

PANDUAN PENGGUNA HOX (APLIKASI DETEKSI HOAKS)

by Haki_1_sucipto Sucipto

Submission date: 09-Dec-2020 10:01AM (UTC+0700)

Submission ID: 1469331434

File name: 18._Binder1.pdf (1.07M)

Word count: 727

Character count: 4440

REPUBLIK INDONESIA
KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

SURAT PENCATATAN CIPTAAN

Dalam rangka pelindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan : EC00201824183, 16 Agustus 2018

Pencipta

Nama : Aditya Gusti Tamam, Sucipto, , dkk

Alamat : Dsn. Gampeng Baru RT 05 RW 02, Ds. Gampeng, Kec. Gampengrejo, Kabupaten Kediri, Jawa Timur, 64182

Kewarganegaraan : Indonesia

Pemegang Hak Cipta

Nama : Universitas Nusantara PGRI Kediri

Alamat : Jl. KH. Achmad Dahlan, No 76, Mojoroto , Kediri, Jawa Timur, 64112

Kewarganegaraan : Indonesia

Jenis Ciptaan : Program Komputer

Judul Ciptaan : HOX: Aplikasi Deteksi Hoaks

Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia : 17 Mei 2018, di Kota Kediri

Jangka waktu pelindungan : Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.

Nomor pencatatan : 000114593

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.
Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.



a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA
DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL

Dr. Freddy Harris, S.H., LL.M., ACCS.
NIP. 196611181994031001

LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Aditya Gusti Tammam	Dsn. Gampeng Baru RT 05 RW 02, Ds. Gampeng, Kec. Gampengrejo
2	Sucipto	Jl. Jamsaren 1/9-D, Rt/Rw: 001/001, Kelurahan Jamsaren, Kecamatan Pesantren
3	Rini Indriati	Griya Intan Asri, Blok DB No 05, Rt/Rw: 001/003, Kelurahan Dermo, Kecamatan Mojoroto.





SISTEM INFORMASI
UNIVERSITAS NUSANTARA PGRI KEDIRI

Warning

hoax

Panduan Aplikasi Deteksi HOAKS

²
**Prodi Sistem Informasi
Fakultas Teknik
Universitas Nusantara PGRI Kediri**

PANDUAN PENGGUNA HOX (APLIKASI DETEKSI HOAKS)

Oleh: Aditya Gusti Tammam

- Untuk memulai proses deteksi hoaks, pertama perlu membuat *pipeline*. Pada daftar *pipeline* berikut, klik tombol **Create**.

The screenshot shows a table titled "Hox Pipelines" with a "Create" button at the top left. The table has two columns: "Reference" and "Status". There are ten entries, each starting with "HOX/2018/06/27/0001" followed by a date. The statuses are: Done, Done, Done, Keyphrase Generated, Similarity, Done, Done, Done, Done, Done.

- Kemudian masukan teks berita/isu yang hendak dideteksi ke dalam kolom yang telah disediakan.

The screenshot shows the "New" pipeline creation page. It has tabs at the top: Draft, Tokenized, Keyword Extracted, Keyphrase Generated, Search Result, Scrapped, Similarity, and Done. The "Raw Text" tab is selected. A text area contains the following text: "Kuasa allahh .. seorang syekh palestina yang sedang ibadah di masjidil aqsa ditembak oleh zionis israel .. dengan kuasa allah peluru sniper zionis yang sebesar jempol tidak bisa menembus kepala syekh .-".

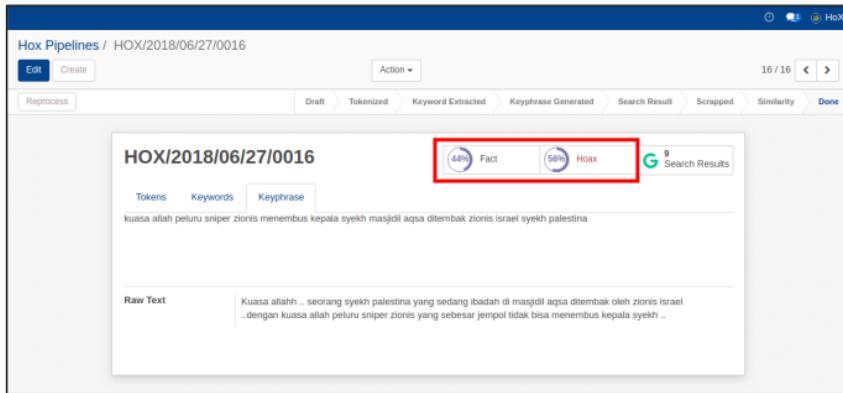
- Sistem akan membuat *keyphrase* berdasarkan teks tadi yang digunakan untuk mencari konten terkait di mesin pencari Google.

The screenshot shows the pipeline details page for "HOX/2018/06/27/0016". It has tabs at the top: Edit, Create, Action, Google Search, Regenerate Keyphrase, Draft, Tokenized, Keyword Extracted, Keyphrase Generated, Search Result, Scrapped, Similarity, and Done. The "Keyphrase Generated" tab is selected. Below it, there's a "Search Results" section with a "G 0" icon and a link to "Search Results". The text "kuasa alih peluru sniper zionis menembus kepala syekh masjidil aqsa ditembak zionis israel syekh palestina" is shown under the "Keyphrase" tab.

4. Setelah sistem mendapatkan konten terkait, maka akan ada proses dimana sistem mengukur tingkat kemiripan setiap konten tersebut dengan teks yang diawal tadi dimasukkan.

Name	External Link	Similarity Percentage (%)
#Disinformasi #Kepala #Syekh #Palestina... - Indonesian Hoaxes ...	https://www.facebook.com/TurnBackHoax/posts/1884905601561961?comment_id=183094229056699&comment_tracking=f%7B%22tr%22%3A%22R0%22%7D	9.73
Peluru Sniper Tertancap di Kepala Seorang Syekh Palestina ...	https://makassar.terkini.id/peluru-sniper-tertancap-di-kepala-seorang-syekh-palestina/	17.86
Kuasa allahh .. seorang syekh palestina saat ditembak zionis israel ...	https://www.youtube.com/watch?v=XM2588Zagz8	68.89
SUBHANALLAH ..ALLAHU AKBAR Kuasa Allah .. seorang Syekh ...		54.31
Foto Peluru Menancap di Kepala Bukan Hoax - PressReader	https://www.pressreader.com/indonesia/jawa-pos/20180612/281590946267350	26.65
Kepala Syekh Palestina Tidak Tembus Ditembak Sniper Zionis ...	https://www.uchnews.id/news/Kepala-Syekh-Palestina-Tidak-Tembus-Ditembak-Sniper-Zionis-Ternyata-Disinformasi-Faktanya/2517163175895996.html	20.87
Ternyata, Peluru Menancap di Kepala Pria Ini Bukan Hoax ...	https://www.jprn.com/news/ternyata-peluru-menancap-di-kepala-pria-ini-bukan-hoaks	21.70
Ternyata, Peluru Menancap di Kepala Pria Ini Bukan Hoaks KASKUS	https://www.kaskus.co.id/thread/5b1f1dafe2cf2cd3b8b4569/ternyata-peluru-menancap-di-kepala-pria-ini-bukan-hoaks/	26.13
Foto Peluru Menancap di Kepala Bukan Hoax Radar Pekalongan	http://radarpekalongan.co.id/34253/foto-peluru-menancap-di-kepala-bukan-hoax/	11.05

5. Dari konten yang paling mirip kemudian sistem akan melakukan analisa apakah konten-konten tersebut berasal dari sumber terpercaya atau tidak. Jika iya maka berpeluang fakta, namun jika sebaliknya maka berpeluang hoaks. Hasil perhitungan probabilitas tersebut tampil dalam bentuk persentase.



Lampiran 1

CONTOH SCRIPT

Contoh script proses *preprocessing*

```
def preprocessing(self):
    line_obj = self.env['hox.pipe.line']
    for pipe in self:
        pos_tags = pre_processing_text_with_polyglot(pipe.raw)[0]
        for index, tag in enumerate(pos_tags):
            line_obj.create({
                'pipe_id': pipe.id,
                'sequence': index + 1,
                'token': tag[0],
                'lower': tag[0].lower(),
                'pos_tag': tag[1]
            })
    pipe.state = 'tokenized'
return True
```

Contoh script proses *keyphrase generation*

```
def generate_keyphrase(self):
    for pipe in self:
        keywords = pipe.pipe_keyword_ids.mapped('keyword')
        keyphrase = keyphrase_generator(keywords, pipe.raw)
        pipe.keyphrase = keyphrase
        pipe.state = 'keyphrase'
    return True
```

Contoh script menghitung similaritas

```
def similarity(self):
    for pipe in self:
        search_results = pipe.google_search_ids
        for result in search_results:
            if result.text:
                text = result.text
            else:
                text = result.description
            result.similarity = compare_similarity(text, pipe.raw)
        pipe.state = 'similarity'
    return True
```

Contoh script perhitungan probabilitas

```
def process(self):
    google_search_obj = self.env['hox.google.result']
    media_obj = self.env['hox.media']
    company = self.env.user.company_id
    for pipe in self:
        search_result_ids = [id for id in
google_search_obj.search([('pipe_id', '=', pipe.id)]).mapped(
            lambda x: x.id if x.similarity_percent >=
company.similarity_tolerance else 0) if id != 0]
        search_results = google_search_obj.search([('id', '=', search_result_ids)])
        for result in search_results:
            web_link = result.extract_domains()
            for web in web_link:
                if web != '':
                    medias = media_obj.search([('website', 'ilike', web)])
                    for media in medias:
                        result.is_verified = media.is_verified
```

```
        verified_results_percentages = search_results.search(
            [ ('is_verified', '=', True), ('id', 'in',
search_result_ids) ]).mapped(
                'similarity_percent')
        unverified_results_percentages = search_results.search(
            [ ('is_verified', '!=', True), ('id', 'in',
search_result_ids) ]).mapped(
                'similarity_percent')
        total_result_percentage = sum(verified_results_percentages) +
sum(unverified_results_percentages)

        if total_result_percentage > 0:
            pipe.fact_probability = sum(verified_results_percentages) /
total_result_percentage
            pipe.hoax_probability = sum(unverified_results_percentages) /
total_result_percentage
            pipe.state = 'done'
        else:
            self.env.user.notify_warning("Return unknown result, please
input another text!",
                                         title="Not Enough Data",
sticky=True)
            pipe.state = 'done'
    return True
```

PANDUAN PENGGUNA HOX (APLIKASI DETEKSI HOAKS)

ORIGINALITY REPORT

3%

SIMILARITY INDEX

1%

INTERNET SOURCES

3%

PUBLICATIONS

0%

STUDENT PAPERS

PRIMARY SOURCES

1

docobook.com

Internet Source

1%

2

Teguh Andriyanto, Rini Indriati. "RESTFul Web Service Untuk Integrasi Data Antar Sistem Informasi", Jurnal Sains dan Informatika, 2017

1%

Publication

Exclude quotes

Off

Exclude matches

Off

Exclude bibliography

Off