao [10] dan Kaikun Dong [11] adalah salah satunya. Penelitian oleh Zhao mengkombinasikan beberapa SVM klasifikasi untuk mendeteksi gambar dewasa lalu menguji terhadap 50.000 gambar di web dan menghasilkan 12.32% *false positive* dan 14.17% *false negative*. Sedangkan penelitian oleh Dong berdasarkan Bag of-Visual-Words dimana fitur gambar visual seperti tekstur dan bentuk lokal digabungkan dengan informasi teks yang diekstraksi dari nama file gambar, header file, atau halaman web kemudian pengklasifikasi SVM diterapkan untuk menyelesaikan klasifikasi citra. Hasil dari penelitan ini menghasilkan tingkat presisi sebesar 95,68%.

Berkembangnya internet menyebabkan konten video dapat juga di bagi melalui internet. Penelitian terkait filtering video dari konten yang tidak baik salah satunya adalah dengan *bimodal codebook* [12].

Metode ini dipakai untuk mengatasi permasalahan *performance* dari metode *multi-modally.* Metode *bimodal* ini menggabungkan representasi buku kode audio berbasis analisis periodisitas dan representasi buku kode visual berbasis analisis arti-penting yang kemudian menghasilkan pengecekan yang lebih baik dari beberapa metode lain.

Selain itu terdapat juga metode dengan menggunakan *neural network* [13] dan *statistical color* [14] yang bisa dipakai untuk penyaringan video. Pada penelitian dengan metode neural network, yang menghasilkan keberhasilan sampai 90%, mengkombinasikan warna kulit dan informasi *motion* yang kemudian diklasifikasikan dengan metode *neural network.* Sedangkan pada penelitian *statictical color* mencetak *generic color model* lalu membuat analisis statistik dengan sample model gambar dewasa. Hasil yang diperoleh cukup baik karena memiliki *performance* yang cukup tinggi.

Penyaringan dengan URL juga adalah salah satu langkah yang bisa dilakukan dalam penyaringan filter konten tidak baik di internet. Salah satunya adalah penyaringan terhadap mesin pencarian sehingga hanya menampilkan konten yang pantas untuk anak-anak. Deepshikha Patel [15] dan Junta Deniarja [16] contohnya mengkustomisasi mesin pencarian yang cocok dan aman untuk anak-anak. Mesin pencarian yang dibuat mengklasifikasikan halaman web menjadi kategori “aman” dan “tidak aman” dengan menggunakan metodologi OWPCM (*Objectionable Web Page Classification Method*) oleh Patel dan metode *Naïve Bayes* oleh Junta. Hasil yang diperoleh dari kedua penelitian ini adalah anak-anak hanya bisa membuka halaman web yang sesuai kategori saja.

Penelitian dengan filtering URL juga bisa dilakukan dengan teknologi blockchain seperti penelitian yang dilakukan oleh Obadah R. Hammoud [17]. Hammoud melakukan penelitian dengan menggunakan infrastuktur etherium dan mendemonstrasikan sistem daftar hitam/putih dengan penggunaan blockchain. Lalu hasil dari penelitian ini menyimpulkan bahwa metode berbasis blockchain yang diusulkan cocok untuk melindungi pengguna dan organisasi saat menjelajahi Internet.

Selain itu terdapat juga penelitian oleh Demirol [18] dengan melakukan *tracking log* terhadap url dan IP address. URL dan alamat IP yang dikenali berbahaya akan di blockir oleh sistem. Selama pengaksesan internet, sistem yang dibangun oleh Demirol menganalisis jenis dan waktu akses dari port dan alamat IP. Sistem kemudian memberitahu pengguna untuk menutup nomor port yang harus ditutup karena diketahui tidak aman. Kelebihan lain pada sistem yang dibangun ini dapat juga memberikan informasi jika ada pengalihan alamat web ke alamat IP tertentu sebagai pencegahan dari serangan *phising.*

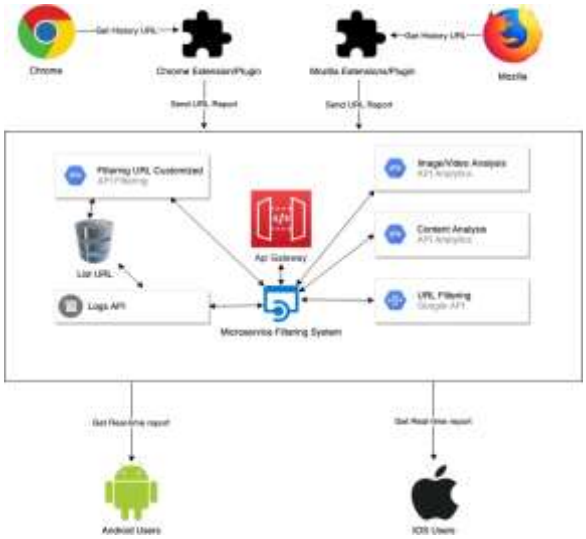
Penelitian lain oleh Chih-Chieh Chiu [19] juga mengembangkan sistem untuk melakukan penyaringan terhadap *website* yang mengandung konten yang tidak baik. Dalam sistem yang dibangun dibuat aplikasi yang bisa diunduh pada perangkat PC dan *mobile* (android & IOS). Aplikasi yang disebut NGA (*Network Guardian Angel*) ini terhubung dengan basis data yang menyimpan setidaknya 100,000 *website* yang terdaftar sebagai *website* tidak baik. Aplikasi yang dibuat di Taiwan ini bahkan telah di unduh sebanyak 100,000 sejak tahun 2007.

Metode Penelitian

Metode penelitian yang digunakan dalam penelitian ini adalah dengan studi literatur. Langkah awal yang dilakukan yaitu pengkajian ulang jurnal-jurnal, buku dan artikel yang terkait dengan topik penyaringan konten internet. Setelah pengkajian, penulis lalu membuat spesifikasi *requirement* (kebutuhan) dari model yang akan dibangun. *Requirement* tersebut kemudian dianalisis sehingga menghasilkan analisis model (berupa rancangan arsitektur). Dari analisis model, penulis kemudian membagi model tersebut ke dalam sub-sub bagian yang diperlukan dalam tahapan penyaringan konten internet. Lalu dalam beberapa sub bagian penulis merancang algoritma yang diperlukan di dalam sub bagian tersebut.

Hasil dan Pembahasan

Menurut Universitas Cambridge, *website* adalah satu set halaman di internet mengenai subjek tertentu yang dibuat oleh seseorang atau organisasi [20]. Dikarenakan subjek dari website dapat bermacam-macam seperti konten yang buruk, oleh karena itu penelitan ini bertujuan untuk membuat rancangan sistem penyaringan terhadap konten website yang tidak baik untuk anak-anak. Selain itu pada rancangan model yang diajukan juga dapat memberikan laporan kepada orang tua secara *real-time*.



Gambar. 1 Model Arsitektur URL monitoring system

Penelitian ini mengembangkan model integrasi antara plugin yang bisa terinstall pada browser dan API yang menyediakan fungsi penyaringan untuk menyaring konten dari internet. Untuk lebih jelas mengenai arsitektur model penelitian ini dapat dilihat pada gambar 1. Lalu dari gambar arsitektur tersebut kami bagi pendekatan model tersebut menjadi 6 bagian, yaitu Plugin Browser, Monitoring System, URL Filtering, Content Analysis, Logs API, dan Client Reporting App (Android/IOS App).

Plugin Browser

Untuk mendapatkan url *history* dari pengaksesan yang dilakukan anak-anak di internet, disini kami mengajukan pendekatan dengan membuat plugin/extensions. Plug-in, disebut juga add-on atau extension, adalah perangkat lunak komputer yang menambahkan fungsi baru ke program host tanpa mengubah program host itu sendiri [21]. Dalam penelitian ini plugin/extensions yang dimaksud adalah plugin/extensions yang menempel pada browser. Plugin pada browser dipilih karena dapat dengan mudah mendapatkan URL history yang telah diakses maupun sedang diakses oleh anak-anak. Contohnya adalah dengan menggunakan Chrome API untuk mendapatkan URL history. Dengan memanggil fungsi chrome.history dapat diperoleh URL yang telah dan sedang diakses pada browser chrome. Setelah mendapatkan url tersebut kemudian akan dikirim ke dalam API filtering untuk divalidasi apakah URL tersebut memiliki konten negatif atau tidak. Jika tidak ada konten negatif maka plugin akan mengijinkan URL untuk dibuka, namun jika tidak maka plugin akan melakukan block terhadap URL tersebut. Untuk lebih jelasnya dapat dilihat pada algoritma 1.

Dari algoritma 1 terdapat penggunaan *library* axios untuk melakukan *request* memvalidasi URL yang telah diperoleh dari *browser*. URL tersebut lalu dikirimkan melalui API gateway ke API monitoring system.

API *Monitoring System*

URL yang dikirim akan masuk melalui API gateway sebagai pintu utama sebelum diterima oleh API Monitoring System. API sendiri atau Application Programming Interface (API) adalah teknologi yang memfasilitasi produktivitas pengembang dengan memungkinkan penggunaan kembali komponen perangkat lunak [22]. Disini kami menyarankan menggunakan API gateway sebagai *service entry point* dari dunia luar karena API gateway berperan dalam mengamankan, melindungi, mengelola, dan menskalakan *request* API dengan mencegat dan menerapkan kebijakan seperti fungsi pembatasan dan keamanan [23]. Kami menyarankan menggunakan API gateway yang memiliki banyak fitur seperti AWS (Amazon web service), Alibaba Cloud, atau Apigee. Namun jika ingin yang gratis kami juga menyarankan open source seperti Kong atau WSO2 API gateway.

Setelah melalui API gateway, maka URL masuk ke dalam API monitoring system. API monitoring system sendiri bergantung kepada microservice lain, diantaranya URL Filtering Google API, Content Filtering, Image/Video Filtering API, Customized Filtering API, dan Logging API.

URL Filtering Google API

URL Filtering Google API adalah API yang melakukan penyaringan terhadap URL berdasarkan Google Content Filtering API [24]. Disini kami mengajukan API ini untuk memudahkan proses validasi karena jumlah website yang berbahaya bagi anak cukup banyak, oleh karena itu dibutuhkan penyedia layanan yang menyimpan data-data website berbahaya, dalam hal ini kami memilih Google Content Filtering API. Selain itu, Google Content Filtering API ini juga memiliki keunggulan dapat mengkategorikan 4 jenis website berdasarkan umur yaitu : High (G), medium (G dan PG), low (G, PG, dan PG-13), dan off (G, PG, PG-13, dan R). Jadi URL yang masuk ke dalam API monitoring system kemudian akan divalidasi menggunakan Google Content Filtering API. Jika website berbahaya maka akan mengembalikan nilai error. Jika tidak maka akan dilanjutkan ke tahap berikutnya.

Konten Teks Filtering API

Pada tahapan selanjutnya yaitu melakukan validasi dengan Content Filtering API. API ini adalah API yang memiliki fungsi untuk melakukan pengecekan apakah sebuah konten/isi dari suatu website memiliki kata-kata yang menjurus pada kata-kata yang tidak baik (contoh kata-kata porno atau terorisme). Pada API ini diterapkan pendekatan NLP

Algoritma 1. Contoh Proses Validasi URL dalam Plugin

```
import axios from 'axios'
const validateURL = 'https://example.api/validate'
//example api gateway url

function validateUrl() {
chrome.history.search({
    'text': '',              // Return every history item....
    'startTime': Now()  // that was accessed now
  },
  async function(historyItems) {
    // check history if its found
    if (historyItems !== undefined &&
historyItems.length >= 0){
      let url = historyItems[0].url;
      // check if url has appropriate content or not
      try {
        if (await axios.post(validateURL, url) {
          processWithURL(url)
        } else {
          blockURL(url)
        }
      } catch (error) {
        console.error(error)
      }
    }
  });
}
```

(*Natural Language Processing). Natural Language Processing* atau NLP ini adalah semantik, rekayasa perangkat lunak, dan kecerdasan buatan yang berhubungan dengan koordinasi antara komputer dan bahasa manusia untuk memproses dan menyelidiki informasi dari bahasa alami. Tujuannya adalah memprogram komputer untuk memahami teks tertulis, termasuk konteks bahasa di dalamnya [25].

Pada API ini dengan pendekatan NLP akan dibuat *corpus* / kumpulan kosakata untuk mengumpulkan kata-kata yang tidak baik. Kata-kata dari *website* akan dikumpulkan dengan menggunakan *library* NLTK [26] yang dibuat dalam bahasa pemrograman Python. Setelah ditemukan kemudian akan dilakukan proses tokenizing atau pemecahan menjadi kata-kata atau token. Token tersebut kemudian akan dilakukan proses *stemming* atau pemecahan imbuhan. Setelah itu kata-kata atau token yang tidak memiliki arti berarti akan dibuang. Lalu kemudian token yang memiliki arti sama (sinonim) akan dikategorikan sebagai token yang sama. Setelah itu akan dibuat corpus dan dihitung dalam bentuk *statistic* dari token-token yang telah di proses. Jika jumlah kata tidak baik terbilang ada banyak maka akan dibuat laporan yang nanti bisa dikirim ke orangtua (melalui perangkat android atau ios). Untuk lebih jelasnya proses NLP dapat dilihat pada gambar 2.

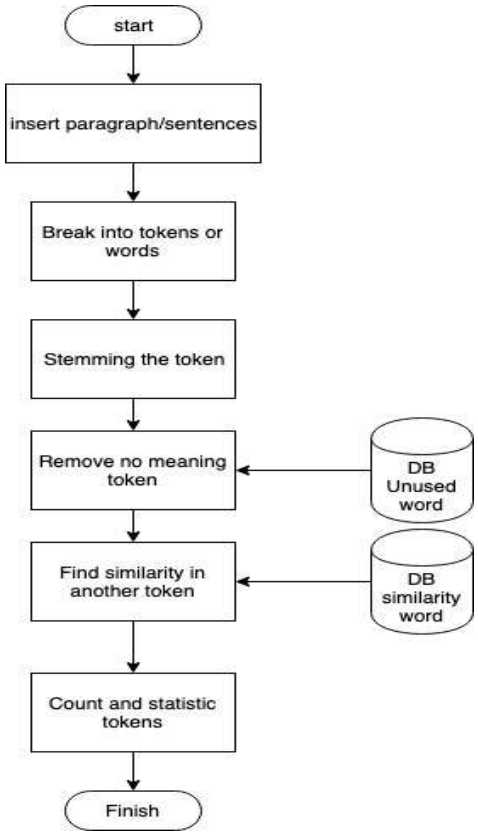


Fig. 2 Process NLP to Corpus

Image/Video Filtering API

API ini adalah API yang mendeteksi apakah gambar atau video termasuk dalam konten yang tidak baik seperti pornografi. Terdapat beberapa metode untuk mendeteksi suatu gambar merupakan gambar pornografi atau bukan. Pada penelitian ini digunakan pendekatan metode SDMIL oleh Wang [27]. Metode ini digunakan karena menurut perbandingan data oleh Abhishek Gangwar [28], metode ini memiliki keakurasian terbaik dalam mendeteksi pornografi yaitu 98.41%. SDMIL atau disebut dengan *Strongly Supervised Deep MIL (Multiple Instant Learning)* akan memodelkan setiap gambar sebagai sebuah kantong dari *instances* gambar yang *overlapping* kemudian melakukan pemodelan terlatih dari permasalahan MIL.

Sedangkan untuk mendeteksi video sendiri digunakan juga metode SDMIL, dengan cara mengubah video sebelumnya menjadi gambar. Pengubahan video sendiri menggunakan *library* transcoder dari Cloudmersive [29]. Untuk lebih jelasnya proses contoh algoritma pengubahan dan pengecekan video dapat dilihat pada contoh algoritma 2.

***Filtering* URL *Customized* API**

Filtering URL Customized API adalah API yang menyediakan list URL tertentu yang tidak boleh diakses oleh anak-anak. Pada API ini jika orangtua menginginkan membatasi website tertentu maka bisa dilakukan pada API ini. API ini terintegrasi dengan *database* yang akan menyimpan URL yang tidak boleh dilihat oleh anak-anak.

Logs API

Logs API adalah API yang akan mencatat log dari API filtering system. Jika API mengalami masalah maka akan tercatat melalui API ini. API ini pada dasarnya diperuntukan untuk developer apabila API mengalami permasalahan.

Client Reporting App

Client Reporting App dibuat untuk memberikan laporan pengaksesan internet anak secara *real-time*. Aplikasi yang dibuat adalah mobile aplikasi (android dan IOS) untuk memudahkan pemberitahuan

dimana dan kapan saja. Adapun secara garis besar *Client Reporting App* ini memiliki fitur:

- Memberikan laporan secara real-time jika anak-anak mengakses website yang baerbahaya atau tidak baik
- Menyediakan layanan untuk menyimpan yang tidak boleh dilihat oleh anak
- Memblock secara manual akses ke website yang dibuka oleh anak

Algoritma 2. Contoh algoritma proses pengubahan video menjadi gambar dan pengecekannya

```
var CloudmersiveVideoApiClient = require('cloudmersive-video-api-client');
var defaultClient =
        CloudmersiveVideoApiClient.ApiClient.instance;
var Apikey = defaultClient.authentications['Apikey'];
        Apikey.apiKey = 'YOUR API KEY';
var apiInstance = new
CloudmersiveVideoApiClient.VideoApi();
var opts = {
         'inputFile':
Buffer.from(fs.readFileSync("\\var\\temp\\inputfile").buffer),
// File | Input file to perform the operation on.
         'fileUrl': 'fileUrl_example', // String | Optional; URL of a video file being used for conversion. Use this option for files larger than 2GB.
         'maxWidth': 56, // Number | Optional; Maximum width of the output video, up to the original video width. Defaults to original video width.
         'maxHeight': 56, // Number | Optional; Maximum height of the output video, up to the original video width. Defaults to original video height.
         'framesPerSecond': 8.14 // Number | Optional; How many video frames per second to be returned as PNG images. Minimum value is 0.1, maximum is 60. Default is 1 frame per second. Maximum of 2000 total frames.
        };
        var callback = function (error, data, response) {
          if (error) {
            console.error(error);
          } else {
            var stats = []
            for (let i = 0; i<data.length; i++) {
              stats = SDMILCheckImage(data[i]); // check image is porn or not
            }
            if (isPornVideo(stats)) { // If summed up in the statistics the image is porn
              return 'the video was porn !!'
            }
          }
        };
        apiInstance.videoConvertToStillFrames(opts, callback);
```

Kesimpulan

Perkembangan internet memiliki keuntungan dan kerugian pada anak. Kemudahan mengakses suatu *website* menjadi suatu ketakutan untuk para orang tua. Bebasnya orang dalam membuat website dapat membuat anak-anak melihat konten yang tidak pantas. Dalam paper ini diajukan sebuah model untuk sistem yang dapat menjaga anak dari *website-website* yang berbahaya. Pada rancangan model yang dibuat terdapat 3 objek yang akan disaring. Pertama yaitu penyaringan terhadap URL yang tidak pantas dengan disupport oleh Google Filtering API. Kedua konten filtering teks dengan menggunakan NLP (Natural Language Processing) yang akan menyaring website dengan kalimat yang tidak pantas untuk anak-anak. Lalu terakhir penyaringan video & gambar dengan *Strongly Supervised Deep MIL (Multiple Instant Learning)* atau SDMIL yang bisa menghasilkan tingkat keakuratan hingga 98% dalam mendeteksi gambar porno. Selain itu pada rancangan model ini kami memberikan fitur tambahan yaitu Customized URL yang disimpan dalam basis data untuk menyimpan URL tertentu yang tidak boleh dilihat anak-anak, Logs API untuk menyimpan log dari sistem filtering ini, dan terakhir fitur laporan yang akan melapor aktivitas anak-anak ke gadget orang tua (android & iphone) secara *real*-time. Dari hasil rancangan model ini jika diimplementasikan diharapkan dapat membantu para orang tua untuk menjaga anak-anaknya dari *website* yang tidak pantas.

Daftar Pustaka

[1] N. Alqahtani, “A state of the art review of Internet risks on children,” *2017 2nd Int. Conf. Anti-Cyber Crimes, ICACC 2017*, pp. 108–112, 2017.

[2] T. Weru, J. Sevilla, J. Olukuru, L. Mutegi, and T. Mberi, “Cyber-smart children, cyber-safe teenagers: Enhancing internet safety for children,” *2017 IST-Africa Week Conf. IST-Africa 2017*, vol. 1, pp. 2–9, 2017.

[3] M. Eneman, “Internet filtering: A solution to harmful and illegal content?,” *Proc. - 2019 IEEE SmartWorld, Ubiquitous Intell. Comput. Adv. Trust. Comput. Scalable Comput. Commun. Internet People Smart City Innov. SmartWorld/UIC/ATC/SCALCOM/IOP/SCI 2019*, pp. 354–359, 2019.

[4] S. N. Hamade, “Parental Awareness and Mediation of Children’s Internet Use in Kuwait,” *Proc. - 12th Int. Conf. Inf. Technol. New Gener. ITNG 2015*, pp. 640–645, 2015.

[5] Y. Kim and T. Nam, “An efficient text filter for adult web documents,” *8th Int. Conf. Adv. Commun. Technol. ICACT 2006 - Proc.*, vol. 1, pp. 438–440, 2006.

[6] O. Jing, “Research on English Text

Information Filtering Algorithm Based on SVM," *Proc. 2020 IEEE Int. Conf. Power, Intell. Comput. Syst. ICPICS 2020*, pp. 1001–1004, 2020.

[7] D. Yan, J. Liu, and F. Yang, "Design and implementation of text filtering with no semantic accidental injury," *Proc. - 2011 4th IEEE Int. Conf. Broadband Netw. Multimed. Technol. IC-BNMT 2011*, pp. 61–65, 2011.

[8] S. C. Kalkan, B. Gozutok, A. Al Nahas, A. Kulunk, and H. Y. Erdinc, "Image Enhancement Effects on Adult Content Classification," *INISTA 2020 - 2020 Int. Conf. Innov. Intell. Syst. Appl. Proc.*, pp. 2–7, 2020.

[9] M. A. Mofadde and S. Sadek, "Adult image content filtering: A statistical method based on Multi-Color Skin Modeling," *2010 IEEE Int. Symp. Signal Process. Inf. Technol. ISSPIT 2010*, no. Icctd, pp. 366–370, 2010.

[10] Z. Zhao and A. Cai, "Combining multiple SVM classifiers for adult image recognition," *Proc. - 2010 2nd IEEE Int. Conf. Netw. Infrastruct. Digit. Content, IC-NIDC 2010*, pp. 149–153, 2010.

[11] K. Dong, L. Guo, and Q. Fu, "An adult image detection algorithm based on Bag-Of- Visual-Words and text information," *2014 10th Int. Conf. Nat. Comput. ICNC 2014*, pp. 556–560, 2014.

[12] Y. Liu, J. Ouyang, and J. Liu, "Bimodal codebooks based adult video detection," *2017 IEEE Glob. Conf. Signal Inf. Process. Glob. 2017 - Proc.*, vol. 2018-Janua, pp. 1397–1400, 2018.

[13] H. Bouirouga, S. E. Lrit, A. Jilbab, and D. Aboutajdine, "Recognition of adult video by combining skin detection features with motion information," *Int. Conf. Multimed. Comput. Syst. -Proceedings*, 2011.

[14] L. Yin, M. Dong, W. Deng, J. Guo, and B. Zhang, "Statistical color model based adult video filter," *Proc. 2012 IEEE Int. Conf. Multimed. Expo Work. ICMEW 2012*, pp. 349–353, 2012.

[15] D. Patel, V. Khan, R. K. Shukla, and M. Kherajani, "A customized children friendly and secure search engine," *2nd Int. Conf. Data, Eng. Appl. IDEA 2020*, pp. 2–6, 2020.

[16] J. Zeniarja *et al.*, "Search Engine for Kids with Document Filtering and Ranking Using Naive Bayes Classifier," *Proc. - 2018 Int. Semin. Appl. Technol. Inf. Commun. Creat. Technol. Hum. Life, iSemantic 2018*, pp. 560–564, 2018.

[17] O. R. Hammoud and I. A. Tarkhanov, "Blockchain-based open infrastructure for URL filtering in an Internet browser," *14th IEEE Int. Conf. Appl. Inf. Commun. Technol. AICT 2020 - Proc.*, 2020.

[18] D. Demirol, G. Tuna, and R. Das, "A simple logging system for safe Internet use," *IDAP 2017 - Int. Artif. Intell. Data Process. Symp.*, pp. 2–6, 2017.

[19] C. C. Chiu and C. S. Yang, "A defense tool to prevent inappropriate website on internet," *Proc. - 2019 Int. Conf. Intell. Comput. Its Emerg. Appl. ICEA 2019*, pp. 51–54, 2019.

[20] A. Jiwasiddi, R. P. N. Suci, R. T. Herman, and P. Weiss, "News website perceived quality; A comparative study for news websites in Indonesia," *Proc. 2016 Int. Conf. Inf. Manag. Technol. ICIMTech 2016*, no. November, pp. 325–328, 2017.

[21] J. Sterne, "plug-in | software | Britannica," *Encyclopedia Britannica*. [Online]. Available: https://www.britannica.com/technology/plug-in. [Accessed: 29-Nov-2022].

[22] G. Ajam, C. Rodriguez, and B. Benatallah, "API Topic Issues Indexing, Exploration and Discovery for API Community Knowledge," *Proc. - 2020 46th Lat. Am. Comput. Conf. CLEI 2020*, pp. 178–185, Oct. 2020.

[23] D. Geethika *et al.*, "Anomaly Detection in High-Performance API Gateways," *2019 Int. Conf. High Perform. Comput. Simulation, HPCS 2019*, pp. 995–1001, 2019.

[24] Google Inc, "Content filtering | Tenor | Google Developers." [Online]. Available: https://developers.google.com/tenor/guides/content-filtering. [Accessed: 29-Aug-2022].

[25] R. Kumar and V. Sahula, "Intelligent Approaches for Natural Language Processing for Indic Languages," *Proc. - 2021 IEEE Int. Symp. Smart Electron. Syst. iSES 2021*, pp. 331–334, 2021.

[26] "NLTK :: Natural Language Toolkit." [Online]. Available: https://www.nltk.org/. [Accessed: 30-Aug-2022].

[27] Y. Wang, X. Jin, and X. Tan, "Strongly-Supervised Deep Multiple Instance Learning," 2016.

[28] A. Gangwar, E. Fidalgo, E. Alegre, and V. González-Castro, "Pornography and child sexual abuse detection in image and video: A comparative evaluation," *IET Semin. Dig.*, vol. 2017, no. 5, pp. 37–42, 2017.

[29] Cloudmersive, "Video and Media Services API - Cloudmersive APIs." [Online]. Available: https://cloudmersive.com/video-and-media-services-api. [Accessed: 30-Aug-2022].