

---

# Meeting Assistant System Berbasis Teknologi Speech-to-Text

Daniel Soesanto<sup>1\*</sup>, Budi Hartanto<sup>2</sup>, Melisa<sup>3</sup>

<sup>1,2,3</sup>Program Studi Teknik Informatika, Universitas Surabaya, Surabaya, Jawa Timur  
Email: <sup>1\*</sup>daniel.soesanto@staff.ubaya.ac.id, <sup>2</sup>budi@staff.ubaya.ac.id, <sup>3</sup>s160416002@student.ubaya.ac.id

(Naskah masuk: 19 Okt 2020, direvisi: 18 Des 2020, diterima: 6 Jan 2021)

## Abstrak

Kegiatan rapat merupakan hal yang dilakukan secara berkala di setiap organisasi. Rapat tersebut diselenggarakan dengan tujuan untuk menyampaikan perkembangan tugas setiap divisi yang akan dicatat oleh seorang notulis. Pencatatan tersebut digunakan sebagai arsip organisasi yang biasanya disebut notulen rapat. Namun terdapat beberapa kendala yang dihadapi dalam penyusunan notulen rapat. Salah satu kendala tersebut misalnya notulis berhalangan hadir dalam rapat sehingga dibutuhkan orang yang menggantikan posisi notulis. Kendala lainnya adalah kecepatan mengetik atau menulis dari notulis yang belum tentu dapat menyamai kecepatan pembahasan dalam rapat, sehingga ada resiko informasi yang terlewat. Oleh sebab itu, dikembangkan sebuah sistem asisten rapat berbasis teknologi *speech-to-text* dari *Google Cloud API*. Tujuan dari pengembangan sistem ini adalah untuk mempermudah proses pembuatan notulen dengan melakukan ekstraksi kata-kata penting dari suatu rapat. Hasil pengujian menunjukkan bahwa sistem asisten rapat dapat menangani perekaman dan pengambilan percakapan penting secara ekstraktif. Sistem dapat mencapai akurasi tertinggi sebesar 81,3% ketika kualitas percakapan peserta memiliki struktur kalimat yang baik serta konteks pembahasan agenda terstruktur dengan baik.

**Kata Kunci:** notulen, *speech-to-text*, ekstraksi, asisten, rapat.

## *Meeting Assistant System Based on Speech-To-Text Technology*

### *Abstract*

*Meeting activities are things that are carried out regularly in every organization. The meeting was held to communicate the progress of each division's duties and would be recorded by a recorder of the minutes of a meeting. These records are used as organizational records, which are usually referred to as meeting minutes. However, there were several obstacles faced in preparing the meeting minutes. One of the obstacles is for example the recorder of the minutes of a meeting is unable to attend the meeting so that a person is needed to replace the position. Another obstacle is the typing or writing speed of the recorder of the minutes of a meeting which may not match the speed of the discussion in a meeting, so there is a risk of missing information. Therefore, a meeting assistant system based on speech-to-text technology from the Google Cloud API was developed. The purpose of developing this system is to facilitate the process of making minutes by extracting important words from a meeting. The test results show that the meeting assistant system can extractive handle the recording and retrieval of important conversations. The system can achieve the highest accuracy of 81.3% when the quality of the participant's conversation has a good sentence structure and a well-structured agenda discussion context.*

**Keywords:** *minutes of a meeting, speech-to-text, extraction, assistant, meeting.*

---

## I. PENDAHULUAN

Rapat merupakan kegiatan yang sering dilakukan di dalam suatu organisasi. Pada kegiatan ini setiap peserta akan berkumpul, bertukar informasi, serta berdiskusi untuk menyelesaikan suatu masalah tertentu. Walaupun sebenarnya menurut penelitian yang pernah dilakukan oleh Green et. al,

rapat termasuk kegiatan dengan tingkat produktivitas kecil yaitu maksimal hanya sebesar 47% [1]. Hal ini disebabkan oleh banyak hal, antara lain karena kurangnya partisipasi menyeluruh dari semua peserta rapat. Selain itu informasi yang ada di dalam rapat, seringkali juga tidak dapat diterima secara utuh oleh semua peserta rapat. Namun kegiatan rapat sampai saat ini masih saja terus dilakukan secara rutin dengan

memanfaatkan berbagai alat bantu, salah satunya adalah notulen rapat. Notulen rapat adalah suatu dokumentasi yang dibuat dengan tujuan untuk mencatat pembahasan penting selama rapat tersebut berlangsung [2]. Pencatatan dalam suatu rapat juga dapat dibedakan menjadi dua jenis, yaitu publik dan privat [3]. Notulen rapat biasanya lebih ke arah pencatatan publik, karena berisi hasil dari diskusi rapat, daftar pekerjaan yang harus dilakukan, serta keputusan apa yang dihasilkan dari rapat tersebut. Sedangkan yang dimaksud dengan pencatatan privat adalah berupa catatan pengingat yang biasanya dicatat setelah rapat berakhir.

Seseorang yang bertugas untuk membuat notulen rapat disebut notulis. Posisi tersebut dapat diisi oleh seorang sekretaris atau seorang yang ditunjuk menjadi asisten ketua suatu rapat. Namun, ketergantungan akan seseorang untuk mengisi posisi notulis dapat menimbulkan masalah bila orang yang bersangkutan berhalangan hadir dalam rapat. Masalah lain adalah ritme pembahasan suatu rapat terkadang tidak dapat diikuti oleh notulis sehingga beberapa poin pembahasan dapat terlewatkan. Menurut penelitian yang dilakukan oleh Nathan et. al. pada 2012, rata-rata terdapat 60% informasi yang hilang pada saat melakukan pencatatan rapat [4]. Dimana informasi yang hilang tersebut, terkait dengan keputusan yang diambil dalam diskusi rapat. Padahal notulen rapat sudah menjadi hal terpenting dalam suatu kegiatan rapat, karena berisi berbagai macam informasi yang dapat mempermudah peserta rapat dalam memahami atau mengulas kembali diskusi rapat yang telah dilakukan sebelumnya [4][5]. Selain itu notulen rapat juga bermanfaat bagi peserta yang berhalangan hadir, untuk tetap dapat memahami inti dari diskusi yang terjadi dalam rapat [6].

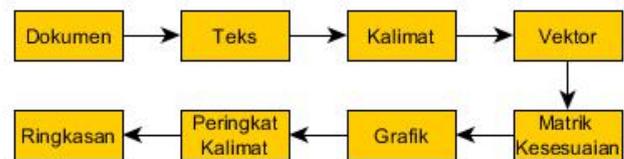
Pada penelitian terdahulu, Khan et. al., sudah pernah membuat sebuah sistem yang dapat menterjemahkan pembicaraan menjadi tulisan [7]. Namun karena tujuan dari sistem ini hanya ditujukan untuk mempermudah orang disabilitas, maka tidak dapat diimplementasikan pada perekaman pembicaraan rapat. Selain itu pada sistem ini juga tidak dilengkapi dengan fitur untuk melakukan ekstraksi kata-kata penting pada pembicaraan yang terjadi.

Pada tahun 2010, Tur et. al. juga melakukan penelitian dengan mengembangkan sebuah sistem asisten rapat yang diberi nama CALO. Pada sistem ini, dibuat pula sebuah ruangan rapat virtual, sehingga rapat dapat dilakukan secara *online* [8]. Selain itu Thompson et. al. pada tahun 2013 juga mempublikasikan penelitiannya mengenai pengembangan *meeting assistant system* yang diberi nama V-ROOM, yang dapat mencatat notulen dalam rapat [9]. Namun kedua sistem yang dikembangkan ini tidak dapat digunakan pada model rapat *offline*, dimana para peserta tetap berbicara secara langsung, dan yang dibutuhkan adalah perekaman dari pembicaraan tersebut.

Pada penelitian yang lain, digunakan algoritma *TextRank* untuk melakukan *automatic summarization* terhadap suatu dokumen atau teks. Kelebihan dari algoritma ini adalah tidak bergantung pada bahasa yang digunakan karena perhitungan bobot kalimat diperoleh dari *overlapping content* menggunakan konsep *bag of words*. Sedangkan untuk menghitung bobot kalimat, umumnya menggunakan metode

*content overlap similarity*, namun *cosine similarity* dapat menjadi alternatif lain untuk perhitungan bobot kalimat dalam teks [10]. Perhitungan tersebut akan diurutkan dari bobot kalimat tertinggi hingga terendah dan diambil sesuai parameter jumlah kalimat yang diinginkan pembuat sistem.

Tujuan dari penelitian ini adalah mengembangkan sebuah sistem yang dapat membantu notulis dalam menyusun notulen dengan menggunakan teknologi *speech-to-text*. Penggunaan teknologi tersebut bertujuan untuk merekam percakapan peserta rapat. Setelah itu mengubahnya menjadi teks yang nantinya akan diproses dengan menggunakan algoritma *TextRank*. Pada implementasinya, telah banyak terdapat *library* atau API untuk fitur *speech-to-text* [11]. Sehingga pada penelitian ini akan menggunakan *Google API*, untuk mempercepat pengembangan sistem. Algoritma *TextRank* yang terinspirasi dari *PageRank* [12], akan menghitung nilai setiap kalimat percakapan dalam suatu pembahasan rapat, seperti halnya Google menilai setiap halaman web di internet. Perhitungan dari algoritma ini menggunakan konsep *graph-based ranking* dimana suatu kalimat direpresentasikan sebagai grafik. Grafik ini akan memiliki hubungan dengan grafik lainnya berdasarkan *overlapping content* dengan grafik tersebut [13][14]. Selain itu, terkait dengan kemampuan meringkas pada algoritma ini, *TextRank* juga telah teruji cukup efektif meringkas deskripsi yang cukup panjang, seperti pada ulasan produk yang dijual online [15]. Hal tersebut yang menyebabkan algoritma *TextRank* ini sesuai apabila digunakan untuk meringkas hasil diskusi dari suatu rapat [16]. Sifat dari pengambilan informasi penting pada algoritma ini adalah ekstraktif, artinya algoritma ini tidak mengubah maupun menghapus satu katapun terhadap hasil pengambilan informasi penting dari suatu dokumen atau teks.



Gambar 1. Alur Proses Algoritma *Textrank*

Pada Gambar 1 ditunjukkan alur proses dari algoritma *textrank* untuk meringkas dokumen. Proses awal adalah seluruh teks dalam dokumen akan diambil dan disimpan dahulu. Setelah itu teks yang sudah diambil akan dipecah lagi menjadi kalimat-kalimat yang menyusun teks tersebut. Langkah berikutnya adalah setiap kalimat yang ada, akan dibentuk menjadi vektor yang merepresentasikan total kemunculan tiap kata yang ada pada kalimat tersebut. Hasil dari vektor tersebut akan digunakan sebagai perhitungan untuk melihat matriks kesesuaian antar kalimat berdasarkan *overlapping content* antar kalimat. Perhitungan nilai *similarity* antarkalimat dapat menggunakan rumus *content overlap similarity*, yaitu dengan menghitung berapa banyak kata yang sama antara 2 kalimat. Namun, metode *cosine similarity* juga dapat digunakan untuk menghitung nilai *similarity* dengan konsep perhitungan yang sama seperti perhitungan *content*

*overlap similarity*. Metode *cosine similarity* dapat dilihat pada Persamaan 1.

$$Sim(v_i, v_j) = \frac{\sum_{k=1} W_{ki} \times W_{kj}}{\sqrt{\sum_{k=1} W_{ki}^2 \times \sum_{k=1} W_{kj}^2}} \quad (1)$$

Dimana  $v_i$  adalah vektor dari kalimat ke  $i$  sedangkan  $v_j$  adalah vektor dari kalimat ke  $j$  sehingga  $W_{ki}$  dan  $W_{kj}$  menyatakan bobot kata pada kalimat ke  $i$  dan ke  $j$  yang diperoleh dari  $v_i$  dan  $v_j$ . Proses berikutnya adalah merepresentasikan semua kalimat yang ada menjadi grafik yang terdiri dari banyak simpul. Untuk menentukan simpul antar kalimat, dilakukan perhitungan berdasarkan *overlapping content* antar kalimat sehingga didapatkan nilai similarity suatu kalimat terhadap kalimat lainnya untuk menjadi acuan nilai *voting* suatu kalimat yang dibentuk dalam *similarity matrix*. Perhitungan setiap elemen matriks antar kalimat menggunakan Persamaan 2.

$$WS(V_i) = (1 - d) + \left( d \times \sum_{V_j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j) \right) \quad (2)$$

Dimana  $WS$  menyatakan bobot kalimat. Untuk  $w$  adalah nilai similarity sehingga  $w_{ji}$  adalah nilai similarity dari kalimat ke  $j$  yang berdampingan dengan kalimat ke  $i$ . Sedangkan  $w_{jk}$  adalah nilai *similarity* dari kalimat ke  $j$  yang berdampingan dengan kalimat ke  $k$ . Untuk  $d$  adalah *damping factor*, dengan probabilitas  $(1-d)$  untuk mendapatkan kalimat yang tidak memiliki hubungan dengan kalimat lainnya. Umumnya, nilai  $d$  adalah 0,85 atau artinya 85% probabilitas hubungan antar kalimat. Hasil perhitungan tersebut akan ditampung dalam bentuk matriks kembali dengan melakukan perhitungan secara berulang. Selanjutnya, nilai matriks yang baru akan diurutkan mulai dari yang tertinggi untuk menjadi acuan kalimat penting dari suatu dokumen. Acuan tersebut akhirnya dapat digunakan untuk *text summarization* dalam suatu dokumen.

## II. METODOLOGI PENELITIAN

Metode penelitian dilakukan dengan menganalisis kebutuhan rapat, mengidentifikasi masalah yang terjadi dalam pembuatan notulen rapat, dan menganalisis kebutuhan sistem berdasarkan masalah tersebut. Pada analisis kebutuhan rapat dilakukan berdasarkan budaya dalam rapat dengan melakukan wawancara terhadap 2 narasumber di Universitas Surabaya. Dari hasil wawancara, diperoleh 3 persamaan terhadap pelaksanaan rapat di lingkungan Universitas Surabaya dimana 2 diantaranya memiliki masalah yang terjadi dalam pelaksanaan rapat.

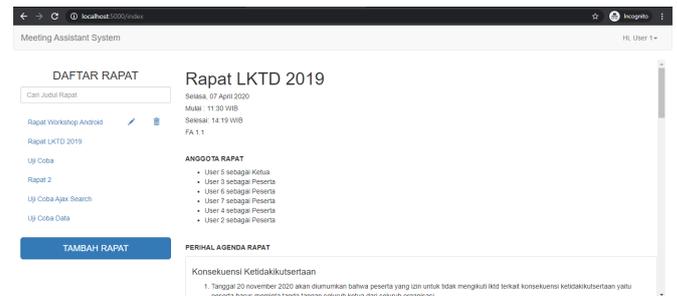
Permasalahan pertama adalah mekanisme komunikasi. Terdapat beberapa poin pembahasan dari setiap agenda rapat yang mudah terlewatkan. Hal ini dikarenakan kecepatan notulis dalam merangkai narasi notulen tidak seirama dengan pembawaan setiap agenda yang dilakukan oleh ketua rapat. Sedangkan yang kedua adalah masalah notulen rapat. Hasil notulen rapat nantinya akan dibahas ulang untuk mengecek kesesuaian dari pembahasan selama rapat akibat dari permasalahan pertama. Selain itu, permasalahan lainnya adalah penyimpanan notulen rapat yang tidak terorganisir sehingga mudah hilang.

Mengacu pada permasalahan tersebut, maka dibutuhkan sistem yang dapat merekam pembahasan setiap agenda rapat selama rapat berlangsung. Kemudian dari hasil rekaman tersebut, diharapkan sistem dapat mengambil percakapan penting dalam setiap agenda pembahasan rapat. Pengambilan percakapan penting ini diharapkan dapat membantu notulis dalam menyusun notulen. Selain itu dibutuhkan pula sistem yang dapat menyimpan hasil pembahasan rapat dalam setiap agenda. Hal ini dilakukan dengan cara mengambil percakapan penting di setiap agenda pembahasan pada suatu rapat.

## III. HASIL DAN PEMBAHASAN

Implementasi sistem dilakukan dengan menggunakan framework *Flask*. Sedangkan implementasi algoritma dan penyederhanaan struktur kalimat dengan menggunakan *library NLTK, spaCy*, dan model *treebank* bahasa Indonesia dari *Universal Dependencies*.

Tampilan dari detail rapat yang akan dilaksanakan, dapat terlihat pada Gambar 2. Pada halaman web tersebut, terlihat bahwa terdapat daftar rapat yang telah dibuat, serta detail rapat yang ada. Detail rapat berisi nama rapat, waktu pelaksanaan rapat, anggota rapat yang diundang, serta perihal agenda rapat yang akan dibahas. Bagian agenda rapat merupakan salah satu yang terpenting, karena akan berdampak pada ekstraksi notulen yang dihasilkan.



Gambar 2. Halaman Detail Rapat

Pada proses rekaman, ketika pengguna tidak berbicara selama 3 detik maka sistem akan menghentikan rekaman secara otomatis. Hal ini dilakukan untuk mengatasi hasil transkripsi yang tidak dapat menambahkan tanda baca, karenanya 1 file rekaman dianggap sebagai 1 kalimat. Ketika pengguna mulai berbicara lagi, pengguna harus menekan tombol bicara lagi untuk merekam pembicaraannya. Bila sistem sedang merekam percakapan, tombol bicara akan dinonaktifkan dan tampilan ikon *microphone* akan berubah menjadi warna merah terang. Bila sistem mendeteksi diam selama 3 detik, maka sistem akan secara otomatis mengaktifkan kembali tombol bicara dan tampilan ikon *microphone* akan berubah menjadi warna merah gelap. Perbedaan tampilan rekaman tersebut dapat dilihat pada Gambar 3.



Gambar 3. Tampilan Tombol Rekaman Pengguna

Setelah pembahasan suatu agenda telah selesai, maka ketua rapat akan menutup agenda dengan menekan tombol tutup agenda yang telah disediakan dibawah tombol rekaman. Sistem nantinya akan mengelompokkan kalimat-kalimat pada pembahasan suatu agenda dan menghitung nilai *similarity* antar kalimat. Selanjutnya, sistem akan mengambil 50% dari total kalimat pembahasan yang memiliki nilai tertinggi dan hasilnya akan disimpan pada *database*. Hal ini dikarenakan 50% sisanya, memiliki relevansi yang rendah terhadap inti dari percakapan agenda tersebut. Hasil pengambilan percakapan penting nantinya akan ditampilkan pada halaman rapat aktif yang dapat dilihat pada Gambar 4.



Gambar 4. Hasil Pengambilan Percakapan Rapat

Uji coba dan evaluasi dilakukan dengan 2 tahap yaitu verifikasi dan validasi untuk melihat apakah sistem sudah menjawab permasalahan yaitu memudahkan proses pembuatan notulen rapat. Pada tahap verifikasi terhadap *text summarization*, sistem akan mengambil 50% dari total kalimat pembahasan setiap agenda rapat. Kalimat pembahasan ini didapatkan dari hasil terjemahan *speech-to-text* dari API dan disimpan dalam *database* dengan pengelompokan per agendanya. Pada proses pengambilan kalimat penting menggunakan algoritma *TextiRank*. Sistem akan mengambil 50% dari total kalimat pembahasan yang memiliki nilai *similarity* antar kalimatnya dengan peringkat tertinggi. Setelah proses tersebut, dilakukan *text segmentation* terhadap hasil *text summarization* dengan tujuan untuk menyederhanakan kalimat percakapan. Penyederhanaan kalimat didasarkan pada pola kalimat minimal yaitu subyek-predikat yang memiliki kelas kata tertentu. Hasil yang diperoleh dari pemotongan kata berdasarkan pola tersebut hanya dapat diaplikasikan pada beberapa kalimat percakapan. Hal ini dikarenakan pemberian

label masih kurang tepat pada beberapa kata. Penyebabnya adalah limitasi model Bahasa Indonesia yang digunakan dalam proses *training* data terhadap kalimat percakapan. Salah satu contoh ketidaksesuaian dalam memberi label kelas kata terdapat pada kata yang menyatakan istilah singkatan atau jabatan yang seharusnya mendapat kelas kata *pronominal* (PRON). Namun hasil *training* data memberikan *label other* (X) yang menyatakan tidak memiliki definisi kelas kata. Hal ini juga berlaku untuk kata serapan seperti kata “survei” seharusnya mendapatkan *label nominal* tetapi *training* data memberikan *label other*. Contoh penguraian kata dalam suatu kalimat percakapan dapat dilihat pada Tabel 1.

Tabel 1. Contoh Pemotongan Berdasarkan Pola Kalimat

Kalimat “.. sudah melakukan survei ke UTC untuk kegiatan jurit malam.”		
Kata	Kelas Kata Seharusnya	PoS Tag UD Hasil Sistem
sudah	VERB	VERB
melakukan	VERB	VERB
survei	NOUN	X
ke	ADP	ADP
UTC	NOUN	X
untuk	ADP	ADP
kegiatan	NOUN	NOUN
jurit	NOUN	NOUN
malam	NOUN	NOUN
<b>T. Segm. Seharusnya</b>		melakukan survei ke UTC untuk kegiatan jurit malam
<b>T. Segm. Hasil Sistem</b>		kegiatan jurit malam

Pada tahap validasi terhadap hasil sistem, diminta bantuan 5 responden untuk memberikan pendapat dari hasil sistem. Tahap ini diawali dengan menyajikan teks narasi percakapan yang telah dikelompokkan per agenda rapat beserta dengan hasil pengambilan percakapan penting yang telah disederhanakan dari sistem. Pendapat dari responden akan digunakan untuk membandingkan seberapa banyak kalimat yang dapat mewakili pembahasan dari narasi, dibandingkan dengan seberapa banyak kalimat yang dihasilkan dari sistem yang dianggap penting oleh responden. Pada perbandingan hasil pengambilan percakapan penting rapat, diperoleh akurasi tertinggi sebesar 81,3% dan terendah sebesar 47,3% yang dapat dilihat pada Tabel 2.

Akurasi tertinggi dapat dicapai ketika pembahasan suatu agenda memiliki struktur yang baik. Contohnya penyampaian informasi tugas suatu divisi beserta tenggat waktu dari tugas tersebut. Selain itu, topik yang dibahas dalam suatu rapat tidak lebih dari 1 topik pembahasan. Sedangkan untuk nilai akurasi terendah disebabkan karena pemotongan kalimat pada proses *text segmentation* yang kurang tepat. Hal ini kemudian dapat menghilangkan arti pada beberapa struktur kalimat percakapan. Permasalahan lain terjadi pada pola kalimat dengan kelas kata tertentu yang hanya dapat diaplikasikan pada beberapa struktur kalimat percakapan.

Tabel 2. Perbandingan Hasil Pengambilan Percakapan Penting Rapat

Ag.*	HS*	Responden										Rt. (%)*
		Ketua		Notulis		Peserta		Dosen 1		Dosen 2		
		S**	T***	S	T	S	T	S	T	S	T	
<b>Rapat 1</b>												
Ag. 1	9	6	3	7	1	5	3	7	4	8	3	56,436
Ag. 2	4	4	1	4	1	4	1	3	1	4	1	76
Ag. 3	7	4	2	6	1	4	1	4	3	5	2	53
Ag. 4	6	3	2	4	0	3	0	2	4	5	2	47,333
<b>Rata-Rata Akurasi Rapat 1</b>											58,192	
<b>Rapat 2</b>												
Ag. 1	15	10	5	13	0	7	0	7	6	12	5	55,333
Ag. 2	9	8	0	8	1	8	0	7	2	8	2	78,828
Ag. 3	4	4	1	4	0	4	0	3	1	4	2	81,333
Ag. 4	11	10	4	11	5	10	2	10	8	9	6	63,583
Ag. 5	5	4	2	4	0	4	0	4	1	4	3	66,762
Ag. 6	8	6	2	5	0	6	1	6	3	7	3	66,470
<b>Rata-Rata Akurasi Rapat 2</b>											68,718	
<b>Rata-Rata Akurasi Keseluruhan</b>											64,508	

Keterangan:

\* Ag. = Agenda; HS = Hasil Sistem; Rt (%) = Rata-rata akurasi

\*\* jumlah kalimat yang sesuai

\*\*\* jumlah kalimat yang ditambahkan responden sebagai kalimat penting

Selain permasalahan *text segmentation* yang kurang tepat, permasalahan lainnya adalah perhitungan bobot setiap kalimat mempengaruhi hasil pengambilan percakapan penting dalam pembahasan rapat. Algoritma *TextRank* menerapkan perhitungan bobot setiap kalimat berdasarkan *overlapping content* atau jumlah kata yang sering muncul di setiap hubungan antar kalimat. Ketika pembahasan agenda rapat berupa diskusi, kecenderungan kata yang sering disebutkan dalam diskusi menjadi acuan untuk memberikan peringkat antar kalimat. Acuan tersebut terkadang kurang tepat untuk beberapa konteks pembahasan rapat. Salah satu contohnya terdapat pada hasil agenda 4 dari rapat 1 yang memiliki nilai akurasi sebesar 47%. Dalam agenda tersebut terdapat diskusi mengenai harga makanan untuk peserta suatu acara. Namun, ada pula pembahasan lain dalam agenda tersebut yaitu penekanan dalam membuat susunan acara. Oleh sebab itu, agenda tersebut memiliki struktur yang tidak baik karena membahas 2 topik dalam agenda tersebut. Selain itu, kecenderungan kata “susunan”, “acara”, dan “pembicara” sering disebutkan oleh peserta rapat. Hal ini mengakibatkan perhitungan kalimat yang mengandung kata tersebut mendapat peringkat tertinggi dalam sistem.

*Meeting Assistant System* menghasilkan 7 kalimat tertinggi sehingga kalimat dengan indeks 0, 1, 3, 4, 5, 7 dan 8 menjadi kalimat penting yang disimpan pada sistem ini. Bila dilihat dari Tabel 3, kecenderungan kata “batasan”, “harga”, “makanan”, dan “anggaran” menghasilkan bobot kalimat yang masuk dalam urutan tertinggi. Hal ini disebabkan, kata-kata tersebut sering diucapkan oleh peserta rapat. Akan tetapi, kalimat indeks 8 memperoleh bobot tertinggi dari urutan

kalimat yang dihasilkan oleh sistem. Hal ini disebabkan karena kalimat indeks 8 mengandung kata “acara” yang telah disebutkan sebanyak 3 kali dalam kalimat tersebut. Perhitungan *TextRank* kalimat menyebabkan kalimat indeks 8 memiliki nilai “voting” yang tinggi di antara 13 kalimat lainnya sehingga mempengaruhi perhitungan bobot kalimat. Nilai “voting” ini dipengaruhi oleh perhitungan bobot kalimat dengan menggunakan rumus *cosine similarity* yang dilakukan antara 2 kalimat. Hasil pemeringkatan kalimat percakapan dari agenda ini dapat dilihat pada Tabel 3.

Untuk verifikasi terhadap *speech recognition* yang ditangani oleh *Google Speech-to-Text API*, saat ini tanda baca tidak didukung dalam *speech recognition* Bahasa Indonesia. Padahal tanda baca menjadi acuan pemotongan keseluruhan teks untuk proses *text summarization*. Untuk mengatasi keterbatasan ini maka sistem menganggap 1 file percakapan sama dengan 1 kalimat. Selain itu, keterbatasan lain yang dimiliki oleh API ini selama verifikasi adalah kecepatan dan kejelasan peserta dalam mengucapkan kata mempengaruhi hasil penerjemahan suara. Dalam uji coba yang dilakukan, ada kesalahan penerjemahan suara pada beberapa kata yang diucapkan peserta. Kesalahan tersebut berakibat pada perbedaan arti antara hasil yang diberikan oleh sistem dengan hasil sebenarnya. Permasalahan ini dapat diatasi selama peserta rapat dapat melafalkan setiap kata percakapannya dengan jelas dan memiliki kecepatan berbicara yang cenderung stabil.

Tabel 3. Urutan Kalimat Berdasarkan Nilai TextRank

Index Kal.	Bobot Kal.	Kalimat
8	0,100694	Saya mengingatkan kembali, <i>Workshop Android</i> merupakan acara yang belum pernah diadakan sehingga divisi acara harus membuat susunan acara terlebih dahulu sebelum bertemu pembicara.
3	0,099652	Sebab, dari anggaran besar seharusnya batasan harga makanan adalah Rp 10.000
0	0,092014	Untuk masalah harga makanan, apakah kami diberi batasan harga atau kami yang menentukan harga tersebut?
7	0,085772	Mereka punya kontak <i>catering</i> dengan batasan harga makanan Rp 10.000 sampai Rp 12.500.
1	0,080702	Dari masalah batasan harga makanan, Saya menentukan Rp 13.000 sebagai batasannya.
4	0,079603	Ada masalah terkait harga Rp10.000 yaitu susah mencari makanan dengan kemasan <i>box</i> .
5	0,071418	Kalau begitu, berdasarkan anggaran besar dimana biaya pendaftaran peserta adalah Rp 30.000 dengan konsumsi seharga Rp 10.000, kalian harus menekan pembiayaan lain agar batasan harga makanan dapat ditingkatkan.
9	0,067401	Tetapi, susunan acara tersebut dapat diubah sesuai dengan hasil diskusi bersama pembicara.
13	0,056619	Untuk masalah batas waktu tugas setiap divisi akan dibagikan lewat grup <i>chat</i> divisi.
10	0,047192	Antisipasi ini dilakukan karena berdasarkan pengalaman di <i>Workshop AR</i> , pembicara membawakan materi dengan cepat sehingga sesi istirahat lebih cepat.
6	0,046908	Alternatif lain, kalian dapat mencoba bertanya ke anggota KSM yang pernah menjadi divisi konkes untuk referensi makanannya.
11	0,029169	Jadi, lebih baik dibahas hingga jelas terkait waktu selama pembawaan materi.
2	0,010714	Sebelumnya, apakah kamu sudah membuat rencana dananya?
12	0,010714	Ada pertanyaan lagi?

#### IV. KESIMPULAN

Melalui hasil uji coba dan evaluasi yang telah dilakukan pada sistem, maka dapat disimpulkan bahwa proses perekaman suara setiap peserta dalam rapat dapat ditangani oleh sistem dengan baik. Namun proses penerjemahan suara yang dilakukan oleh *Google Speech-to-Text* API memiliki batasan tertentu.

Pengambilan kalimat penting dapat mewakili beberapa konteks pembahasan rapat dengan nilai akurasi tertinggi sebesar 81,3% dan terendah sebesar 47,3%. Ada 3 faktor yang mempengaruhi nilai akurasi tersebut yaitu yang pertama struktur pembahasan agenda mempengaruhi hasil pengambilan kalimat percakapan yang penting. Hal ini dikarenakan perhitungan bobot kalimat didasarkan pada *overlapping content* atau kesamaan kata antarkalimat yang diucapkan peserta. Kedua, penyederhanaan kalimat berdasarkan pola kalimat dan kelas kata tertentu tidak dapat diaplikasikan pada beberapa kalimat percakapan, karena dapat mengubah arti dari kalimat tersebut. Ketiga, limitasi pada model *training data* Bahasa Indonesia mempengaruhi pemberian kelas kata. Dimana hal ini juga menjadi penyebab kesulitan lain dari penyederhanaan kalimat berdasarkan pola kalimat dan kelas kata tertentu pada sistem ini.

Adapun batasan dari penelitian ini adalah diperlukan penambahan tanda baca pada hasil terjemahan *speech-to-text* untuk mempermudah *preprocessing* data sebelum diproses pada algoritma *TextRank*. Selain itu penggunaan model *training data* Bahasa Indonesia dengan kelas kata yang lebih lengkap juga masih perlu dikembangkan. Terakhir adalah masih perlunya optimasi terhadap agenda rapat dengan konteks pembahasan yang strukturnya kurang baik.

#### REFERENSI

- [1] Green, W.A. & Lazarus, H. (1991). Are Today's Executives Meeting with Success? *Journal of Management Development*, Vol. 10(1), pp. 14–25.
- [2] Daft, R.L. & Lengel, R.H. (1984). Organizations as Information Processing Systems. *Research in Organizational Behavior*, Vol. 6, pp. 191–233.
- [3] Whittaker, S., Laban, R. & Tucker, S. (2006). Analysing Meeting Records: an Ethnographic Study and Technological Implications. *Lecturer Notes in Computer Science (including Subseries Lecturer Notes in Artificial Intelligence and Lecturer Notes in Bioinformatics)*, vol. 3869 LNCS, July 2005, pp. 101–113.
- [4] Nathan, M. (2012). In Case You Missed It: Benefits of Attendee-Shared Annotations for Non-Attendees of Remote Meetings. *Proceeding of ACM Conference on Computer Supported Cooperative Work (CSCW)*, pp. 339–348.
- [5] Whittaker, S., Kalnikaité, V. & Ehlen, P. (2012). Markup as You Talk: Establishing Effective Memory Cues While Still Contributing to a Meeting. *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW)*, p. 349.
- [6] Tucker, S., Bergman, O., Ramamoorthy, A. &

- 
- Whittaker, S. (2010). Catchup: a Useful Application of Time-travel in Meetings. *Proceeding of the ACM Conference on Computer Supported Cooperative Work (CSCW)*, pp. 99–102.
- [7] Khan, M. (2020). *Reader: Speech Synthesizer and Speech Recognizer*, Vol. 1166.
- [8] Tur, G. (2010). The CALO Meeting Assistant System. *IEEE Transactions on Audio Speech and Language Processing*, Vol. 18(6), pp. 1601–1611.
- [9] Thompson, P., James, A. & Nanos, A. (2013). V-ROOM: Virtual Meeting System Trial. *Proceeding of 2013 IEEE 17th International Conference on Computer Supported Cooperative Work in Design (CSCWD) 2013*, pp. 563–569.
- [10] Pragantha, J. (2017). Penerapan Algoritma TextRank Untuk Automatic Summarization Pada Dokumen Berbahasa Indonesia. *Jurnal Ilmu Teknik dan Komputer*, Vol. 1(1), pp. 71–78.
- [11] Shakhovska, N., Medykovskyj, M. & Bychkovska, L. (2015). Building a Smart News Annotation System for Further Evaluation of News Validity and Reliability of Their Sources. *Przegląd Elektrotechniczny*, Vol. 91(7), pp. 43–44.
- [12] Brin, S. & Page, L. (2012). Reprint of: The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Computer Networks*, Vol. 56(18), pp. 3825–3833.
- [13] Mihalcea, R. & Tarau, P. (2004). TextRank: Bringing Order into Texts. *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, Vol. 45(4).
- [14] Mihalcea, R. (2004). Graph-based Ranking Algorithms for Sentence Extraction, Applied to Text Summarization. *Proceedings of the ACL 2004 on Interactive Poster and Demonstration Sessions*, No. 4, pp. 20.
- [15] Lavanya, A.P. & Rajeswari, K.C. (2020). Aspect Based Extractive Summarization of Online Product Reviews. *International Journal of Scientific & Technology Research*, Vol. 9(3), pp. 3588–3591.
- [16] Garg, N., Favre, B., Reidhammer, K. & Hakkani-Tür, D. (2009). ClusterRank: a Graph Based Method for Meeting Summarization. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pp. 1499–1502.