

**PEMBANGKITAN PARAFRASE UNTUK KALIMAT SITASI  
MAKALAH ILMIAH DENGAN ALGORITMA  
*SIMULATED ANNEALING***

**RINGKASAN DISERTASI**

**Oleh**

**RIDWAN ILYAS**

**NIM: 33216307**

**(Program Studi Doktor Teknik Elektro dan Informatika)**



**INSTITUT TEKNOLOGI BANDUNG**

**JULI 2023**

# Pembangkitan Parafrase Untuk Kalimat Sitasi Makalah Ilmiah

## Dengan Algoritma *Simulated Annealing*

Ridwan Ilyas  
NIM: 33216307

### I. Latar Belakang

Dalam penulisan ilmiah, penggunaan kalimat sitasi sangatlah penting untuk memperkuat argumen yang dibangun oleh penulis. Kalimat Sitasi adalah kalimat yang digunakan untuk merujuk atau mengutip pernyataan atau hasil penelitian dari sumber tertentu, yang disajikan dalam tulisan ilmiah, dan disertai dengan penulisan nama penulis, tahun terbitan, dan nomor halaman (Teufel, 1999). Dalam dunia akademis, penggunaan kutipan memiliki banyak tujuan (Teufel dkk., 2006). Salah satu tujuannya adalah untuk mengungkapkan kelemahan dari suatu teori atau metodologi tertentu. Kutipan juga dapat digunakan untuk membandingkan perbedaan antara satu konsep dengan konsep lainnya. Selain itu, kutipan juga dapat menunjukkan perbedaan dalam pendekatan atau data yang digunakan dalam penelitian. Tidak hanya itu, kutipan juga bisa menunjukkan kemiripan dalam pendekatan atau data yang digunakan dalam penelitian. Kutipan juga dapat digunakan sebagai basis untuk menunjukkan sumber masalah atau tantangan dalam penelitian. Terakhir, kutipan dapat digunakan sebagai kutipan netral yang menunjukkan fakta atau informasi umum yang relevan dengan topik yang dibahas.

Dalam penulisan akademik, parafrase pada kalimat sitasi perlu dibuat karena beberapa alasan (Hood, 2010). Pertama, dengan melakukan parafrase pada kalimat sitasi, penulis dapat menunjukkan pemahaman yang lebih baik terhadap sumber yang digunakan. Kedua, parafrase pada kalimat sitasi juga dapat membantu membuat tulisan menjadi lebih bervariasi dan menarik bagi pembaca. Ketiga, parafrase pada kalimat sitasi dapat membantu meningkatkan kualitas tulisan secara keseluruhan. Dengan melakukan parafrase pada kalimat sitasi, penulis dapat memperjelas konsep atau ide yang disampaikan oleh sumber dan membantu pembaca untuk lebih memahami tulisan.

Pembangkitan parafrase merupakan sebuah *task* yang bertujuan untuk menghasilkan teks baru dari teks asli dengan menggunakan bahasa yang sama tetapi memiliki variasi leksikal (Androutopoulos dan Malakasiotis, 2010).

Dalam melakukan *task* tersebut, metode pembangkit parafrase menghasilkan kalimat baru yang memiliki perbedaan kata-kata tetapi tetap mempertahankan kesamaan makna dengan kalimat asli. Metode pembangkit parafrase sering kali dibandingkan dengan mesin penerjemah, yang membedakan keduanya hanya pada masukan dan luaran yang dihasilkan, di mana mesin parafrase menghasilkan kalimat dalam bahasa yang sama seperti teks asli (Quirk dkk., 2003).

Penelitian pembangkitan parafrase telah banyak dilakukan dengan berbagai pendekatan dalam bidang Kecerdasan buatan. Pendekatan dalam bidang ini dapat dikelompokkan ke dalam tiga jenis yaitu *Knowledge-Based*, *Learning* dan *Problem Solving Agent* (Russell dkk., 2010). Masing-masing pendekatan memiliki kekurangan dan kelebihan masing-masing, digunakan dalam konteks dokumen yang berbeda-beda dan dapat dipakai sebagai landasan untuk penelitian pembangkitan parafrase dalam konteks kalimat situs.

Pendekatan dengan *Knowledge-Based* untuk pembangkitan parafrase dilakukan berbasis aturan yang dibuat secara manual (McKeown, 1983). Karena pekerjaan membuat daftar aturan perubahan parafrase yang sangat banyak, beberapa peneliti mencoba mengumpulkan aturan parafrase secara otomatis (Lin dan Pantel, 2001). Namun, terdapat keterbatasan metode ekstraksi aturan sehingga menghasilkan pola aturan yang panjang. Dari hasil pengujian didapati kinerja mesin yang belum optimal (Zhou dan Bhat, 2021). Selain menggunakan daftar aturan, penggunaan tesaurus dipakai untuk mengekstrak semua sinonim dari kata-kata yang diganti. Pilihan terbaik kemudian dipilih sesuai dengan konteks frasa kalimat input (Kauchak dan Barzilay, 2006). Meskipun sederhana dan efektif, keragaman parafrase yang dihasilkan cenderung terbatas.

Pendekatan *Learning* untuk pembangkitan parafrase dilakukan dengan model *Seq2seq* karena perkembangan metode jaringan syaraf tiruan khususnya arsitektur LSTM (Prakash dkk., 2016). Jaringan syaraf konvulsional (CNN) yang lebih umum digunakan untuk memproses citra, dapat digunakan untuk membangun model *seq2seq* karena memiliki parameter yang lebih sedikit dan lebih cepat dalam pelatihan (Vizcarra dan Ochoa-Luna, 2020). Model *Transformer* (Vaswani dkk., 2017) telah menunjukkan kemampuan untuk meningkatkan kinerja pada berbagai *task* pembangkitan teks. Model *Transformer* dikembangkan untuk pembangkitan parafrase karena kemampuan menangkap pola kalimat yang cukup panjang (Wang dkk., 2018). Metode pembangkitan parafrase dengan arsitektur jaringan syaraf tiruan menunjukkan

hasil yang lebih baik, namun masih memiliki keterbatasan karena memerlukan data latih yang besar.

Pendekatan *Problem Solving Agent* untuk pembangkitan parafrase dilakukan dengan menggunakan strategi *searching*. Salah satu algoritma yang dipakai adalah *Simulated annealing* (Liu dkk., 2019). Penelitian tersebut mengembangkan sebuah metode untuk membangkitkan kalimat dengan arsitektur *generate and test model*. Penelitian ini menunjukkan kemampuan untuk menghasilkan kalimat parafrase dengan kualitas yang baik tanpa memerlukan paralel korpus sebagai data pelatihan. Dalam penelitian tersebut, proses pembangkitan dilakukan dengan cara melakukan perubahan kalimat menggunakan teknik substitusi, penambahan, dan penghapusan kata. Penelitian tersebut menggunakan daftar kata sebagai sumber daya bahasa untuk melakukan operasi penggantian atau penambahan. Dalam menentukan kandidat terbaik digunakan model bahasa N-gram. Pendekatan dengan *Problem Solving Agent* memiliki keunggulan tidak memerlukan dataset yang besar sebagai data latih. Dari beberapa penelitian yang menggunakan dataset publik seperti quora dan twitter, menunjukkan hasil yang baik.

Algoritma *Simulated annealing* adalah sebuah algoritma optimasi yang terinspirasi dari proses pendinginan logam, dimana suatu benda dipanaskan dan kemudian dibiarkan mendingin secara perlahan sehingga molekulnya dapat menyusun diri dengan rapi (Russell dkk., 2010). *Simulated annealing* merupakan salah satu metode optimasi yang menggunakan prinsip *local search*. *Local search* adalah pendekatan mencari solusi hanya dengan mengeksplorasi sebagian kecil dari ruang keadaan, dimulai dari suatu solusi acak atau solusi yang sudah ada, dan mencoba meningkatkan solusi tersebut dengan menggerakkan solusi ke *state* yang lebih baik dalam ruang keadaan. Dalam algoritma *Simulated annealing*, *state* hasil pencarian dievaluasi menggunakan suatu fungsi objektif. Jika *state* baru yang ditemukan nilainya lebih baik daripada *state* sebelumnya, maka solusi baru tersebut diterima. Namun, jika *state* baru tersebut nilainya lebih rendah daripada *state* sebelumnya, maka solusi baru tersebut tetap dapat diterima dengan probabilitas tertentu. Probabilitas ini menurun seiring dengan berjalannya waktu, sehingga semakin lama algoritma ini berjalan, maka semakin kecil probabilitas untuk menerima *state* yang lebih rendah.

Penelitian ini memilih fokus pada *task* pembangkitan parafrase pada makalah ilmiah karena *task* ini secara langsung membutuhkan proses pengerjaan *task*

lainnya. Pembangkitan parafrase membutuhkan adanya korpus baik itu sebagai data latih, atau data uji model yang dikembangkan. Proses menghasilkan korpus parafrase harus melalui kegiatan penelitian *task* ekstraksi parafrase. Pembangkitan parafrase dengan pendekatan algoritma *simulated annealing* memerlukan adanya fungsi objektif untuk mengawal proses pembangkitan. Fungsi objektif dibangun melalui kegiatan penelitian *tasks* klasifikasi parafrase. Dengan memilih topik pembangkitan parafrase kalimat sitasi, maka penelitian ini dapat dianggap mengerjakan semua *task* lainnya.

## I.1 Rumsan Masalah

Pembangkitan parafrase telah banyak dikembangkan dengan berbagai pendekatan seperti *rule-based* (Lin dan Pantel, 2001; Zhou dan Bhat, 2021), *machine learning* (Prakash dkk., 2016; Vizcarra dan Ochoa-Luna, 2020; Zhao dkk., 2009) dan *problem solving agent* (Li dkk., 2020; Liu dkk., 2019). Pendekatan dengan *rule-based* menunjukkan hasil yang terbatas dan kesulitan membangun daftar aturan, sedangkan dengan pendekatan *learning* membutuhkan paralel korpus parafrase yang cukup besar. Pembangkitan parafrase telah diterapkan pada bebas domain (Brad, 2017; Pavlick dkk., 2015), domain berita (Quirk dkk., 2003; Shinyama dkk., 2002) dan domain sosial media *twitter* (Xu dkk., 2015) atau *quora* (Miao dkk., 2019). Namun, metode pembangkitan parafrase yang sudah ada tidak dapat langsung diadaptasi untuk menghasilkan kalimat baru pada domain makalah ilmiah karena keterbatasan sumberdaya bahasa, kalimat sitasi memiliki maksud penyampaian argumentatif, memiliki batasan minum kesamaan leksikal, bentuk kalimat kompleks dan pilihan leksikal yang terbatas. Penggunaan pendekatan *Problem Solving Agent* menjadi potensi untuk dikembangkan, namun komponen metode yang ada seperti fungsi objektif, operasi perubahan *state* dan strategi pemilihan *state* tidak dapat langsung diterapkan dari metode yang sudah ada.

Dari rumusan masalah tersebut, dibentuk pertanyaan penelitian: “Bagaimana cara mengembangkan metode pembangkit parafrase untuk kalimat sitasi dengan pendekatan *problem solving agent* menggunakan algoritma *simulated annealing*, dengan mempertimbangkan perubahan sumber daya bahasa, pengembangan fungsi objektif, teknik perubahan kalimat, dan strategi pemilihan *state* yang optimal?”

## **I.2 Tujuan dan Sasaran Penelitian**

Tujuan dari penelitian ini adalah menghasilkan metode pembangkitan kalimat parafrase pada domain kalimat sitasi dengan algoritma *simulated annealing*. Metode pembangkitan didukung dengan adanya korpus sebagai data acuan untuk evaluasi pembangkitan. Model dihasilkan dengan cara mengembangkan arsitektur pembangkitan teks berbasis perubahan leksikal. Selain metode pembangkitan, penelitian ini bertujuan untuk membangun korpus paralel parafrase kalimat sitasi sebagai acuan pembuatan fungsi objektif dan mengevaluasi hasil pembangkitan.

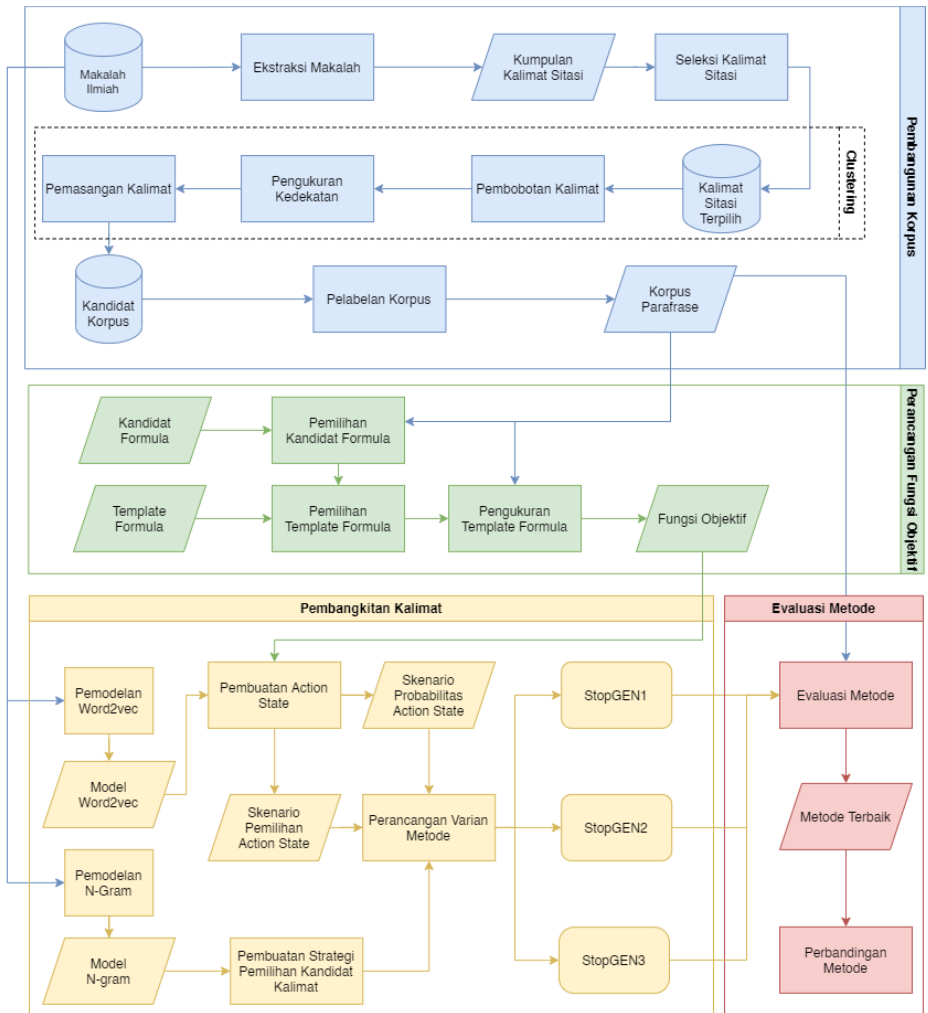
## II. Metode Penelitian

Metode penelitian pembangkitan parafrase untuk kalimat sitasi memiliki beberapa bagian. Bagian pertama adalah pembangunan korpus paralel kalimat sitasi dari kumpulan makalah ilmiah. Bagian kedua adalah pembuatan fungsi objektif yang cocok untuk mengukur nilai derajat parafrase antar dua kalimat. Bagian terakhir adalah membangun metode pembangkit kalimat parafrase berdasarkan algoritma *simulated annealing*.

Bagian penelitian awal adalah pembangunan korpus. Proses ini dimulai dengan mengekstraksi makalah sehingga menghasilkan kumpulan kalimat sitasi. Tahap selanjutnya adalah melakukan clustering kalimat untuk mendapatkan kandidat korpus. Terakhir adalah memasang kalimat dan memberi label parafrase oleh anotator.

Kegiatan penelitian selanjutnya adalah perancangan fungsi objektif. Pada penelitian ini dimulai dengan memilih matriks yang paling tepat untuk memisahkan kalimat parafrase dan bukan parafrase. Selanjutnya memilih template fungsi objektif yang memasang matriks terpilih dalam satu formula.

Kegiatan utama penelitian adalah merancang metode pembangkitan kalimat. Setelah dihasilkan fungsi objektif dari proses selanjutnya, maka dibangun beberapa alternatif metode. Alternatif yang ada dibedakan berdasarkan strategi skenario perubahan kalimat. Untuk mendapatkan variasi terbaik, maka dilakukan evaluasi berdasarkan korpus pasangan parafrase dari proses sebelumnya. Evaluasi juga dilakukan dengan membandingkan hasil pembangkitan metode lain.



Gambar II-1 Metode Penelitian



### I.3 Rancangan Stochastic Paraphrase Generator (StoPGEN)

Penelitian ini mengusulkan StoPGEN sebagai model untuk membangkitkan kalimat parafrase dengan basis Algoritma *Simulated annealing*. StoPGEN dikembangkan dengan 3 jenis arsitektur. Arsitektur pertama menggunakan skema pemilihan kata target secara acak seperti yang dilakukan oleh UPSA. Arsitektur kedua menggunakan skema pemilihan secara berurutan dari awal kalimat hingga akhir. Arsitektur ketiga menggunakan skema pemilihan secara berurutan dengan tambahan penggunaan model bahasa probabilistik untuk memilih *successor*.

#### ***Simulated annealing* untuk Pembangkitan Kalimat**

**FUNCTION** StopGEN( $x[0\dots n]$ : Array of token) **return**  $S[0\dots n]$ : Array of token  
{function of sentence generator using *simulated annealing*}

#### **DECLARATION**

T, initial temperature  
 $\Delta T$ , temperature drop  
s, current state {state is array of token}  
Pscore, objective function  
pLM(), language model probability function {return score}  
most\_similar(), get most similar token from word index  
neighbors(), get most similar token by neighbors from word index

#### **ALGORITHM**

```
T ← T0
Sk ← x
while T > 0 and n do
  action ← get_random_action()
  if action = substitution do
    sk+1[n] = max(pLM(most_similar(sk+1[n])))
  else if action = insertion do
    sk+1[n] = max(pLM(neighbors([sk+1[n-2], sk+1[n-1], sk+1[n+1], sk+1[n+2])))
  else
    remove(sk+1[n])
    ΔE ← Pscore(sk+1) - Pscore(sk)
    if min(1, e-ΔE/T) >= rand(0,1) then
      Sk ← sk+1
    end if
    T ← T - ΔT
  end while
return sk {new array of token}
```

Gambar II-2 Algoritma *Simulated annealing* untuk Pembangkitan Kalimat

StoPGEN dikembangkan dari algoritma *simulated annealing* sehingga dimasukkan dalam kategori kecerdasan buatan dengan pendekatan *problem solving* (*Searching*). Jika pendekatan ini dilihat dalam aspek pembelajaran mesin (*machine learning*) maka masuk ke dalam kategori *unsupervised*. Dalam komponen stopgen terdapat pemodelan sumber daya bahasa yaitu word2vec dan n-gram. Kedua pemodelan tersebut masuk dalam kategori *unsupervised*.

StopGEN merupakan algoritma yang berfungsi sebagai generator kalimat menggunakan teknik *simulated annealing*. Dalam algoritma ini, input yang diberikan adalah sebuah array token, dan algoritma menghasilkan kalimat baru dengan melakukan beberapa modifikasi pada array tersebut.

### **I.3.1 StoPGEN Variasi 1**

Model utama StoPGEN 1 dikembangkan berdasarkan baseline penelitian UPSA dengan mengembangkan sumber daya bahasa dan fungsi objektif. Model menerima input susunan kalimat lalu memilih target token secara acak. Setelah itu sistem memilih aksi secara acak terhadap target token yang sudah terpilih. Jika aksi terpilih hapus, maka token terpilih langsung dihapus dan menghasilkan *New State* dan diperlakukan sebagaimana algoritma *simulated annealing*. Jika aksi terpilih adalah substitution atau insertion, maka dipilih token baru dari sumber daya bahasa berbasis *Word Embedding*. Proses pembangkitan kalimat selesai berdasarkan mekanisme pemberhentian suhu.

### **I.3.2 StoPGEN Variasi 2**

Model StoPGEN variasi 2 dikembangkan dari model StoPGEN sebelumnya dengan mengganti mekanisme pemilihan target token. Target token tidak dipilih secara acak melainkan berdasarkan urutan pada kalimat masuk. Perubahan pada algoritma *simulated annealing* terjadi pada proses penurunan suhu. Suhu pada algoritma diturunkan agar jumlah aksi sama dengan sejumlah target token.

### **I.3.3 StoPGEN Variasi 3**

Model StoPGEN variasi 3 dikembangkan dari variasi StoPGEN 2 dengan menambahkan mekanisme evaluasi kandidat token pengganti atau penambah pada aksi substitution atau insertion. Hasil pemilihan token dari word embedding memungkinkan menghasilkan lebih dari bentuk kalimat. Pada model sebelumnya token dipilih berdasarkan kedekatan tertingggi vektor token.

Pada model StoPGEN ini kandidat kalimat dipilih berdasarkan peluang kemunculan setiap segmen 3-gram.

### **III. Hasil dan Pembahasan**

Pada bagian ini akan dijelaskan hasil menjalankan eksperimen. Pertama dijelaskan mengenai kegiatan pembangunan korpus. Selanjutnya dijelaskan kegiatan pembuatan fungsi objektif. Terakhir dijelaskan mengenai evaluasi metode pembangkit parafrase

#### **III.1 Pembangunan Korpus**

Makalah ilmiah yang telah dikumpulkan di ekstrak menggunakan alat bantu Dr. Inventor Framework (Donoghue dkk 2014). Hasil dari ekstraksi makalah disimpan dalam basis data sehingga dapat dikelola untuk hasil selanjutnya.

Kandidat korpus dari hasil tahapan penelitian sebelumnya telah diberi label parafrase. Jumlah pasangan kalimat yang telah diberi label sebanyak 4675 dengan jumlah kalimat parafrase 2386 dan jumlah kalimat yang tidak parafrase 2289. Dataset pada penelitian ini dibangun dengan melakukan survey kepada 5 orang narasumber untuk menentukan dua pasang kalimat parafrase atau bukan. Bentuk survey berupa nilai biner ya atau tidak. Untuk melihat tingkat persetujuan pada dataset perhitungan Fleiss' Kappa.

Dari hasil perhitungan nilai kappa untuk dataset parafrase kalimat sitasi yang telah dilabeli, menghasilkan nilai 0.26. Dengan nilai tersebut, maka tingkat signifikansi atau persetujuan pada level *Fair Agreement* atau kesepakatan yang sangat baik. Oleh karena itu, dataset yang dipakai dianggap cukup layak untuk digunakan.

#### **III.2 Pembuatan Fungsi Objektif**

Eksperimen pembuatan fungsi objektif dilakukan untuk mendapatkan formula yang dapat menghitung derajat parafrase dua buah kalimat. Proses ini dimulai dengan mengevaluasi kandidat komponen penghitung. Selanjutnya dengan menguji komponen perhitungan dengan cara memasukannya ke dalam template fungsi objektif.

### III.2.1 Penggunaan Meteor

Kedua pararel korpus diukur menggunakan METEOR. Setelah diukur setiap pasangan kalimat, lalu dihitung nilai KLD dan dibuat tabel sebaran distribusi dengan selang 0,02. Dengan menggambarkan tabel distribusi dalam bentuk grafik, terlihat kecenderungan sebaran datanya dan perbedaan antara kedua kelompok korpus. Dari hasil pengukuran distribusi METEOR dengan KLD, didapatkan nilai **0,464537968**.

### III.2.2 Penggunaan Meta-discourse

Penggunaan Meta-discourse dimaksudkan untuk meningkatkan kemampuan memisahkan kalimat parafrase dan bukan parafrase. Dalam penggunaan meta-discourse, perlu dilakukan eksperimen tertentu untuk menentukan fitur mana saja yang paling optimal untuk dipakai. Jika melakukan pengujian seluruh kombinasi yang mungkin dari 1 sd 107 fitur yang ada, memerlukan komputasi yang sangat tinggi.

### III.2.3 Penggunaan Pinc Score

Penggunaan PINC Score untuk masuk sebagai parameter pada formula pengukuran parafrase, diukur sebagaimana penggunaan Meta-discourse. Dari hasil pengukuran, nilai KLD PINC Score adalah **0,24695152**. Jika merujuk pada definisi, maka idealnya PINC Score untuk pasangan kalimat parafrase tinggi, sehingga kurvanya berada disebelah kanan. Dari penggambaran distribusi kemunculan PINC Score terlihat bahwa kurva pasangan kalimat **bukan** parafrase justru lebih kanan dibanding dengan parafrase. Hal ini dapat dipahami dengan mudah karena kebanyakan kalimat bukan parafrase memiliki kesamaan leksikal yang rendah.

### III.2.4 Tempate Fungsi Dengan Weighted Linear

METEOR dan probabilitas kemunculan Meta-discours dipakai dalam formula Weighted Liner pertama yang fungsinya untuk menentukan nilai parafrase. Penggabungan ini bertujuan untuk memperkuat kemampuan METEOR dalam memberi kelas parafrasa pada pasangan kalimat. Diperlukan adanya nilai alpha sebagai konstanta pengali yang dapat memperkuat salah satu parameter dan melemahkan parameter lainnya guna membagi peran setiap parameter. Nilai alpha dari 0,1 sd 0,2 dengan melihat nilai KLD . Formula yang dipakai dalam penggabungan ini adalah:

$$PScore_1 = \alpha * PF + (1 - \alpha) MT$$

dimana:

$\alpha$  = konstanta pengali

$MT$  = Nilai METEOR

$PF$  = Probability Feature

Selanjutnya dianalisis penggunaan Weighted Formula dipakai PINC Score dan Meteor. Bentuk formula pengembangan dari  $PScore_1$ . Meteor dipakai sebagai formula untuk menghitung kesamaan semantik, sedangkan PINC Score berperan sebagai pengukur perbedaan leksikal.

$$PScore_2 = \alpha * MT + (1 - \alpha) Pinc$$

dimana:

$a$  = konstanta pengali

$Pinc$  = PINC Score

$MT$  = Meteor Score

Selanjutnya dianalisis penggunaan Weighted Formula dipakai PINC Score dan Probability Feature. Bentuk formula masih pengembangan dari  $PScore_1$ . Probability Feature dipakai sebagai formula untuk menghitung kesamaan semantik, sedangkan PINC Score berperan sebagai pengukur perbedaan leksikal.

$$PScore_3 = \alpha * PF + (1 - \alpha) Pinc$$

dimana:

$a$  = konstanta pengali

$Pinc$  = PINC Score

$PF$  = Probability Feature

Dari hasil menguji komponen-komponen yang ada pada semua kandidat fungsi obektif, ditemukan hasil terbaik pada  $PScore_2$  yang disusun antar komponen pengukuran Meteor dan Pinc Score. Formula ini dibangaun dengan nilai  $\alpha$  0,9 untuk Meteor yang berarti faktor penilaian Meteor sangat dominan di bandung dengan Pinc Score.

$$PScore_2 = 0,9 * MT + (1 - 0,9) Pinc$$

### III.3 Pembangkitan Parafrase

Eksperimen pembangkitan teks dibagi menjadi dua pendekatan. Pendekatan pertama adalah kuantitatif dengan tujuan melihat kemampuan model dalam menghasilkan kalimat baru yang dibandingkan dengan kalimat target. Pendekatan kedua adalah kualitatif dengan tujuan melihat tingkat penerimaan keterbacaan kalimat hasil pembangkitan.

#### III.3.1 Evaluasi Kuantitatif

Penelitian ini melakukan pengujian kemampuan model StopGEN dengan beberapa varian arsitektur dengan semua skenario *action factor*. Selanjutnya StopGEN dibandingkan performansinya pada data set public dan dataset kalimat sitasi dari makalah ilmiah. Penggunaan model yang dibandingkan, dikelompokkan dalam dua pendekatan yaitu *Supervised* dan *Unsupervised*. Penelitian ini menggunakan matrik evaluasi BLEU, Rouge 1, Rouge 2 dan Rouge L.

##### a. Data Set Standar

Model yang kami kembangkan kami coba lebih awal dengan dataset standar untuk melihat visibilitasnya. StopGEN yang dikembangkan dapat mengungguli model yang baik dengan pendekatan *Supervised* maupun *Unsupervised*. StopGEN mengungguli CGHM sebagai model *Supervised* terbaik secara signifikan pada daset Quora. Sedangkan pada dataset twitter hanya mengungguli beberapa poin saja. Model *Unsupervised* yang menjadi dasar pengembangan model ini yaitu UPSA, dapat diungguli di kedua dataset dengan angka yang tidak terlalu jauh. Hal ini karena UPSA dan StopGEN memiliki keunggulan adaptif dalam domain apa pun.

Pada di atas dapat dilihat bahwa StoPGEN dapat mengungguli metode lain dalam dua korpus yang berbeda. StoPGEN menghasilkan nilai BLEU 6.26, Rouge 1 28.60 dan Rouge 2 8.75 pada data set twitter. StoPGEN menghasilkan nilai BLEU 22.37, Rouge 1 61.09 dan Rouge 2 40.79 pada data set Quora.

Tabel III-1 Hasil Evaluasi pada Data Set Standar

Model	Twitter			Quora		
	BLEU	Rouge 1	Rouge 2	BLEU	Rouge 1	Rouge 2
<i>Supervised</i>						
VAE	3.46	15.13	3.40	13.96	44.55	22.64
LagVAE	3.74	17.20	3.79	15.52	49.20	26.07
CGHM	5.32	19.96	5.44	15.73	48.73	26.12
<i>Unsupervised</i>						
UPSA	5.30	19.96	5.44	18.21	59.51	32.63
StoPGEN	<b>6.26</b>	<b>28.60</b>	<b>8.75</b>	<b>22.37</b>	<b>61.09</b>	<b>40.79</b>

## b. Data Set Kalimat Sitasi

Kami telah menjelaskan sebelumnya bahwa StopGEN dikembangkan dengan tiga buah variasi yang masing-masing berbeda dalam membangkitkan kalimat. Hasil kualitatif menunjukkan bahwa membaginya secara merata menghasilkan performansi yang lebih baik.

Kami menguji Dataset Citation menggunakan model penelitian kami dan membandingkannya dengan model lain. Model *Supervised* yang dipilih diambil berdasarkan penggunaannya yang sudah secara umum digunakan pada domain mesin penerjemah. Model *Unsupervised* yang dipilih adalah baseline utama penelitian ini yaitu UPSA dan model UPSA yang telah dimodifikasi. Kami memodifikasi model UPSA dengan mengganti sumber bahasa yang dipakai dengan domain makalah ilmiah. UPSA menempatkan model bahasa untuk digunakan sebagai successor kata dan fungsi objektifnya.

Model *Supervised* dilatih dengan pasangan kalimat parafrase sitasi yang telah dilabeli True. Semua model menggunakan pendekatan sequence to sequence. Dari hasil eksperimen, model *Supervised* yang terbaik adalah dengan

menggunakan Bidirectional LSTM (Ma dan Hovy, 2016) mengungguli model mesin penerjemah lainnya dengan nilai BLEU 28.93.

Tabel III-2 Hasil Perbandingan Model Pada Kalimat Sitasi

<i>Unsupervised Model</i>				
	<b>BLEU</b>	<b>rouge1</b>	<b>rouge2</b>	<b>rougeL</b>
StoPGEN <sub>3</sub>	55.37	71.28	47.46	66.32
UPSA	21.20	45.93	15.43	41.55
Modified UPSA	33.81	51.25	26.67	45.94
<i>Supervised Model</i>				
LSTM encoder-decoder	25.77	22.60	7.68	20.13
bidirectional LSTM	28.93	26.10	11.75	23.44
Transformer	18.91	20.70	7.83	18.46

Model UPSA original yang kami gunakan langsung tanpa merubah dari sumbernya, hasilnya lebih buruk dibandingkan dengan UPSA yang telah dimodifikasi dengan mengganti sumber bahasanya dengan korpus makalah ilmiah. Meskipun demikian, StopPGEN masih bisa mengungguli model UPSA yang telah dimodifikasi. StopGEN mengungguli semua model dengan menghasilkan BLEU 55.37, Rouge 1 71.28, Rouge 2 47.46 dan RougeL 66.32.

### III.3.2 Evaluasi Kualitatif

Evaluasi kualitatif digunakan untuk melihat kemampuan hasil pembangkitan StopGEN dalam konteks bahasa yang dihasilkan. Dilakukan survey kepada pembaca sebanyak 30 orang dengan spesifikasi bidang computational linguistics untuk melihat penerimaan. Penelitian ini membandingkan penerimaan hasil pembangkitan seluruh variasi StopGEN dan perbandingannya dengan model lain. Kami juga mencoba melihat hasil setiap model lebih detail dengan memperlihatkan contohnya langsung

#### a. Penerimaan Kalimat

Dilakukan survey dengan memperlihatkan hasil pembangkitan dan pembaca diminta memilih status penerimaan pada setiap kalimat. Pada survey ini pembaca dibantu dengan adanya bantuan pewarnaan sebagai tandari dari aksi



yang terpilih. Dari hasil survey pada variant StoPGEN, model ke 3 memiliki nilai tertinggi dengan nilai penerimaan rata-rata adalah setengah dari seluruh jawaban survey. Hal ini semakin mendukung hasil secara kuantitatif bawah variasi model ini yang terbaik. Sedangkan dua variasi model lainnya menghasilkan nilai yang cukup jauh

Tabel III-3 Nilai Penerimaan Semua Variasi Model StopGEN

<b>Metode</b>	<b>Penerimaan</b>
StoPGEN <sub>1</sub>	26.45
StoPGEN <sub>2</sub>	27.09
StoPGEN <sub>3</sub>	50.96

Pada survey kedua (lihat Lampiran D), kami menggunakan variasi kalimat yang berbeda dan tidak menyertakan label aksi pada unit kalimat (lihat Tabel IV-21). Hasilnya StopGEN<sub>3</sub> memiliki nilai penerimaan yang mengungguli model baseline UPSA dan UPSA yang telah dimodifikasi untuk domain makalah ilmiah (lihat Tabel IV-19). UPSA model dasar memiliki nilai penerimaan rendah karena kami tidak memodifikasi sumber bahasa yang digunakan. Setelah kami mengganti model bahasa yang dipakai, nilai penerimaannya meningkat cukup tinggi menunjukkan luaran yang lebih baik.

Tabel III-4 Nilai Penerimaan Model Perbandingan

<b>Metode</b>	<b>Penerimaan (tanpa label aksi)</b>
UPSA	16.80
Modified UPSA	26.40
StoPGEN <sub>3</sub>	50.80

StopGEN<sub>3</sub> menghasilkan nilai penerimaan yang paling tinggi meski hasil yang ditampilkan berbeda dengan survey sebelumnya. Angka penerimaan 50.80 menunjukkan lebih dari setengah hasil yang ditampilkan mendapatkan penerimaan dari pembaca. Nilai yang lebih sedikit dari survey pertama bisa dianggap tidak signifikan.

## **b. Hasil Stokastik Model StoPGEN**

Model StoPGEN yang kami kembangkan memiliki kemampuan untuk menghasilkan kalimat parfrase secara stokastik pada setiap waktu

pembangkitan. Hasil pembangkitan kalimatnya dapat dikonfigurasi sesuai kebutuhan kecenderungan antara perbedaan leksikal atau kesamaan semantic. Hasil ini dapat ditunjukkan pada Tabel III-5 hasil pembangkitan kalimat dengan 10 kali percobaan.

Tabel III-5 Contoh Hasil Stokastik Model StoPGEN

<b>Input</b>	We use pre-trained glove (pennington et. Al., 2014) embeddings for our purposes
<b>Target</b>	we use glove (pennington et. al., 2014) for our word embeddings
<b>Output 1</b>	used trained glove (pennington et. al., 2014) embeddings for implement through
<b>Output 2</b>	use pre 300 dimensional glove embeddings (pennington et. al., 2014) word glove embeddings for our purposes
<b>Output 3</b>	we use glove trained pretrained embedding (pennington et. al., 2014) matrix for our through
<b>Output 4</b>	use trained glove word vectors trained glove word (pennington et. al., 2014) embeddings for our purposes
<b>Output 5</b>	we use pre trained (pennington et. al., 2014) embeddings for our purposes glove

Pada Tabel III-5 ditunjukkan hasil pembangkitan dengan nilai  $\alpha$  sebesar 0.9 untuk MeteorScore dan 0.1 untuk Pinch Score. Arsitektur yang dipakai adalah StoPGEN yang secara kualitatif dan kuantitatif menghasilkan nilai tertinggi. Berdasarkan Tabel III-5 di atas, dari 10 hasil pembangkitan, terdapat variasi dalam kalimat-kalimat yang dihasilkan. Meskipun tidak semua kalimat sempurna secara tata bahasa, secara keseluruhan contoh-contoh tersebut cukup dapat diterima.

Dalam tabel, terlihat bahwa meskipun input dan target memiliki struktur yang sama, model berhasil menghasilkan beberapa variasi dalam frase dan struktur kalimat. Beberapa output menggantikan kata-kata tertentu dengan sinonim atau melakukan perubahan sedikit pada susunan kata-kata. Beberapa output juga menambahkan informasi tambahan, seperti dimensi vektor atau penekanan pada penggunaan kata "pretrained". Meskipun beberapa kalimat mungkin terdengar

sedikit tidak alami atau tidak standar, tetapi secara keseluruhan contoh-contoh tersebut masih dapat diterima.

Variasi dalam hasil pembangkitan kalimat menjadi tujuan yang diinginkan dalam pengembangan model ini. Hal ini dapat menunjukkan bahwa model memiliki kemampuan untuk menghasilkan berbagai alternatif yang masih relevan dengan input dan target yang diberikan. Meskipun masih ada ruang untuk perbaikan dalam hal tata bahasa, namun hasil-hasil yang dihasilkan memberikan indikasi bahwa model sedang dalam proses pembelajaran dan pengembangan yang positif.

### III.4 Perbedaan Metode UPSA dengan StoPGEN

Metode UPSA sebagai baseline dari penelitian ini telah menginspirasi untuk mengembangkan metode pembangkitan parafrase. Dalam mengembangkan metode yang baru, algoritma *simulated annealing* menjadi inspirasi utama dengan pengembangan dalam beberapa komponen yang ada di dalamnya.

Metode UPSA mengembangkan fungsi objektif dengan komponen Semantic Preservation berdasarkan similarity sentence embedding, Expression Diversity berdasarkan inver BLEU Score dan Language Fluency Berdasarkan probabilitas N-GRAM. Metode StoPGEN mengembangkan fungsi objektif dengan template fungsi weighted linear dengan memiliki nilai  $\alpha$  sebagai bobot kecenderungan yang dapat diatur, memiliki komponen Semantic *Similarity* berdasarkan *Meteor Score* dan *Language Divergence* berdasarkan PINC Score.

Metode UPSA memiliki strategi pemilihan unit teks dalam kalimat secara acak. Metode StoPGEN memiliki strategi pemilihan unit teks dalam kalimat secara *Sequential* dari kata pertama hingga kata terakhir dalam kalimat. Strategi ini didapat dengan terlebih dahulu mengadaptasi apa yang dilakukan oleh metode UPSA, lalu kemudian mengembangkannya dan dibandingkan hasil performansinya, sehingga didapat strategi terbaik adalah dengan memilih unit tesk secara berurutan.

Metode UPSA menggunakan *vocabulary* yang degenerate dari dataset untuk menghasilkan kandidat *successor* untuk aksi penambahan atau penggantian unit teks. Metode StoPGEN membuat model bahasa berbasis word Embedding dengan word2vec untuk mendapatkan kandidat successor. Metode UPSA didesain untuk diaplikasikan pada dataset standar yang lebih umum yaitu

Twitter dan Quora. Metode StoPGEN dikembangkan dengan dasar untuk dipakai pada dataset kalimat sitasi. Metode UPSA tidak melakukan pemilihan successor dari beberapa kandidat terpilih. Metode StoPGEN menggunakan probabilitas n-gram untuk menentukan successor terpilih dari beberapa kandidat.

Tabel III-6 Perbandingan Metode StoPGEN dengan UPSA

No	Komponen	StoPGEN	UPSA
1	Fungsi Objektif	$f(x) = (\alpha)ssim(x) + (1-\alpha)ldiv(x)$ <p>dimana:  <math>\alpha</math> : Bobot kecenderungan (1 – 0)                      ssim: Semantic Similarity, Meteor Score                      ldiv: Language Divergence, PINC Score</p>	$f(x) = fsem(x, x_0). fexp(x, x_0). fflu(x)$ <p>dimana:                      fsem: Semantic Preservation, similarity sentence embedding                      fexp: Expression Diversity, inverse BLEU Score                      fflu: Language Fluency, probabilitas N-GRAM</p>
2	Strategi Pemilihan Unit Teks	Sequential	Acak
3	Model Bahasa Successor	Word2vec	Dictionary
4	Data Set	Makalah Ilmiah	Sosial Media Twitter dan Quora
5	Strategi Pemilihan New State	Menghitung Peluang Kemunculan N-Gram	Tidak Ada

#### IV. Kesimpulan

Penelitian ini telah berhasil mengembangkan algoritma *simulated annealing* untuk pembangkitan parafrase kalimat sitasi. Algoritma *simulated annealing* dikembangkan dengan membuat fungsi objektif dan skema operasi yang lebih sederhana. Selain itu, penelitian ini telah membangun korpus *parallel* parafrase kalimat sitasi.

Hasil pembangkitan StoPGEN dibandingkan dengan metode lain pada domain makalah ilmiah. StoPGEN mengungguli model *Supervised* yang biasa digunakan untuk pembangkitan kalimat mesin penerjemah. StoPGEN juga menunjukkan hasil yang lebih baik dari model *Unsupervised* UPSA dan UPSA yang telah disesuaikan sumber bahasanya dengan makalah ilmiah.

Model yang dikembangkan juga menghasilkan performansi yang lebih baik pada dataset public dibandingkan dengan model dari penelitian lainnya. StoPGEN mendapatkan performansi yang unggul untuk dataset twitter dan quora. StoPGEN mendapatkan BLEU score 6.26 untuk dataset twitter dan 22.37 untuk dataset quora, mengungguli model lainnya.

Dalam pengukuran kualitatif dengan survey, variant StoPGEN terbaik mendapatkan nilai penerimaan 50.96. Jika model tersebut dibandingkan dengan metode lain mendapatkan nilai penerimaan sebenar 50.80. Dari hasil pengamatan keluaran, StoPGEN dapat menghasilkan keluaran stokastik dengan tetap menjaga kesamaan semantic namun memiliki perbedaan leksikal.

Berdasarkan uji hipotesis yang ditetapkan, penelitian ini telah membuktikan bahwa Algoritma *simulated annealing* dapat meningkatkan hasil pembangkitan kalimat sitasi dibandingkan dengan metode yang sudah. Penelitian ini mengembangkan algoritma *simulated annealing* dengan cara membentuk fungsi objektif baru, model bahasa, teknik perubahan kalimat dan strategi pemilihan state.

Penelitian ini menyelesaikan persoalan pembangkitkan parafrase pada domain makalah ilmiah dengan tujuan mempertahankan konteks argument kalimat dan keluaran yang stokastik. Konteks argument kalimat sitasi dipertahankan dengan

tidak merubah target sitasi dan hanya fokus pada satu sitasi. Keluaran stokastik memungkinkan pengguna mendapatkan banyak alternatif kalimat sitasi dari satu kalimat input.

## V. Saran

Berdasarkan beberapa kelemahan usulan metode yang ditemukan, beberapa saran pengembangan penelitian antara lain:

1. Menganti teknik perubahan kalimat pada level sintaktik sehingga variasi bentuk kalimat yang dikembangkan menjadi lebih banyak dan lebih kompleks
2. Mempertahankan parasa argumentatif dalam kalimat sitasi sehingga makna dari kalimat baru yang dihasilkan tetap sama dengan kalimat masukan
3. Mengembangkan metode untuk penanganan kalimat sitasi lebih dari satu target sitasi
4. Mengembangkan metode untuk penanganan kalimat majemuk baik setara maupun bertingkat yang sering ditemui dalam kalimat sitasi
5. Mengembangkan komponen pada algoritma *simulated annealing* dengan basis model bahasa besar (*Large Language Model*) baik untuk melakukan parafrase pada domain yang spesifik atau domain yang lebih umum

## Riwayat Hidup

Kandidat Doktor lahir di Sumedang 27 Januari 1990 dan merupakan anak pertama dari pasangan Bapak Cecep Cahya dan Ibu Eri Hariah, kakak dari adik bernama Taufik Arsyad. Kandidat Doktor menyelesaikan studi S1 pada Program Studi Teknik Informatika Universitas Jenderal Achmad Yani tahun 2012, kemudian menyelesaikan Program Magister Informatika di Institut Teknologi Bandung pada tahun 2015. Kandidat Doktor merupakan suami dari seorang istri bernama Delima Istio Prawiradhani Putri dan Ayah dari dua anak yaitu Harrasha Dipri Qahwasyari (5th) dan Heksa Dipri Hammada (2th). Sejak tahun 2017 sampai sekarang Kandidat Doktor bekerja sebagai Dosen di Program Studi S1 Teknik Informatika Universitas Jenderal Achmad Yani.

## Daftar Publikasi

No	Tahun	Judul	Jenis	Penerbit
1	2023	<i>Generating Paraphrase Using Simulated Annealing for Citation Sentences</i>	Jurnal Internasional (Q2)	<i>Informatics</i>
2	2018	<i>Building Candidate Monolingual Parallel Corpus from Scientific Papers</i>	Seminar Internasional	<i>International Conference on Asian Language Processing</i>

Ilyas, R., Khodra, M., Munir, R., Mandala, R., dan Widyantoro, D. (2023): *Generating Paraphrase Using Simulated Annealing for Citation Sentences*, *Informatics*, 10(2), 34. <https://doi.org/10.3390/informatics10020034>

Ilyas, R., Widyantoro, D. H., dan Khodra, M. L. (2018): *Building Candidate Monolingual Parallel Corpus from Scientific Papers*, *2018 International Conference on Asian Language Processing (IALP)*, IEEE, Bandung, Indonesia, 230–233. <https://doi.org/10.1109/IALP.2018.8629246>



## Daftar Referensi

- Androutsopoulos, I., dan Malakasiotis, P. (2010): A survey of paraphrasing and textual entailment methods, *Journal of Artificial Intelligence Research*, **38**, 135–187. <https://doi.org/10.1613/jair.2985>
- Brad, F. (2017): Neural Paraphrase Generation using Transfer Learning.
- Hood, S. (2010): *Appraising research: evaluation in academic writing*, Palgrave Macmillan, Houndmills, Basingstoke, Hampshire ; New York, 227.
- Kauchak, D., dan Barzilay, R. (2006): Paraphrasing for automatic evaluation, *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics* -, Association for Computational Linguistics, New York, New York, 455–462. <https://doi.org/10.3115/1220835.1220893>
- Li, Z., Mou, L., Jiang, X., Lyu, M. R., dan King, I. (9 Juli 2020): Unsupervised Text Generation by Learning from Search, arXiv, diperoleh 15 Maret 2023 melalui situs internet: <http://arxiv.org/abs/2007.08557>.
- Lin, D., dan Pantel, P. (2001): Discovery of inference rules for question-answering, *Natural Language Engineering*, **7**(4), 343–360. <https://doi.org/10.1017/S1351324901002765>
- Liu, X., Mou, L., Meng, F., Zhou, H., Zhou, J., dan Song, S. (2019): Unsupervised paraphrasing by simulated annealing, *arXiv*, 302–312. <https://doi.org/10.18653/v1/2020.acl-main.28>
- McKeown, K. R. (1983): Paraphrasing Questions Using Given and New Information.
- Miao, N., Zhou, H., Mou, L., Yan, R., dan Li, L. (2019): CGMH: Constrained sentence generation by metropolis-hastings sampling, *33rd AAAI Conference on Artificial Intelligence*, **33**(Mcmc), 6834–6842. <https://doi.org/10.1609/aaai.v33i01.33016834>
- Pavlick, E., Rastogi, P., Ganitkevitch, J., Durme, B. V., dan Callison-Burch, C. (2015): PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification, *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers), Beijing, China, July 26-31, 2015*, 425–430.
- Prakash, A., Hasan, S. A., Lee, K., Datla, V., Qadir, A., Liu, J., dan Farri, O. (2016): Neural Paraphrase Generation with Stacked Residual LSTM

Networks, diperoleh melalui situs internet:  
<http://arxiv.org/abs/1610.03098>.

- Quirk, C., Brockett, C., dan Dolan, W. (2003): Monolingual Machine Translation for Paraphrase Generation.
- Russell, S. J., Norvig, P., dan Davis, E. (2010): *Artificial intelligence: a modern approach* (Prentice Hall series in artificial intelligence, 3rd ed), Prentice Hall, Upper Saddle River, 1132.
- Shinyama, Y., Sekine, S., dan Sudo, K. (2002): Automatic paraphrase acquisition from news articles, *HLT '02: Proceedings of the second international conference on Human Language Technology Research*, 313–318. <https://doi.org/10.3115/1289189.1289218>
- Teufel, S. (1999): *Argumentative Zoning : Information Extraction from Scientific Text University of Edinburgh, PhD thesis University of Edinburgh*, 352. <https://doi.org/10.1.1.105.485>
- Teufel, S., Siddharthan, A., dan Tidhar, D. (2006): An annotation scheme for citation function, *COLING/ACL 2006 - SIGdial06: 7th SIGdial Workshop on Discourse and Dialogue, Proceedings of the Workshop*, (July), 80–87. <https://doi.org/10.3115/1654595.1654612>
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., dan Polosukhin, I. (2017): Attention is all you need, *Advances in Neural Information Processing Systems*, **2017-Decem**(Nips), 5999–6009.
- Vizcarra, G., dan Ochoa-Luna, J. (2020): Paraphrase Generation via Adversarial Penalizations, *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, Association for Computational Linguistics, Online, 249–259. <https://doi.org/10.18653/v1/2020.wnut-1.32>
- Xu, W., Callison-burch, C., dan Dolan, W. B. (2015): SemEval-2015 Task 1 : Paraphrase and Semantic Similarity in Twitter ( PIT ), (SemEval), 1–11.
- Zhao, S., Wang, H., Lan, X., dan Liu, T. (2009): Leveraging Multiple MT Engines for Paraphrase Generation.
- Zhou, J., dan Bhat, S. (2021): Paraphrase Generation: A Survey of the State of the Art, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 5075–5086. <https://doi.org/10.18653/v1/2021.emnlp-main.414>

