

Implementasi Aplikasi *Speech to Text* untuk Memudahkan Wartawan Mencatat Wawancara dengan Python

I Komang Setia Buana

Politeknik Siber Dan Sandi Negara
e-mail: komang.setia@stsn-nci.ac.id

Diajukan: 8 April 2020; Direvisi: 19 Juni 2020; Diterima: 1 Juli 2020

Abstrak

Wawancara merupakan kegiatan komunikasi melalui proses pertukaran informasi antara reporter dan sumber berita. Banyak wartawan lebih sibuk mencatat pada saat wawancara sehingga hasil wawancara tidak efektif. Oleh karena itu dibutuhkan alat perekam untuk merekam jawaban dari narasumber. Akan tetapi alat perekam dibutuhkan waktu untuk mendengarkan hasil rekaman, kemudian mencatatnya. Padahal dituntut untuk mengumpulkan 3 sampai 4 berita dalam sehari. Dalam ilmu komputer terdapat bidang ilmu yaitu *Speech to Text*, teori ini akan bermanfaat untuk kondisi tersebut. *Speech to text* merupakan fitur untuk mengubah suara menjadi teks. Keunggulannya adalah layanan pengenalan suara. *Speech to Text* berfungsi untuk manajemen waktu agar lebih efektif. Umumnya kecepatan berbicara dengan kecepatan mencatat berbeda. Sehingga hal tersebut menyulitkan wartawan yang bertugas. Disisi lain, teori ini akan sangat berguna bagi kaum disabilitas. Karena user atau pengguna hanya perlu menggunakan suara untuk melakukan aktivitas mengetik selayaknya orang normal pada umumnya. Dari permasalahan tersebut, dibuatkan aplikasi yang bisa mengubah suara ke dalam suatu teks/tulisan dengan menggunakan bahasa pemrograman Python. Untuk melakukan proses mengubah suara menjadi teks menggunakan modul *speech recognition*. Uji coba menggunakan 6 sample audio hasil rekaman, didapat bahwa dengan menggunakan bahasa pemrograman Python, mampu mengonversi suara ke tulisan dengan tingkat keberhasilan mencapai 94,75 %.

Kata kunci: Wartawan, Wawancara, Python, *Speech to text*.

Abstract

Interviews are communications through the process of exchanging information between reporters and news sources. The journalist is required to write notes and take down crucial points in a short period of time, making the results of the interview otherwise ineffective. To combat these problems a recording device can be ideal in situations like these. Even though listening to a playback of the recording may take more time for a journalist to produce a finished and edited version of their interview. With journalists having to write 3 to 4 articles per day and having to conduct many interviews, you can see how this can become a difficult feat. This is where *speech to text* comes into play. *Speech to text* is a new and effective strategy that journalists use to help them convert the recording of an interview into proper text. Imagine how much more effective, in a shorter period of time, you can be if this software was used in the field. Another advantage of *speech to text* is its effectiveness in helping people who might lack proper writing skills but are proficient in speech. A useful application in converting speech into text is "Python". Python is one of the main applications that are responsible for converting speech into text by using a *speech recognition* module. To prove its effectiveness, Python conducted a trial using 6 recorded audio samples. The trial concluded that while using Python, they were able to convert speech into writing with a 94,75 % success rate.

Keywords: Reporters, Interviews, Python, *Speech to text*.

1. Pendahuluan

Wawancara atau *interview* merupakan kemampuan dasar jurnalistik yang sangat penting. Wawancara merupakan kegiatan komunikasi melalui proses pertukaran informasi antara reporter dan narasumber. Berita yang baik adalah berita yang dihasilkan dari sebuah penggalian informasi yang terampil. Oleh karena itu dibutuhkan jurnalis yang harus benar-benar fokus dan memperhatikan jawaban narasumbernya. Banyak wartawan yang pada saat melakukan wawancara tidak mampu fokus dengan

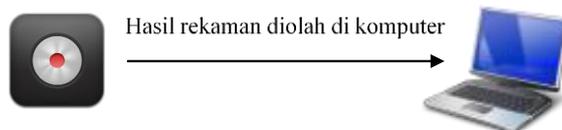
pernyataan yang dikemukakan oleh narasumber yang disebabkan karena fokus mencatat [1]. Oleh karena permasalahan tersebut, dibutuhkan alat perekam untuk merekam jawaban dari narasumber. Akan tetapi menggunakan alat perekam membutuhkan waktu lagi untuk mendengarkan hasil rekaman kemudian mencatatnya, padahal wartawan dituntut untuk mengumpulkan berita sebanyak 3 sampai 4 dalam sehari.

Di dalam ilmu komputer terdapat bidang ilmu yang disebut dengan *Speech to Text*. *Speech to Text* mengonversi teknis ke teks yang mengambil *input* dari *microphone* dalam bentuk suara kemudian dikonversi ke tulisan yang ditampilkan di desktop [2][3]. Pengenalan suara merupakan istilah ilmu komputer dan juga dikenal sebagai pengenalan suara otomatis. Ini adalah fitur untuk mengubah suara menjadi teks. Salah satu keunggulan utama untuk layanan pengenalan suara adalah pengurangan kesalahan ejaan kata-kata yang mungkin terjadi oleh seseorang saat mengetik. Disisi lain, penerapan *Speech to Text* akan sangat berguna bagi kaum disabilitas. Karena *user* atau pengguna hanya perlu menggunakan suara untuk melakukan aktivitas mengetik selayaknya orang normal pada umumnya. Keuntungan keseluruhan adalah manajemen waktu. Kebanyakan orang dapat berbicara lebih cepat daripada mereka melakukan pengetikan [4].

Dari permasalahan yang dialami oleh wartawan pada saat mengolah hasil rekaman yang dituangkan dalam bentuk tulisan, maka dibuatkan aplikasi yang bisa mengubah suara yang berasal dari narasumber ke dalam suatu teks/tulisan. Dalam pembuatan aplikasi penulis menggunakan bahasa pemrograman Python 3.8.1.

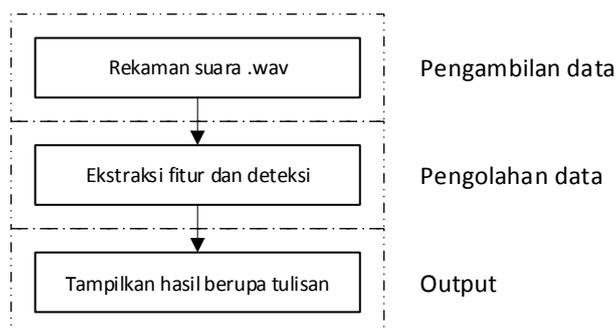
2. Metode Penelitian

Perancangan sistem dalam penelitian ini dibagi menjadi dua kegiatan yaitu pengambilan data dan pengolahan data. Pengambilan data adalah hal yang dilakukan pertama kali, setelah itu data yang sudah diambil diolah menggunakan aplikasi.



Gambar 1. Arsitektur Sistem

Gambar 1 memperlihatkan arsitektur sistem, di mana hasil rekaman yang sudah didapat diproses oleh aplikasi Python. Format *file* rekaman harus *.wav* karena aplikasi hanya bisa mendeteksi *file* format *.wav* saja. Alur kerja sistem dijelaskan seperti Gambar 2 berikut.



Gambar 2. Alur kerja sistem.

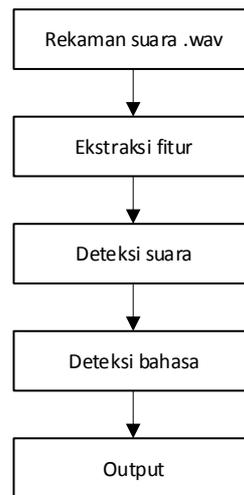
2.1. Pengambilan Data

Data didapat dari hasil rekaman suara orang atau merekam suara berita dari Youtube dengan menggunakan aplikasi *handphone*.

2.2. Pengolahan Data

Dalam melakukan pengolahan data, rekaman suara harus sudah disiapkan terlebih dahulu. Sebenarnya aplikasi yang dibuat bisa secara *realtime* mengubah suara ke teks dengan menyiapkan *microphone*. Akan tetapi karena aplikasi berupa desktop sehingga sangat menyusahkan kalau membawa komputer atau laptop ke mana-mana, sehingga lebih efisien hasil rekaman diolah langsung ke laptop.

Tahap pengolahan data untuk bisa mendeteksi suara dan menampilkan tulisan:



Gambar 3. Pengolahan data *output* tulisan.

Tahap pengolahan merupakan tahap inti dari sistem yang akan dirancang. Tahap pengolahan dibagi menjadi beberapa proses:

1. Tahap Ekstraksi Fitur

Ekstraksi Fitur adalah bagian terpenting dari pengenalan suara karena ia memainkan peran penting untuk memisahkan satu ucapan dari yang lain. Karena setiap suara memiliki karakteristik yang berbeda yang tertanam dalam *signal* suara [5]. Untuk ekstraksi fitur dilakukan perbaikan *noise* dengan menggunakan algoritma FastICA. FastICA merupakan suatu algoritma yang ditujukan untuk melakukan proses ekstraksi berdasarkan algoritma ICA. Metode ICA akan diterapkan untuk memisahkan antara suara asli atau suara baik dengan suara gangguan (*noise*), sehingga dapat diperoleh suara yang mempunyai kualitas yang lebih bagus. *File* yang akan diperbaiki sudah tercampur dengan *noise*, maka diasumsikan proses pencampuran *file* dengan *noise* adalah untuk membuat sebuah variabel yang mempunyai nilai BA (*dominan noise*). Prinsip ICA untuk memisahkan dua sumber bunyi adalah dengan diketahui kedua variabel sumber tercampur AB dan sumber tercampur BA. Dengan pencampuran dua sumber itu maka variabel yang dibutuhkan sudah terpenuhi, sehingga persamaan FastICA dapat dilakukan untuk mendapatkan nilai A dengan B. *File* yang rusak pada mulanya mempunyai nilai AB dengan persamaan ICA nilai B pada AB semakin direduksi sehingga didapatkan nilai A yang hampir *independent* [6][7].

2. Tahap Deteksi Suara

Tahap deteksi suara menggunakan bahasa pemrograman Python dengan menggunakan modul *speech recognition*. Untuk menggunakan modul *speech recognition*, di dalam *source code* import modul tersebut. Perintah awal yang diperlukan dalam *speech recognition* adalah Recognizer(). Karena akan menggunakan suara rekaman sehingga menggunakan *method* record() yang berada di Recognizer(). *Try & Except* digunakan untuk deteksi *error* yang terjadi pada saat proses deteksi suara.

3. Tahap Deteksi Bahasa

Pada tahap ini suara yang berasal dari narasumber dideteksi dan hanya menggunakan bahasa Indonesia. Ketika menggunakan bahasa Inggris atau bahasa lain, hasil tulisan yang didapat kurang sempurna karena di dalam *coding speech recognition* sudah diset ke bahasa Indonesia. Untuk melakukan deteksi bahasa harus terkoneksi internet karena modul *speech recognition* terhubung dengan Google Speech Recognition dengan sintak *recognize_Google* (*audio, language = 'in'*).

4. Output

Output yang dihasilkan berupa tulisan yang akan digunakan oleh wartawan untuk membuat berita.

2.3. Instrumen Penelitian

Adapun instrumen penelitian yang digunakan pada penelitian ini adalah:

1. Laptop Lenovo Y50 Intel Core i7
2. Software Python 3.8.1
3. Smartphone Android, OPPO A5
4. Windows 10 Enterprise

3. Hasil dan Pembahasan

Hasil dari aplikasi ini adalah untuk mendeteksi suara yang didapat dari rekaman suara orang atau rekaman dari Youtube. Penulis menggunakan suara Youtube karena di Youtube tersebut selain ada suara orang juga ada *backsound* musik yang menutupi suara orang tersebut. Sehingga suara musik tersebut bisa dijadikan *sample noise*. Penulis menggunakan suara orang langsung untuk menguji coba menggunakan suara langsung. Pada saat perekaman suara langsung penulis menghidupkan musik dan membuat suara gaduh dengan memukul-mukul meja untuk mendapatkan *noise* dari rekaman tersebut. *Sample* data yang digunakan ada 6 *sample* dengan tingkat *noise* sesuai yang dibuat oleh penulis yaitu:

- a. *Noise* sedikit jika suara orang lebih dominan dari suara *noise*.
- b. *Noise* banyak jika suara orang sama atau lebih kecil dari suara *noise*.

Tabel 1. *Sample* audio.

| No | Nama file | Durasi | Noise |
|----|-------------|------------------|---------|
| 1 | Sample1.wav | 17 detik | sedikit |
| 2 | Sample2.wav | 20 detik | banyak |
| 3 | Sample3.wav | 33 detik | sedikit |
| 4 | Sample4.wav | 1 menit | banyak |
| 5 | Sample5.wav | 1 menit 32 detik | banyak |
| 6 | Sample6.wav | 2 menit | sedikit |

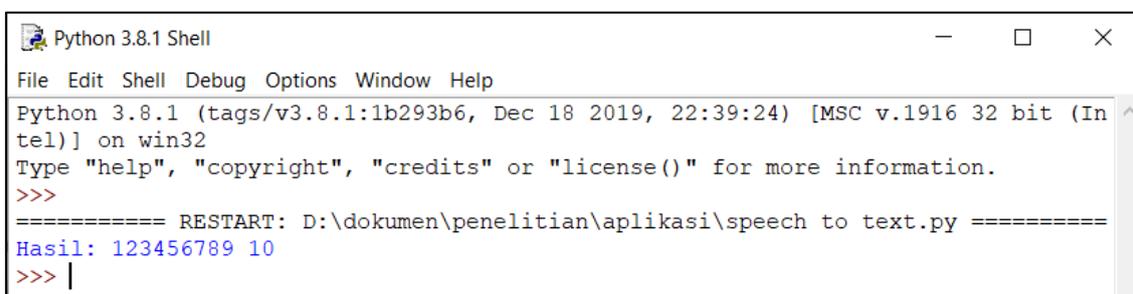
Dari *sample* audio yang digunakan menggunakan 6 *sample* dengan durasi yang berbeda-beda. Didapat hasil sebagai berikut:

Tabel 2. Hasil pengujian aplikasi.

| No | Nama File | Jumlah Kata | Kata yang Terdeteksi | Persentase |
|-------------------------------|-------------|-------------|----------------------|------------|
| 1 | Sample1.wav | 10 | 10 | 100 % |
| 2 | Sample2.wav | 32 | 28 | 87,5 % |
| 3 | Sample3.wav | 52 | 50 | 96,2 % |
| 4 | Sample4.wav | 78 | 70 | 89,7 % |
| 5 | Sample5.wav | 156 | 149 | 95,6 % |
| 6 | Sample6.wav | 186 | 185 | 99,5 % |
| Total persentase keberhasilan | | | | 94,75 % |

Penjelasan dari Tabel 2 seperti berikut :

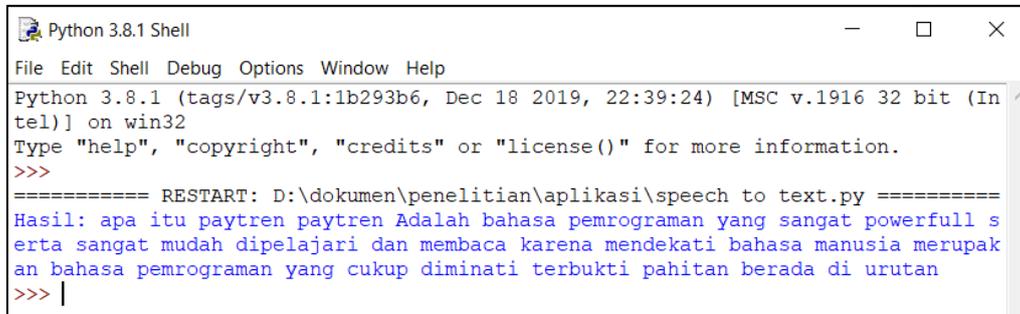
1. Menggunakan *file* sample1.wav dengan durasi 17 detik.
Sample audio yang digunakan dengan merekam suara sendiri.



Gambar 4. *File* audio durasi 17 detik.

Dari hasil yang terlihat pada Gambar 4, hasilnya berupa angka, di mana suara yang direkam berupa kalimat. Hasil dari aplikasi 100 % berhasil diterjemahkan ke teks dengan durasi 17 detik, sesuai dengan Gambar 4.

2. Menggunakan *file* sample2.wav dengan durasi 20 detik.
Sample audio yang digunakan dengan merekam suara di Youtube.

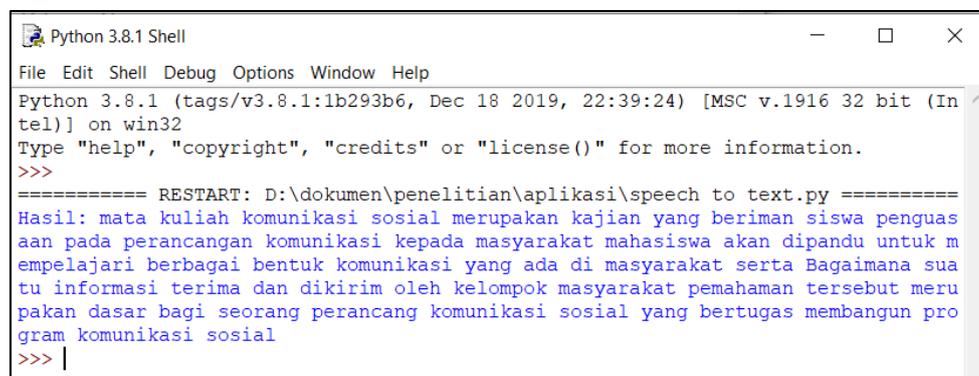


```
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 22:39:24) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:\dokumen\penelitian\aplikasi\speech to text.py =====
Hasil: apa itu paytren paytren Adalah bahasa pemrograman yang sangat powerfull serta sangat mudah dipelajari dan membaca karena mendekati bahasa manusia merupakan bahasa pemrograman yang cukup diminati terbukti pahitan berada di urutan
>>> |
```

Gambar 5. *File* audio durasi 20 detik.

Gambar 5 merupakan hasil yang didapat dengan durasi 20 detik. Isi berita dari audio sample2.wav yaitu “Apa itu Python? Python adalah bahasa pemrograman yang sangat *powerfull*, serta sangat mudah dipelajari dan dibaca, karena mendekati bahasa manusia. Python merupakan bahasa pemrograman yang cukup diminati, terbukti Python berada di urutan...”. Dari hasil pada Gambar 5, terlihat bahwa dari 32 kata dengan durasi 20 detik, dapat menerjemah secara benar sebanyak 28 kata. Dari data tersebut, hasil dari aplikasi 87,5 % berhasil diterjemahkan ke teks.

3. Menggunakan *file* sample3.wav dengan durasi 33 detik.
Sample audio yang digunakan dengan merekam suara sendiri.

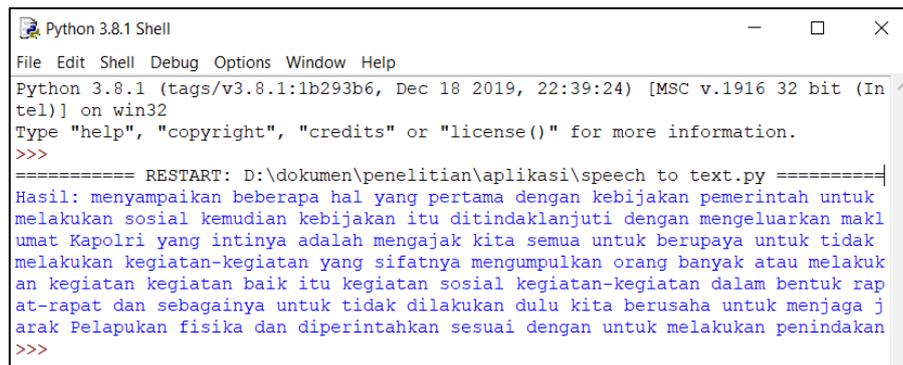


```
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 22:39:24) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:\dokumen\penelitian\aplikasi\speech to text.py =====
Hasil: mata kuliah komunikasi sosial merupakan kajian yang memberi mahasiswa penguasaan pada perancangan komunikasi kepada masyarakat mahasiswa akan dipandu untuk mempelajari berbagai bentuk komunikasi yang ada di masyarakat serta Bagaimana suatu informasi terima dan dikirim oleh kelompok masyarakat pemahaman tersebut merupakan dasar bagi seorang perancang komunikasi sosial yang bertugas membangun program komunikasi sosial
>>> |
```

Gambar 6. *File* audio durasi 33 detik.

Gambar 6 merupakan hasil yang didapat dengan durasi 33 detik. Isi berita dari audio sample3.wav yaitu “Mata kuliah komunikasi sosial merupakan kajian yang memberi mahasiswa penguasaan pada perancangan komunikasi kepada masyarakat. Mahasiswa akan dipandu untuk mempelajari berbagai bentuk komunikasi yang ada di masyarakat, serta bagaimana suatu informasi terima dan dikirim oleh kelompok masyarakat. Pemahaman tersebut merupakan dasar bagi seorang perancang komunikasi sosial yang bertugas membangun program komunikasi sosial.” Dari hasil pada Gambar 6, terlihat bahwa dari 52 kata dengan durasi 33 detik, dapat menerjemah secara benar sebanyak 50 kata. Dari data tersebut, hasil dari aplikasi 96,2 % berhasil diterjemahkan ke teks.

4. Menggunakan *file* sample4.wav dengan durasi 1 menit.
Sample audio yang digunakan dengan merekam suara di Youtube.



```

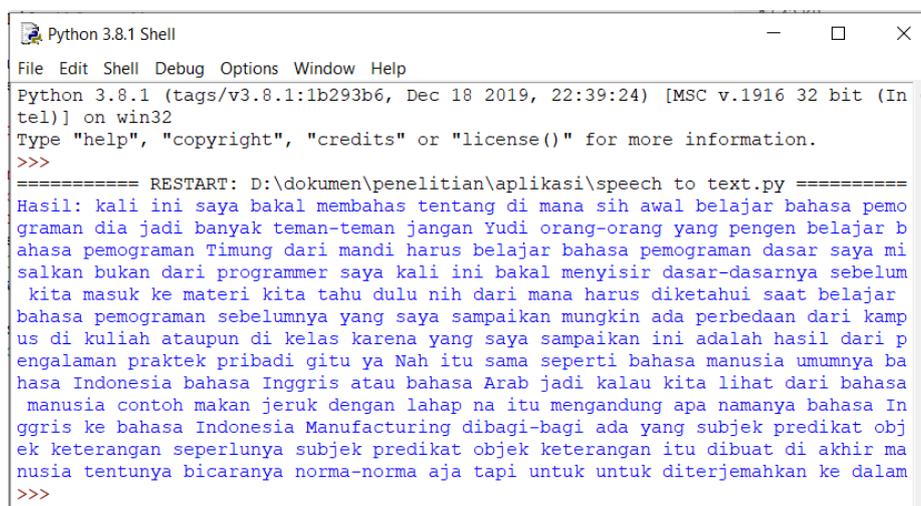
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 22:39:24) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:\dokumen\penelitian\aplikasi\speech to text.py =====
Hasil: menyampaikan beberapa hal yang pertama dengan kebijakan pemerintah untuk
melakukan sosial kemudian kebijakan itu ditindaklanjuti dengan mengeluarkan makl
umat Kapolri yang intinya adalah mengajak kita semua untuk berupaya untuk tidak
melakukan kegiatan-kegiatan yang sifatnya mengumpulkan orang banyak atau melakuk
an kegiatan kegiatan baik itu kegiatan sosial kegiatan-kegiatan dalam bentuk rap
at-rapat dan sebagainya untuk tidak dilakukan dulu kita berusaha untuk menjaga j
arak Pelapukan fisika dan diperintahkan sesuai dengan untuk melakukan penindakan
>>>

```

Gambar 7. file audio durasi 1 menit

Gambar 7 merupakan hasil yang didapat dengan durasi 1 menit. Isi berita dari audio sample4.wav yaitu “Menyampaikan beberapa hal yang pertama seiring dengan kebijakan pemerintah untuk melakukan *social distancing*, fisik *distancing* kemudian kebijakan itu ditindaklanjuti oleh Bapak Kapolri dengan mengeluarkan maklumat Kapolri yang intinya adalah mengajak kita semua untuk berupaya untuk tidak melakukan kegiatan-kegiatan yang sifatnya mengumpulkan orang banyak atau melakukan kegiatan, baik itu kegiatan sosial, kegiatan-kegiatan dalam bentuk rapat-rapat, dan sebagainya, untuk tidak dilakukan dulu, kita berusaha untuk menjaga jarak, melakukan fisik *distancing* secara disiplin dan diperintahkan sesuai dengan maklumat untuk melakukan penindakan”. Dari hasil pada Gambar 7, terlihat bahwa dari 78 kata dengan durasi 1 menit, dapat menerjemah secara benar sebanyak 70 kata. Dari data tersebut, hasil dari aplikasi 89,7 % berhasil diterjemahkan ke teks.

5. Menggunakan file sample5.wav dengan durasi 1 menit 32 detik.
Sample audio yang digunakan dengan merekam suara di Youtube.



```

Python 3.8.1 Shell
File Edit Shell Debug Options Window Help
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 22:39:24) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:\dokumen\penelitian\aplikasi\speech to text.py =====
Hasil: kali ini saya bakal membahas tentang di mana sih awal belajar bahasa pemograman
dia jadi banyak teman-teman jangan Yudi orang-orang yang pengen belajar b
ahasa pemograman Timung dari mandi harus belajar bahasa pemograman dasar saya mi
salkan bukan dari programmer saya kali ini bakal menyisir dasar-dasarnya sebelum
kita masuk ke materi kita tahu dulu nih dari mana harus diketahui saat belajar
bahasa pemograman sebelumnya yang saya sampaikan mungkin ada perbedaan dari kamp
us di kuliah ataupun di kelas karena yang saya sampaikan ini adalah hasil dari p
engalaman praktek pribadi gitu ya Nah itu sama seperti bahasa manusia umumnya ba
hasa Indonesia bahasa Inggris atau bahasa Arab jadi kalau kita lihat dari bahasa
manusia contoh makan jeruk dengan lahap na itu mengandung apa namanya bahasa In
ggris ke bahasa Indonesia Manufacturing dibagi-bagi ada yang subjek predikat obj
ek keterangan seperlunya subjek predikat objek keterangan itu dibuat di akhir ma
nusia tentunya bicaranya norma-norma aja tapi untuk untuk diterjemahkan ke dalam
>>>

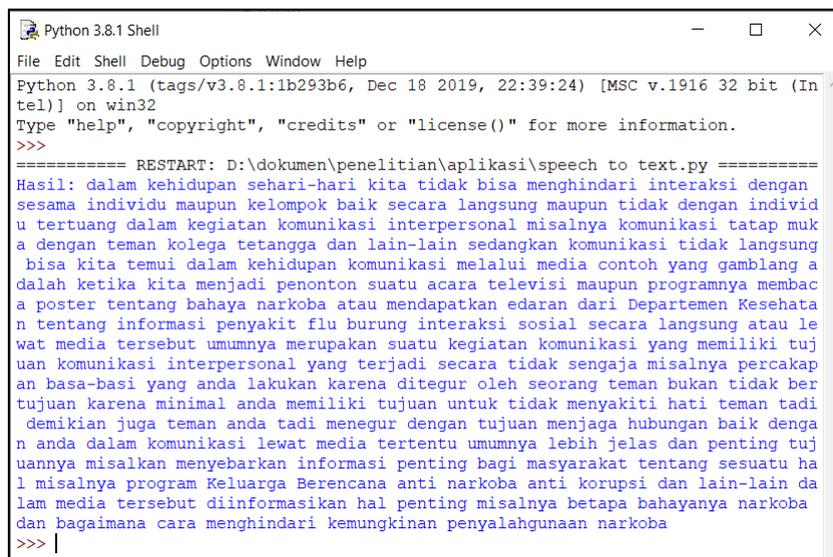
```

Gambar 8. File audio durasi 1 menit 32 detik.

Gambar 8 merupakan hasil yang didapat dengan durasi 1 menit 32 detik. Isi berita dari audio sample5.wav yaitu “Kali ini saya bakal membahas tentang di mana sih awal belajar bahasa pemograman, ini jadi banyak teman-teman yang *newbie*, orang-orang yang pengen belajar bahasa pemograman bingung dari mana nih harus belajar bahasa pemograman, dasar saya misalkan bukan dari *programmer*. Saya kali ini bakal menyisir, dasar-dasarnya sebelum kita masuk ke materi, kita tahu dulu nih dari mana harus diketahui saat belajar bahasa pemograman. Sebelumnya yang saya sampaikan mungkin ada perbedaan dari kampus di kuliah ataupun di kelas, karena yang saya sampaikan ini adalah hasil dari pengalaman *praktek* pribadi gitu ya, nah itu sama seperti bahasa manusia umumnya, bahasa Indonesia, bahasa Inggris, atau bahasa Arab, jadi kalau kita lihat dari bahasa manusia, contoh makan jeruk dengan lahap, nah itu mengandung apa namanya *grammar* kalau bahasa Inggris kalau bahasa

Indonesia mengandung strukturnya, dibagi-bagi ada yang subjek, predikat, objek, keterangan. Sebetulnya, subjek predikat objek keterangan itu dibuat di akhir, jadi manusia tentunya bicaranya norma-norma aja tapi untuk diterjemahkan ke dalam”. Dari hasil pada Gambar 8, terlihat bahwa dari 156 kata dengan durasi 1 menit 32 detik, dapat menerjemah secara benar sebanyak 149 kata. Dari data tersebut, hasil dari aplikasi 95,6 % berhasil diterjemahkan ke teks.

6. Menggunakan *file* sample6.wav dengan durasi 2 menit
Sample audio yang digunakan dengan merekam suara sendiri.



```
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 22:39:24) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:\dokumen\penelitian\aplikasi\speech to text.py =====
Hasil: dalam kehidupan sehari-hari kita tidak bisa menghindari interaksi dengan
sesama individu maupun kelompok baik secara langsung maupun tidak dengan individ
u tertuang dalam kegiatan komunikasi interpersonal misalnya komunikasi tatap muk
a dengan teman kolega tetangga dan lain-lain sedangkan komunikasi tidak langsung
bisa kita temui dalam kehidupan komunikasi melalui media contoh yang gamblang a
dahlah ketika kita menjadi penonton suatu acara televisi maupun programnya membac
a poster tentang bahaya narkoba atau mendapatkan edaran dari Departemen Kesehata
n tentang informasi penyakit flu burung interaksi sosial secara langsung atau lew
at media tersebut umumnya merupakan suatu kegiatan komunikasi yang memiliki tuju
an komunikasi interpersonal yang terjadi secara tidak sengaja misalnya percakap
an basa-basi yang anda lakukan karena ditegur oleh seorang teman bukan tidak ber
tujuan karena minimal anda memiliki tujuan untuk tidak menyakiti hati teman tadi
demikian juga teman anda tadi menegur dengan tujuan menjaga hubungan baik deng
n anda dalam komunikasi lewat media tertentu umumnya lebih jelas dan penting tuj
uannya misalkan menyebarkan informasi penting bagi masyarakat tentang sesuatu ha
l misalnya program Keluarga Berencana anti narkoba dan lain-lain da
lam media tersebut diinformasikan hal penting misalnya betapa bahayanya narkoba
dan bagaimana cara menghindari kemungkinan penyalahgunaan narkoba
>>> |
```

Gambar 9. *File* audio durasi 2 menit.

Gambar 9 merupakan hasil yang didapat dengan durasi 2 menit. Isi berita dari audio sample6.wav yaitu “Dalam kehidupan sehari-hari, kita tidak bisa menghindari interaksi dengan sesama individu maupun kelompok, baik secara langsung maupun tidak. Interaksi dengan individu tertuang dalam kegiatan komunikasi interpersonal, misalnya komunikasi tatap muka dengan teman, kolega, tetangga, dan lain-lain. Sedangkan komunikasi tidak langsung bisa kita temui dalam kehidupan komunikasi melalui media. Contoh yang gamblang adalah ketika kita menjadi penonton suatu acara televisi apa pun programnya, membaca poster tentang bahaya narkoba atau mendapatkan edaran dari Departemen Kesehatan tentang informasi penyakit flu burung. Interaksi sosial secara langsung atau lewat media umumnya merupakan suatu kegiatan komunikasi yang memiliki tujuan tertentu. Komunikasi interpersonal yang terjadi secara tidak sengaja, misalnya percakapan basa-basi yang Anda lakukan karena ditegur oleh seorang teman, bukan tidak bertujuan karena minimal Anda memiliki tujuan untuk tidak menyakiti hati teman tadi. Demikian juga teman Anda tadi menegur dengan tujuan menjaga hubungan baik dengan Anda. Dalam komunikasi lewat media tertentu umumnya lebih jelas dan penting tujuannya, misalkan menyebarkan informasi penting bagi masyarakat tentang sesuatu hal, misalnya program Keluarga Berencana, anti narkoba, anti korupsi, dan lain-lain. Dalam media tersebut diinformasikan hal penting, misalnya betapa bahayanya narkoba dan bagaimana cara menghindari kemungkinan penyalahgunaan narkoba”. Dari hasil pada Gambar 9, terlihat bahwa dari 186 kata dengan durasi 2 menit, dapat menerjemah secara benar sebanyak 185 kata. Dari data tersebut, hasil dari aplikasi 99,5 % berhasil diterjemahkan ke teks.

4. Kesimpulan

Kesimpulan yang dapat diambil dari penelitian ini adalah :

- Aplikasi untuk *speech to text* dengan menggunakan bahasa pemrograman Python berhasil dibuat.
- Modul *speech recognition* dan algoritma FastICA pada bahasa pemrograman Python, mampu mengonversi suara berupa rekaman suara orang dengan format .wav ke tulisan/teks, namun tanda baca tidak bisa dibuat oleh aplikasi ini. Dari hasil uji coba yang sudah dilakukan tingkat keberhasilan mencapai 94,75 %.

Daftar Pustaka

- [1] S. Ecip and dkk, *Teknik Mencari dan Menulis Berita*, 3rd ed. Tangerang Selatan: Universitas Terbuka, 2016.
- [2] P. Khilari and Prof. Bhope, "Implementation Of Speech To Text Conversion," *Int. J. Innov. Res. Sci. Eng. Technol.*, vol. 4, no. 7, 2015.
- [3] A. Trivedi, N. Pant, P. Shah, S. Sonik, and S. Agrawal, "Speech to text and text to speech recognition systems-Areview," *IOSR J. Comput. Eng.*, vol. 20, no. 2, 2018.
- [4] A. Katyal, A. Kaur, and J. Gill, "Automatic Speech Recognition: A Review," *Int. J. Eng. Adv. Technol.*, vol. 3, no. 3, 2014.
- [5] P. Khilari and P. Bhope, "A Review On Speech To Text Conversion Methods," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 4, no. 7, 2015.
- [6] I. B. Setiawan, M. Bezaleel, and A. S. Prasida, "Aplikasi Noise Reduction untuk Perbaikan Kualitas Suara pada Data Audio Menggunakan Algoritma FastICA," *J. Teknol. Informasi-Aiti*, vol. 8, p. 2, 2011.
- [7] N. Saideep, D. G. Kurup, and S. Tripathi, "Detection of closely spaced sinusoids in noise using FastICA algorithm," *Int. Conf. Adv. Comput. Commun. Informatics, ICACCI*, vol. 2017-Janua, pp. 305–309, 2017, doi: 10.1109/ICACCI.2017.8125858.