

***IMAGE CAPTIONING BAHASA INDONESIA DENGAN
MENGUNAKAN VISION-LANGUAGE MODEL***

Laporan Tugas Akhir

Disusun sebagai syarat kelulusan tingkat sarjana

Oleh

RAIHAN ASTRADA FATHURRAHMAN

NIM : 13519113



**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
September 2023**

***IMAGE CAPTIONING BAHASA INDONESIA DENGAN
MENGUNAKAN VISION-LANGUAGE MODEL***

Laporan Tugas Akhir

Oleh

RAIHAN ASTRADA FATHURRAHMAN

NIM : 13519113

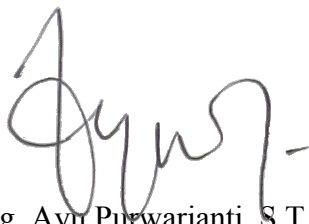
Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung

Telah disetujui dan disahkan sebagai Laporan Tugas Akhir
di Bandung, pada tanggal 15 September 2023

Pembimbing I,



Dr. Eng. Ayu Purwarianti, S.T., M.T.

NIP 197701272008012011

Pembimbing II,



Genta Indra Winata, S.T., Ph. D.

LEMBAR PERNYATAAN

Dengan ini saya menyatakan bahwa:

1. Pengerjaan dan penulisan Laporan Tugas Akhir ini dilakukan tanpa menggunakan bantuan yang tidak dibenarkan.
2. Segala bentuk kutipan dan acuan terhadap tulisan orang lain yang digunakan di dalam penyusunan laporan tugas akhir ini telah dituliskan dengan baik dan benar.
3. Laporan Tugas Akhir ini belum pernah diajukan pada program pendidikan di perguruan tinggi mana pun.

Jika terbukti melanggar hal-hal di atas, saya bersedia dikenakan sanksi sesuai dengan Peraturan Akademik dan Kemahasiswaan Institut Teknologi Bandung bagian Penegakan Norma Akademik dan Kemahasiswaan khususnya Pasal 2.1 dan Pasal 2.2.

Bandung, 15 September 2023



Raihan Astrada Fathurrahman

NIM 13519113

ABSTRAK

IMAGE CAPTIONING BAHASA INDONESIA DENGAN MENGUNAKAN VISION-LANGUAGE MODEL

Oleh

RAIHAN ASTRADA FATHURRAHMAN

NIM : 13519113

Berkembangnya skema *pre-train & fine-tune* yang berhasil memperoleh kinerja yang baik di bidang *computer vision* dan *natural language processing* mendorong banyaknya penelitian yang kemudian mengeksplorasi *Vision-Language Model* atau yang lebih dikenal sebagai Model VL. Penelitian *image captioning* Bahasa Indonesia yang telah dilakukan sebelumnya umumnya masih menggunakan data yang terbatas, baik dari segi kualitas maupun kuantitas. Selain itu, penelitian yang telah dilakukan juga belum memanfaatkan model VL. Padahal, model VL mampu mencapai *state-of-the-art* pada permasalahan *image captioning* karena memiliki generalisasi yang baik dari *pre-training* pada data berskala besar.

Untuk mengatasi kekurangan ini, tugas akhir ini melakukan pembangunan 60,000 data *image captioning* yang diperoleh melalui perbaikan kalimat dari data MSCOCO yang diterjemahkan secara otomatis ke Bahasa Indonesia. *Dataset* tersebut kemudian digunakan untuk melatih model-model VL yang mampu meraih *state-of-the-art* pada data bahasa Inggris, seperti BLIP, GIT, dan OFA, untuk menangani *image captioning* dalam bahasa Indonesia. Model-model tersebut dilatih melalui skema *transfer learning* pada *image captioning dataset* berbahasa Indonesia dengan kualitas dan kuantitas *dataset* yang bervariasi, seperti menggunakan data *machine translated*, *human translated*, dan kombinasi keduanya.

Hasil eksperimen menunjukkan bahwa model BLIP yang dikenai *finetune* dengan gabungan data *machine translated* dan *human translated* memiliki kemampuan adaptasi bahasa terbaik dalam menangani *image captioning* Bahasa Indonesia. Model tersebut berhasil mencapai nilai BLEU 1,2,3,4 secara berturut-turut sebesar 57.9, 43.3, 31.5, 23.2 dan nilai CIDEr sebesar 143.5. Rata-rata nilai BLEU dan CIDEr tersebut meningkat sebesar 78% dan 52% dibandingkan dengan *baseline* yang tidak menggunakan model VL. Selain itu, evaluasi manual menunjukkan bahwa penggunaan data *human translated* bersamaan dengan data *machine translated* mampu memberikan *caption* yang lebih akurat dan alami pada model VL yang digunakan.

Kata kunci: *image captioning, human translated data, machine translated data, vision-language model.*

KATA PENGANTAR

Puji dan syukur Penulis panjatkan kepada Allah SWT, karena atas nikmat dan karunia-Nya Tugas Akhir yang berjudul “*Image Captioning Bahasa Indonesia menggunakan Vision-Language Model*” dapat diselesaikan sebagai salah satu syarat untuk menyelesaikan Tingkat Sarjana (S1) Program Studi Teknik Informatika di Institut Teknologi Bandung.

Penyelesaian Tugas Akhir ini dapat dilakukan karena bantuan dan bimbingan dari banyak pihak. Penulis mengucapkan terima kasih kepada seluruh pihak yang telah membantu Penulis, baik secara langsung maupun tidak langsung, selama pengerjaan tugas ini. Penulis mengucapkan terima kasih kepada:

1. Ibu Dr. Eng. Ayu Purwarianti, S.T., Saudara Genta Indra Winata, S.T., Ph.D., dan Saudara Samuel Cahyawijaya, S.T., M.Phil., selaku dosen pembimbing Tugas Akhir yang senantiasa memberikan nasihat, arahan, dan juga bimbingan selama proses pengerjaan Tugas Akhir.
2. Bapak Ir. Windy Gambetta, M.B.A., dan Bapak Ir. Rila Mandala, M.Eng., Ph.D. selaku dosen penguji yang telah memberikan kritik, saran, dan masukan untuk Tugas Akhir penulis.
3. Seluruh staf pengajar Program Studi Teknik Informatika ITB yang telah memberikan ilmu dan pengalaman yang sangat berharga selama masa perkuliahan.
4. Seluruh karyawan dan staf Tata Usaha STEI ITB yang telah membantu menyelesaikan segala proses administrasi, terutama dalam keperluan Tugas Akhir.
5. Kedua orangtua yang senantiasa memberikan dukungan penuh, secara moral, materiil, dan doa, terhadap seluruh proses yang Penulis jalani selama perkuliahan di ITB.
6. Kedua saudari, Nisa dan Nana, yang senantiasa menyemangati dan memberikan motivasi selama berkuliah di ITB.

7. Teman-teman dari 'Fujeeholic' yaitu David, Dhimas, dan Fasya yang telah membantu menjadi *annotator* dalam pembuatan *dataset* serta menyemangati & menemani penulis selama berkuliah di ITB.
8. Teman-teman dari 'group aspol' yaitu Fauzan, Krishna, dan Wishnu yang telah menyemangati, menemani, dan menyediakan tempat untuk beristirahat penulis terutama selama menjalani tingkat akhir di ITB.
9. Teman-teman satu bimbingan, yaitu Dehan, Rahma, dan Wilbert, yang memotivasi penulis untuk selalu memberikan progress di setiap minggunya.
10. Teman-teman dari 'Support Group' yaitu Alif, Andres, Aras, Daffa, Diaz, Dims, Dzaki, Fabhian, Feihan, Andro, Naufal, Dika, Cubing, Asadel, Reihan, Rezda, Ryo, Tugus, Viel yang telah membantu Penulis dalam mengerjakan tugas dan menghibur Penulis selama perkuliahan.
11. Teman-teman dari 'GWS Lort' yaitu Alam, Allief, Azhar, Dito, Faziz, Jafar, Faiq, Bowo, Randy, Vaza, Widya.
12. Teman-teman penulis lainnya yaitu Alghi, Chris, Fadel, Irvin, Tamir, Yahya.
13. Teman-teman satu angkatan STEI dan ASYNC 2019 yang menjadi teman seperjuangan selama perkuliahan.

Harapannya tugas akhir ini dapat memberikan kontribusi dan manfaat untuk pengembangan ilmu pengetahuan dan semoga dapat berguna bagi pembacanya.

Bandung, 15 September 2023



Raihan Astrada Fathurrahman

NIM 13519113

DAFTAR ISI

BAB I PENDAHULUAN.....	1
I.1 Latar Belakang.....	1
I.2 Rumusan Masalah.....	3
I.3 Tujuan	4
I.4 Batasan Masalah	4
I.5 Metodologi.....	4
I.6 Sistematika Pembahasan.....	6
BAB II STUDI LITERATUR	8
II.1 Image Captioning	8
II.1.1 Arsitektur Seq2Seq	8
II.2 Vision-Language Model.....	9
II.2.1 Arsitektur VL	10
II.2.2 Pre-training Objectives VL	13
II.3 Image Captioning Dataset	15
II.4 Evaluasi Kinerja.....	16
II.4.1 Automated Metrics	16
II.4.2 Human Evaluation	21
II.5 Perplexity	21
II.6 XGLM (Lin dkk., 2021)	22
II.7 Penelitian Terkait.....	23
II.7.1 BLIP (Li dkk., 2022).....	23
II.7.2 GIT (Wang dkk., 2022a)	24

II.7.3	OFA (Wang dkk., 2022b)	26
II.7.4	SCN & Soft Attention Indo (Sinurat, 2019)	27
BAB III	ANALISIS MASALAH DAN RANCANGAN SOLUSI.....	30
III.1	Analisis Persoalan	30
III.2	Analisis Solusi	33
III.3	Deskripsi Rancangan Solusi	37
III.3.1	Pembuatan <i>Dataset</i>	37
III.3.2	Eksperimen	39
BAB IV	IMPLEMENTASI, EKSPERIMEN, DAN EVALUASI.....	42
IV.1	Pembangunan <i>Image Captioning Dataset</i> Bahasa Indonesia.....	42
IV.2	Desain Eksperimen	47
IV.3	Hasil Eksperimen	49
IV.4	Evaluasi	53
IV.4.1	Analisis Hasil Eksperimen.....	57
IV.4.2	Analisis Error Hasil.....	59
BAB V	KESIMPULAN DAN SARAN	60
V.1	Kesimpulan	60
V.2	Saran	60

DAFTAR LAMPIRAN

Lampiran A. Parameter Eksperimen pada BLIP, OFA, dan GIT.....	67
Lampiran B. Panduan Proses Anotasi	68
Lampiran C. Rubrik <i>Human Evaluation</i>	69
Lampiran D. Contoh Hasil.....	71

DAFTAR GAMBAR

Gambar II.1. Arsitektur Umum Seq2Seq pada <i>Image Captioning</i>	9
Gambar II.2. <i>Pre-training</i> pada model ViLBERT (Lu dkk., 2019).....	10
Gambar II.3. Arsitektur <i>Single-stream</i> (Shin dkk., 2022).....	11
Gambar II.4. Arsitektur <i>Dual-stream</i> (Shin dkk., 2022).....	12
Gambar II.5. Arsitektur BLIP (Li dkk., 2022)	23
Gambar II.6. <i>Pre-training</i> BLIP (Li dkk., 2022)	24
Gambar II.7. Arsitektur GIT (Wang dkk., 2022a)	25
Gambar II.8. Arsitektur OFA (Wang dkk., 2022b).....	27
Gambar II.9. Arsitektur SCN & <i>Soft Attention</i> Indo (Sinurat, 2019).....	28
Gambar III.1. Contoh Hasil Terjemahan <i>Dataset</i>	32
Gambar III.2. Alur Pembangunan <i>Dataset</i>	37
Gambar III.3. Alur <i>Fine-tuning HT / MT Dataset</i>	40
Gambar III.4. Alur <i>Fine-tuning</i> dengan kombinasi HT & MT <i>Dataset</i>	41
Gambar IV.1. Contoh kategori kesalahan <i>Typos & Mechanic</i>	44
Gambar IV.2. Contoh kategori kesalahan <i>Translation</i>	44
Gambar IV.3. Contoh kategori kesalahan <i>Word Edit</i>	45
Gambar IV.4. Contoh kategori kesalahan <i>Major Changes</i>	46
Gambar IV.5. Hasil <i>Human Evaluation</i> Kategori ‘ <i>Similarity</i> ’	54
Gambar IV.6. Hasil <i>Human Evaluation</i> Kategori ‘ <i>Suitability</i> ’	55
Gambar IV.7. Hasil <i>Human Evaluation</i> Kategori ‘ <i>Naturalness</i> ’	56
Gambar B.1. Panduan Perbaikan Terjemahan <i>Dataset</i>	68
Gambar C.1. Rubrik Penilaian <i>Similarity</i>	69
Gambar C.2. Rubrik Penilaian <i>Suitability</i>	70

Gambar C.3. Rubrik Penilaian <i>Naturalness</i>	70
---	----

DAFTAR TABEL

Tabel III.1. Perbandingan Penelitian <i>Image Captioning</i> Bahasa Indonesia.....	30
Tabel III.2. Perbandingan Karakteristik <i>Dataset Image Captioning</i>	34
Tabel III.3. Perbandingan Metrik <i>Image Captioning</i>	35
Tabel IV.1. Statistik Jenis Perubahan	43
Tabel IV.2. Rincian Lingkungan Eksperimen.....	48
Tabel IV.3. Hasil Eksperimen terhadap Model BLIP	49
Tabel IV.4. Hasil Eksperimen terhadap Model OFA.....	50
Tabel IV.5. Hasil Eksperimen terhadap Model GIT	52
Tabel IV.6. Hasil <i>Image Captioning</i> Bahasa Indonesia.....	53
Tabel A.1. <i>Hyperparameter</i> eksperimen BLIP, OFA, dan GIT.....	67
Tabel D.1. Hasil <i>Image Captioning</i>	71

DAFTAR ISTILAH

Istilah	Penjelasan
<i>Computer vision</i>	Menggunakan pembelajaran mendalam untuk menganalisis data visual.
<i>Convolutional neural network</i>	Salah satu jenis <i>neural network</i> yang menerapkan lapisan konvolusi, biasanya digunakan untuk mendeteksi dan mengenali objek pada sebuah gambar.
<i>Dataset</i>	Kumpulan data yang digunakan dalam proses pembelajaran mesin.
<i>Decoder</i>	Bagian yang mengubah <i>input</i> yang telah di- <i>encode</i> menjadi sebuah teks.
<i>Encoder-decoder</i>	Sebuah arsitektur yang digunakan pada <i>task</i> dalam <i>machine learning</i> yang menggunakan dua komponen utama, <i>encoder</i> dan <i>decoder</i> .
<i>Framework</i>	Kerangka kerja yang digunakan untuk mengembangkan sebuah tugas tertentu.
<i>Fine-tuning</i>	Menggunakan pengetahuan yang diperoleh sebelumnya untuk permasalahan baru.

Istilah	Penjelasan
<i>Image captioning</i>	Pemberian deskripsi terhadap gambar.
<i>Long Short-Term Memory</i>	Modifikasi dalam sebuah <i>recurrent neural network</i> , yang mampu mengingat kumpulan informasi yang telah disimpan dalam jangka panjang, serta menghapus informasi yang tidak relevan.
<i>Modality</i>	Jenis sumber data masukan yang dianalisis model, dapat berupa teks gambar, suara, video.
<i>Natural language processing</i>	Menggunakan pembelajaran mendalam untuk menganalisis data bahasa manusia.
<i>Pre-training</i>	Melatih terlebih dahulu model dengan suatu <i>task</i> atau <i>dataset</i> .
<i>Pre-trained models</i>	Model yang memanfaatkan data <i>pretraining</i> .
<i>State-of-the-art</i>	Level tertinggi & terkini dari perkembangan sebuah alat/teknik/topik tertentu.

Istilah	Penjelasan
<i>Task</i>	Suatu tugas yang dilakukan oleh mesin.
<i>Transformers</i>	Arsitektur model yang digunakan untuk pemrosesan bahasa alami menggunakan mekanisme <i>self-attention</i> .
<i>Vision-language model</i>	Model yang dapat memproses masukan visual dan tekstual yang didesain untuk memahami dan membuat relasi antar keduanya.
<i>Vision transformers</i>	Arsitektur <i>transformers</i> yang diterapkan pada data gambar dalam pemrosesan citra.

BAB I

PENDAHULUAN

Bab Pendahuluan berisi penjelasan mengenai landasan kerja dan arah kerja pada pembuatan tugas akhir mengenai *Image Captioning* Bahasa Indonesia dengan menggunakan *Vision-Language Model*. Bab ini terdiri dari beberapa subbab, yaitu latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi, dan sistematika pembahasan.

I.1 Latar Belakang

Image captioning merupakan sebuah *task* untuk memberikan sebuah deskripsi tekstual dari suatu gambar. Proses *image captioning* menggabungkan dua buah domain dalam bidang kecerdasan buatan, yaitu *computer vision* dan juga *natural language processing*. Model dalam *image captioning* perlu untuk mempelajari dan menggabungkan dua bentuk informasi. Pertama, informasi tentang objek dalam gambar, dan kedua, pemahaman tentang kalimat deskripsi. Pemahaman dari kedua informasi tersebut kemudian dimanfaatkan untuk menciptakan kalimat deskripsi yang memiliki makna dan tata bahasa yang benar (Stefanini dkk., 2022).

Selama beberapa tahun belakangan, minat terhadap penelitian dalam bidang *image captioning* telah meningkat secara signifikan. Pada awalnya, untuk menyelesaikan *task* terkait *image captioning*, digunakan sebuah pendekatan *information retrieval*. Pada pendekatan tersebut, satu kalimat diambil dari kumpulan kalimat yang telah ditetapkan sebelumnya berdasarkan sebuah *query* gambar (Farhadi dkk., 2010; Ordonez dkk., 2011; Hodosh dkk., 2013). Selain pendekatan *information retrieval*, terdapat juga pendekatan menggunakan templat. Dalam pendekatan templat, model mendeteksi kumpulan informasi visual dan menghasilkan kalimat melalui hubungan dengan gambar atau aturan tata bahasa tertentu (Mitchell dkk., 2012; Ushiku dkk., 2012). Dengan berkembangnya *deep learning*, model *image captioning* kemudian mulai mengadopsi *neural network* yang mengikuti arsitektur *encoder-decoder* (Vinyals dkk., 2016; Wang dkk., 2016; You dkk., 2016).

Dalam sebuah arsitektur *encoder-decoder*, *encoder* akan menerjemahkan gambar masukan menjadi representasi yang kemudian digunakan oleh *language model* pada *decoder* untuk menghasilkan *caption* (Bai & An, 2018). Pada awalnya, strategi yang banyak dilakukan adalah dengan memanfaatkan *pre-trained models* secara terpisah untuk bagian *vision* dan *language*. *Pre-trained models* tersebut telah dilatih sebelumnya pada *task* lain dengan menggunakan data label berskala besar. Model selanjutnya dikenai *fine-tune* pada data khusus untuk *image captioning task* yang berisi pasangan gambar dan deskripsi teks terkait. Proses *fine-tuning* tersebut dilakukan agar model mempelajari *grounding* antara *vision* dan *language* (Lu dkk., 2019).

Seiring berkembangnya teknologi, kemudian muncul model-model yang memanfaatkan skema *pre-training* pada data skala besar & *fine-tuning* pada *task* yang lebih spesifik. Penggunaan skema *pre-train* & *fine-tune* berhasil memperoleh kinerja yang baik di bidang *computer vision* dan *natural language processing*. Hal tersebut kemudian memicu penelitian yang melatih model berskala besar pada kedua buah *modality* yang digunakan pada dua bidang tersebut, yaitu *vision* dan *language*, yang kemudian lebih dikenal dengan *Vision-Language Model* (Lu dkk., 2019; Li dkk., 2019; Tan & Bansal., 2019). Terdapat tiga komponen utama pada sebuah *Vision-Language (VL) Models*, yaitu *encoder* untuk memproses tiap *modality*, interaksi antara kedua *modality*, dan *pre-training objective* saat melakukan *pre-training* (Du dkk., 2022; Long dkk., 2022). Proses *pre-training* dilakukan pada kumpulan teks dan gambar berskala besar, sehingga model VL dapat mempelajari representasi universal antar *modality*, yang bermanfaat untuk mencapai kinerja yang kuat dalam *downstream task* pada *vision* dan *language*, salah satunya yaitu *image captioning* (Du dkk., 2022).

Representasi universal yang terdapat pada model VL kemudian mampu dimanfaatkan pada kasus *image captioning* bahasa Indonesia. Pada penelitian] *image captioning* bahasa Indonesia yang telah dilakukan sebelumnya, permasalahan terdapat pada penggunaan *pre-trained models* yang dilakukan secara terpisah (Nugraha & Arifianto, 2019; Sinurat, 2019; Mulyanto dkk., 2019;

Mulyawan dkk., 2021). Padahal, penggunaan *pre-trained models* secara terpisah menghasilkan generalisasi yang kurang baik ketika menggunakan data yang terbatas (Lu dkk., 2019).

Selain itu, Bahasa Indonesia merupakan bahasa yang masih tergolong sebagai *low-resource language* (Wilie dkk., 2020). Saat ini belum terdapat banyak *image captioning dataset* berbahasa Indonesia yang memiliki kualitas yang baik. Penelitian-penelitian sebelumnya dilakukan menggunakan *dataset* berbahasa Inggris yang diterjemahkan secara otomatis ke Bahasa Indonesia. Hal tersebut kemudian mampu mengurangi makna dari kalimat serta adanya *noise* pada data (Dashtipour dkk., 2016). Kualitas dari *dataset* yang kurang baik tersebut kemudian dapat memengaruhi kinerja model yang dihasilkan. Oleh karena itu, pada penelitian-penelitian sebelumnya telah dilakukan beberapa upaya untuk menanggulangi hal tersebut. Namun demikian, meskipun adanya upaya tersebut, ketersediaan *dataset* terkait *image captioning* berbahasa Indonesia yang memiliki kualitas yang baik masih sangat terbatas.

I.2 Rumusan Masalah

Berdasarkan latar belakang yang telah disampaikan, terdapat beberapa penelitian yang telah mencoba untuk mengembangkan model *image captioning* bahasa Indonesia. Namun demikian, penelitian-penelitian tersebut masih menggunakan data yang terbatas, baik dari segi kualitas maupun kuantitas. Selain itu, penelitian yang telah dilakukan juga belum memanfaatkan model VL. Padahal model-model VL mampu mencapai *state-of-the-art* pada permasalahan *image captioning* karena memiliki generalisasi yang baik dari *pre-training* pada data berskala besar.

Berdasarkan persoalan yang telah dipaparkan, tugas akhir ini menyelesaikan beberapa rumusan masalah serta menjawab pertanyaan berikut:

1. Seberapa baik adaptasi model-model VL dalam menangani permasalahan *image captioning* bahasa Indonesia?

2. Bagaimana dampak dari penggunaan data yang melewati proses pengecekan kualitas oleh manusia terhadap kinerja model VL pada *image captioning* bahasa Indonesia?

I.3 Tujuan

Tujuan yang ingin dicapai dari tugas akhir ini adalah sebagai berikut:

1. Menguji kemampuan adaptasi model-model VL pada *image captioning* bahasa Indonesia.
2. Menganalisis dampak penggunaan data yang melewati proses pengecekan kualitas oleh manusia terhadap kinerja model VL pada *image captioning* Bahasa Indonesia.

I.4 Batasan Masalah

Batasan-batasan yang digunakan dalam dari tugas akhir yang dilakukan adalah sebagai berikut:

1. Model VL yang digunakan untuk *image captioning* adalah BLIP (Li dkk., 2022), GIT (Wang dkk., 2022a), dan OFA (Wang dkk., 2022b).

I.5 Metodologi

Metodologi yang dilakukan dalam menyelesaikan tugas akhir ini adalah sebagai berikut:

1. Analisis permasalahan

Pengerjaan tugas akhir diawali dengan mengidentifikasi dan menganalisis permasalahan yang ada pada *image captioning* bahasa Indonesia. Masalah yang diidentifikasi berupa permasalahan terkait kualitas dan kuantitas data yang digunakan pada *image captioning* Bahasa Indonesia, serta belum dimanfaatkannya model VL pada *image captioning* Bahasa Indonesia.

2. Perancangan solusi

Tahap perancangan solusi dilakukan dengan merumuskan solusi dan skema eksperimen yang dilakukan. Pada tahapan ini, didapatkan sebuah metode solusi yang menerapkan model-model VL untuk *image captioning* berbahasa Indonesia, serta eksperimen berupa variasi penggunaan jenis kualitas dan kuantitas data pada proses *fine-tuning*.

3. Pembuatan *dataset*

Tahap pembuatan *dataset* dilakukan dengan mempersiapkan & membuat *dataset* yang digunakan pada eksperimen. Pada tahapan ini, pertama dilakukan penerjemahan otomatis pada *dataset* MSCOCO (Lin dkk., 2014) ke dalam Bahasa Indonesia. Selanjutnya, dilakukan proses anotasi berupa pengecekan kualitas & perbaikan pada 60,002 sampel hasil terjemahan *dataset*. Para annotator mengecek hasil terjemahan kalimat serta melakukan perbaikan kalimat pada hasil terjemahan tersebut. Proses perbaikan kalimat dilakukan dengan menggunakan hasil terjemahan dari sistem penerjemah sebagai acuan dalam melakukan perbaikan.

4. Persiapan Eksperimen

Tahap persiapan eksperimen dilakukan dengan menginisiasi model VL yang digunakan dalam eksperimen. Model dicoba untuk dijalankan dan jika terdapat eror maka dilakukan perbaikan terhadap eror tersebut. Selanjutnya, dilakukan konfigurasi terhadap lingkungan eksperimen yang digunakan. Pada tahapan ini, dipastikan eksperimen telah siap dilakukan, sebelum selanjutnya dijalankan dan dievaluasi.

5. Eksperimen

Tahap eksperimen dilakukan dengan melakukan pelatihan model-model VL pada data *image captioning* berbahasa Indonesia. Skema eksperimen dilakukan dengan melakukan *fine-tuning* model VL pada berbagai variasi kualitas dan kuantitas data, seperti menggunakan *dataset* Bahasa Indonesia yang diterjemahkan secara otomatis, menggunakan *dataset* Bahasa Indonesia yang telah diperbaiki oleh annotator, serta menggunakan

gabungan kedua jenis data. Setelah model dilatih, model kemudian diuji dan dievaluasi dengan menggunakan data *test*.

6. Analisis Hasil

Dari hasil eksperimen yang diperoleh, dilakukan analisis terhadap hasil *caption* gambar dalam bahasa Indonesia. Analisis tersebut dilakukan dengan menggunakan beberapa metrik evaluasi, yaitu BLEU, CIDEr, dan juga *human evaluation*. Selain itu, dilakukan juga analisis eror pada *caption* yang dihasilkan oleh model.

7. Kesimpulan

Pada tahapan ini, ditarik sebuah kesimpulan untuk menjawab rumusan masalah. Hasil dari kesimpulan menentukan keberhasilan tujuan dari tugas akhir yang telah ditetapkan sebelumnya.

I.6 Sistematika Pembahasan

Laporan tugas akhir ini terdiri dari beberapa bab, yaitu Bab I Pendahuluan, Bab II Studi Literatur, Bab III Analisis Permasalahan dan Rancangan Solusi, Bab IV Implementasi, Eksperimen, dan Evaluasi, serta Bab V Kesimpulan dan Saran.

Bab I Pendahuluan berisi mengenai landasan dan arah kerja pada pembuatan tugas akhir mengenai *Image Captioning* Bahasa Indonesia dengan menggunakan *Vision-Language Model*. Bab ini terdiri dari beberapa subbab, yaitu latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi, dan sistematika pembahasan.

Bab II Studi Literatur berisi mengenai beberapa pemahaman dari literatur-literatur lain yang berkaitan dengan tugas akhir ini, yaitu *Image Captioning* Bahasa Indonesia dengan menggunakan *Vision-Language Model*. Kajian literatur mencakup pembahasan terkait *image captioning*, *vision-language model*, *image captioning dataset*, metrik evaluasi, *perplexity*, XGLM, serta penelitian-penelitian terkait.

Bab III Analisis Masalah dan Rancangan Solusi berisi mengenai hasil analisis terhadap persoalan *Image Captioning* Bahasa Indonesia dengan menggunakan *Vision-Language Model*, analisis terkait alternatif solusi yang dapat dilakukan untuk menyelesaikan persoalan, serta deskripsi rancangan solusi yang dilakukan pada tugas akhir.

Bab IV Implementasi, Eksperimen, dan Evaluasi berisi hasil dari solusi terhadap masalah yang telah dirumuskan. Bab ini terdiri dari beberapa subbab yang meliputi implementasi yang dilakukan, eksperimen yang dilakukan, serta evaluasi dan analisis terhadap hasil dari eksperimen.

Bab V Kesimpulan dan Saran berisi dua subbab yaitu kesimpulan dan saran. Pada bagian kesimpulan dijelaskan lebih lanjut terkait kesimpulan yang menjawab tujuan tugas akhir. Kemudian, subbab saran berisi saran-saran pengembangan yang dapat dilakukan selanjutnya berkaitan dengan penelitian tugas akhir ini.

BAB II

STUDI LITERATUR

Pada bagian studi literatur dijelaskan lebih lanjut terkait beberapa pemahaman dari literatur-literatur lain yang berkaitan dengan tugas akhir.

II.1 Image Captioning

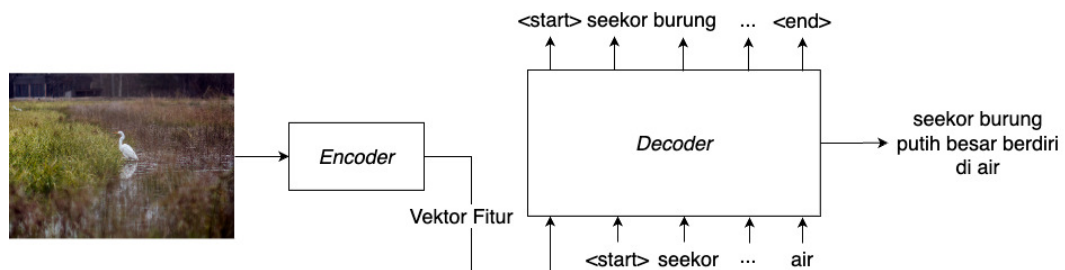
Image captioning adalah permasalahan memberikan sebuah deskripsi dalam bentuk bahasa alami pada konten yang berupa gambar. *Image captioning* menggabungkan kemampuan mesin dalam memahami suatu objek gambar dengan kemampuan mesin untuk menghasilkan kalimat deskripsi yang memiliki makna dan sintaksis yang benar. Hal itu dilakukan dengan menggunakan kerangka *encoder-decoder* (Stefanini dkk., 2022).

Image captioning dapat dilihat sebagai pengembangan lebih lanjut dari *task* klasifikasi gambar. Namun demikian, model *image captioning* merupakan model yang lebih detail dalam mendeskripsikan sebuah gambar. Hal tersebut karena model *image captioning* tidak hanya mengklasifikasikan gambar berdasarkan konten visualnya, tetapi juga menggunakan detail semantiknya, serta atribut dan hubungan ruang yang ada pada elemen-elemen visual (Sharma & Padha, 2023).

II.1.1 Arsitektur Seq2Seq

Arsitektur Seq2Seq pada permasalahan *image captioning* terinspirasi dari penggunaan *encoder-decoder framework* pada permasalahan *neural machine translation*. Arsitektur ini pada awalnya dirancang untuk menerima kalimat suatu Bahasa dan kemudian menerjemahkan kalimat tersebut ke Bahasa lain. Arsitektur tersebut kemudian diadaptasi pada permasalahan *image captioning* dengan membuat sebuah gambar sebagai masukan dan keluarannya adalah sebuah kalimat yang merupakan *caption* dari gambar tersebut. (Bai & An, 2018). Penggunaan *encoder-decoder* kemudian menjadi dasar dari berbagai metode *image captioning*

yang dilakukan. Arsitektur Seq2Seq pada *image captioning* secara umum dapat ditunjukkan oleh Gambar II.1.

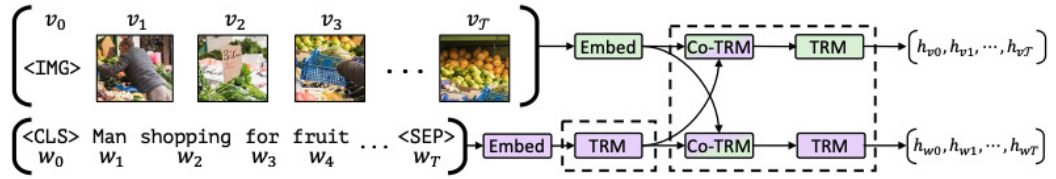


Gambar II.1. Arsitektur Umum Seq2Seq pada *Image Captioning*

Pada Gambar II.1 umumnya sebuah *encoder*, dapat berupa *convolutional neural network* (CNN) ataupun *vision transformer*, digunakan untuk melakukan *encoding* untuk mendapatkan sebuah representasi dari gambar. Representasi tersebut kemudian di-*decode* oleh *decoder*, dapat berupa *Long Short-Term Memory* (LSTM) ataupun *transformer*, untuk menghasilkan *caption* dari gambar.

II.2 *Vision-Language Model*

Vision-Language Model atau Model VL, merupakan sebuah model yang dirancang untuk mempelajari representasi gabungan dari informasi visual dan teks. Representasi gabungan tersebut digunakan pada model VL agar dapat menyelesaikan berbagai macam *task* yang berkaitan dengan *vision* dan *language*, seperti *image captioning & visual question answering* (Mogadala dkk., 2021). Model VL telah di *pre-train* pada kumpulan data berskala besar dan kemudian dilakukan *fine-tuning* pada *downstream task* tertentu. *Pre-training* pada kumpulan gambar-teks berskala besar memungkinkan model VL untuk mempelajari representasi *cross-modal* universal, yang kemudian bermanfaat untuk mencapai kinerja yang baik pada *downstream task* dari *vision-language* (Du dkk., 2022).



Gambar II.2. *Pre-training* pada model ViLBERT (Lu dkk., 2019)

Contoh proses *pre-training* pada model VL dapat dilihat pada Gambar II.2. Berdasarkan Gambar II.2, terdapat 3 langkah untuk melakukan *pre-training* pada model VL. Langkah pertama adalah melakukan *encoding* pada gambar dan teks ke dalam representasi yang memiliki sebuah semantik. Langkah kedua, adalah merancang sebuah arsitektur untuk memodelkan interaksi antara kedua modalitas. Langkah ketiga, adalah menyusun *pre-training objective* yang efektif untuk melatih model VL. Terakhir, model VL yang telah mempelajari fitur *vision-language* secara universal, kemudian dapat dikenai *fine-tuned* pada *downstream task* (Du dkk., 2022).

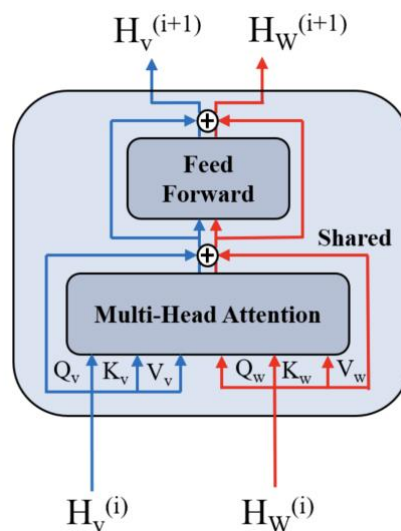
II.2.1 Arsitektur VL

Menurut Du dkk. (2022), berdasarkan konteks menggabungkan informasi dari berbagai modalitasnya, model VL dapat diklasifikasikan menjadi tiga jenis utama yaitu *fusion encoder*, *dual encoder*, dan kombinasi antara keduanya.

Fusion Encoder merupakan tipe *encoder* yang memanfaatkan penggabungan *embedding* teks dan fitur gambar sebagai input awal. Kemudian, interaksi antara *vision* dan *language* dimodelkan dengan menggunakan beberapa teknik penggabungan, seperti menggunakan *self-attention* atau *cross-attention*. Hasil penggabungan ini tersedia dalam bentuk *hidden state* dari lapisan terakhir, yang pada akhirnya dianggap sebagai sebuah representasi gabungan dari kedua modalitas teks dan gambar. Pada *fusion encoder*, terdapat dua skema penggabungan yang

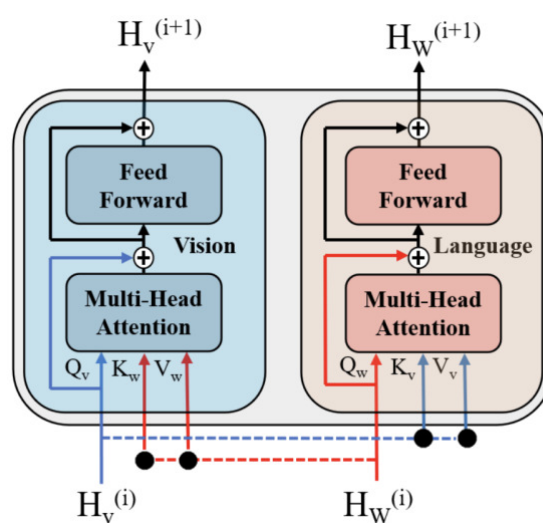
dapat digunakan untuk merepresentasikan *cross-modal interaction*, yaitu *single-stream architecture* dan *dual-stream architecture*.

Single-stream architecture merupakan arsitektur yang mengasumsikan bahwa korelasi dan *alignment* antara dua modalitas cenderung bersifat sederhana, sehingga dapat ditangani dengan satu *encoder* berbasis *transformer*. Pada *single-stream architecture*, *embedding* teks dan fitur gambar digabungkan dan kemudian diperkaya dengan *embedding* khusus untuk menandakan posisi atau jenis modalitas, sebelum kemudian diarahkan ke *encoder* berbasis *transformer*. Kelebihan yang ditemukan pada arsitektur ini adalah kemampuannya dalam menghadapi variasi masukan pada berbagai *downstream task*, misalnya pada *image captioning* yang memerlukan input *image-caption* atau *visual question answering* yang melibatkan *question*, *answer*, dan *image*. *Single-stream architecture* mampu untuk mengatasi *task* tersebut dengan lebih baik karena sifat dari *transformer* yang dapat memahami representasi tidak terstruktur secara lebih baik. Namun, kekurangan juga terdapat karena arsitektur ini mempelajari *task* berdasarkan gabungan kedua *modality*. Dengan adanya penggabungan tersebut, interaksi antar modalitas yang lebih kompleks menjadi sulit untuk ditangkap dan diproses dengan lebih baik.



Gambar II.3. Arsitektur *Single-stream* (Shin dkk., 2022)

Untuk mengatasi keterbatasan terhadap *single-stream architecture*, dibentuklah *dual-stream architecture*. *Dual-stream architecture* merupakan arsitektur yang mengasumsikan bahwa interaksi di dalam satu modalitas dan interaksi antarmodal perlu dibedakan dengan baik agar mampu menghasilkan representasi-representasi multimodal yang lebih mendalam dan berkualitas. Pada arsitekturnya, *dual-stream architecture* memanfaatkan mekanisme *cross-attention*, yang mana vektor *query* diambil dari salah satu modalitas, sementara vektor *key* & *value* diambil dari modalitas lainnya. *Cross-attention layer* umumnya mengandung dua *unidirectional cross-attention sub-layers*, yaitu dari *language* ke *vision* dan sebaliknya. *Sub-layers* ini bertanggung jawab untuk saling bertukar informasi dan melakukan *alignment* antar kedua modalitas. Kelebihan dari arsitektur ini adalah kemampuannya untuk dapat lebih mendalami hubungan di antara modalitas yang digunakan. Meski demikian, arsitektur ini memanfaatkan sumber daya komputasi yang lebih besar jika dibandingkan *single-stream architecture*. Hal ini disebabkan karena adanya penerapan skema penggabungan yang lebih rumit. Jadi, meskipun *dual-stream architecture* memberikan peningkatan dalam hal interaksi antar modalitas, diperlukan komponen interaksi yang lebih kompleks untuk dapat mendapatkannya.



Gambar II.4. Arsitektur *Dual-stream* (Shin dkk., 2022)

Dual Encoder merupakan *encoder* yang muncul dari kekurangan *fusion encoder* yang menggunakan sumber daya komputasi yang besar untuk memodelkan interaksi. Berbeda dengan *fusion encoder*, *dual encoder* menggunakan *encoder* yang terpisah untuk setiap *modality* yang digunakan. Selain itu, *dual encoder* juga menggunakan metode penggabungan yang sederhana, seperti *shallow attention layer* atau *dot product* untuk memproyeksikan ke ruang vektor yang sama. *Encoder* jenis ini lebih efisien untuk digunakan pada task *image-retrieval* karena metode penggabungan yang cukup sederhana. Akan tetapi, *dual encoder* kurang bagus untuk memahami relasi antar *modality*.

Kombinasi Fusion Encoder dan Dual Encoder merupakan *encoder* yang mengkombinasikan kedua tipe dari arsitektur *fusion encoder* dan *dual encoder*. Penggunaan kombinasi kedua *encoder* tersebut merupakan upaya untuk memaksimalkan kedua manfaat dari arsitektur lain untuk mendapatkan hasil yang lebih baik. Salah satu contoh model yang menerapkan ini adalah model FLAVA (Singh dkk., 2022). Pada model FLAVA, implementasi *dual encoder* digunakan untuk mendapatkan representasi tiap modalitas secara terpisah, yang nantinya akan diteruskan kepada *fusion encoder*. Pemrosesan secara terpisah tersebut memberikan dasar yang lebih baik pada saat membuat representasi *cross-modal*.

II.2.2 Pre-training Objectives VL

Untuk membuat model VL mampu memahami korelasi antara *vision* dengan *language*, maka digunakan sebuah cara tertentu untuk melakukan *pre-training* pada model yang disebut dengan *pre-training objectives*. Menurut Zhang dkk. (2023), secara umum *pre-training objective* pada model VL dapat dikategorikan menjadi 3 bagian yaitu *contrastive objective*, *generative objective*, dan *alignment objective*.

Contrastive objective merupakan sebuah objektif agar VL mempelajari representasi yang diskriminatif dengan cara mendekatkan sampel yang mirip serta menjauhkan yang tidak mirip pada sebuah ruang fitur tertentu. Beberapa contoh objektif yang termasuk dalam kategori ini yaitu *Image Contrastive Learning*, *Image-Text Contrastive Learning*, dan *Image-Text-Label Contrastive Learning*. *Image*

Contrastive Learning merupakan objektif yang bertujuan mempelajari fitur gambar secara diskriminatif dengan memaksa *query* gambar untuk dekat dengan *key* positif (data augmentasi) dan jauh dari *key* negatif (gambar lain) di ruang *embedding* yang sama. Kemudian, *Image-Text Contrastive Learning* merupakan objektif yang bertujuan mempelajari representasi gambar-teks yang diskriminatif dengan cara menarik *embedding* dari pasangan gambar dan teks yang sesuai menjadi semakin dekat dan menjauhkan gambar dan teks yang lain. Terakhir, *Image-Text-Label Contrastive Learning* merupakan objektif yang melakukan *encoding* terhadap gambar, teks, dan label ke dalam ruang fitur yang sama.

Generative objective merupakan sebuah objektif agar model VL mampu mempelajari fitur semantik dengan cara melatih model untuk menghasilkan data gambar/teks. Beberapa contoh objektif yang termasuk dalam kategori ini yaitu, *Masked Image Modelling*, *Masked Language Modelling*, *Masked Cross-Modal Modelling*, dan *Image-to-Text Generation*. *Masked Image Modelling* merupakan objektif yang dilakukan dengan menyamarkan bagian dari gambar secara acak dan melatih *encoder* untuk merekonstruksi bagian yang disamarkan tersebut. Berbeda dengan *Masked Image Modelling*, *Masked Language Modelling* melakukan penyamaran terhadap token pada kalimat yang kemudian perlu direkonstruksi. Kemudian, *Masked Cross-Modal Modelling* merupakan gabungan dari *Masked Image & Masked Language* yang merekonstruksi bagian gambar dan teks yang hilang. Kemudian, *Image-to-Text Generation* merupakan objektif yang membuat model memprediksi teks secara *autoregressive* berdasarkan masukan gambar.

Alignment objective merupakan objektif yang difokuskan untuk melakukan *alignment* terhadap pasangan gambar-teks. Beberapa contohnya, yaitu *Image-Text Matching* dan *Region-Word Matching*. *Image-Text Matching* merupakan objektif yang memodelkan korelasi global antara gambar dan teks menjadi sebuah skor yang mengukur probabilitas *alignment* antara gambar dan teks. Kemudian, *Region-Word Matching* merupakan objektif yang bertujuan memodelkan korelasi *cross-modal* antara *image region* dan *word token*.

II.3 Image Captioning Dataset

Dataset yang digunakan pada *image captioning* merupakan sebuah *dataset* yang terdiri atas pasangan gambar dan teks. Teks yang terdapat pada *dataset* merupakan *caption* dari gambar yang terkait. Berdasarkan hasil studi literatur yang dilakukan, *dataset* untuk *task image captioning* Bahasa Indonesia umumnya menggunakan *dataset* Bahasa Inggris yang diterjemahkan secara otomatis menggunakan bantuan sistem penerjemah ke Bahasa Indonesia. Beberapa *dataset* yang dapat digunakan sebagai *dataset image captioning* yaitu dataset SBU1M, Flickr8k, Flickr30k, MSCOCO, dan Conceptual Captions (CC).

SBU1M (Ordonez dkk., 2011) merupakan kumpulan data gambar dan teks yang dikumpulkan secara otomatis dari situs web *Flickr*. *Dataset* ini berisi 1,000,000 *url* gambar *flickr* beserta keterangan yang sesuai. *Dataset* ini merupakan *dataset* yang relatif lama dan jarang digunakan dalam beberapa tahun terakhir.

Flickr8k (Hodosh dkk., 2013) merupakan sebuah *dataset* yang juga didapatkan melalui situs web *Flickr*. Namun, tidak seperti cara pengumpulan otomatis pada SBU1M, gambar di Flickr8k dipilih melalui *query* untuk suatu objek dan aksi tertentu menggunakan platform Amazon Mechanical Turk (AMT). *Dataset* ini berisi sejumlah 8,000 gambar. Gambar tersebut kemudian diberi keterangan oleh anotator di AMT, sehingga setiap gambar berisi lima keterangan yang dibuat secara independen.

Flickr30k (Young dkk., 2014) merupakan versi lebih besar dari kumpulan data Flickr8k. *Dataset* ini berisi 31,783 gambar yang berasal dari situs web *Flickr* dan 158,915 *captions*. Pembuatan *dataset* ini mengikuti strategi yang sama dengan Flickr8k, yang mana gambar dikumpulkan dari situs *web* Flickr dan teks yang diperoleh melalui *crowdsourcing* dari anotator menggunakan platform AMT. Setiap satu gambar kemudian diberikan *caption* oleh 5 orang anotator yang berbeda.

MSCOCO (Lin dkk., 2014) merupakan *dataset* berskala besar yang banyak digunakan untuk *task image captioning*. *Dataset* ini berisi gambar yang

dikumpulkan dari Flickr dan kemudian diberikan deskripsi oleh annotator melalui Platform AMT.

CC3M (Sharma dkk., 2018) merupakan *dataset* berskala web yang berisi lebih dari 3,300,000 pasangan gambar dan teks Bahasa Inggris. *Dataset* ini menggunakan gambar dan keterangan yang diambil secara otomatis dari bagian *alt text* pada halaman *web* HTML. Pada *dataset* CC3M, setiap gambar hanya memiliki 1 keterangan dan *dataset* ini belum melewati pengecekan seorang annotator.

II.4 Evaluasi Kinerja

Evaluasi kinerja dari model *image captioning* yang telah dibuat, biasanya diukur dengan menggunakan metrik-metrik evaluasi yang ada. Terdapat beberapa jenis metrik yang dapat digunakan untuk mengevaluasi kinerja dari model.

II.4.1 Automated Metrics

Automated metrics merupakan metrik-metrik pengukuran yang dilakukan menggunakan *script* otomatis untuk menghitung nilai kinerja dari model. Metrik-metrik ini mengevaluasi teks yang dihasilkan dari model dengan membandingkannya terhadap teks referensi. Contoh perbandingan yang dilakukan seperti menghitung skor kemiripan menggunakan statistik n-gram sederhana dan tumpang tindih kata. Beberapa contoh dari *automated metric* yang digunakan pada *image captioning* adalah BLEU, METEOR, ROUGE, CIDEr, dan SPICE.

BLEU merupakan sebuah metrik yang dikembangkan untuk *machine translation task*. Metrik BLEU membandingkan kalimat yang dihasilkan oleh model (kalimat kandidat) dengan kalimat *ground truth* (kalimat referensi) yang dibuat oleh manusia. BLEU menghitung tumpang tindih antara prediksi unigram (BLEU-1) atau umumnya n-gram (BLEU-2, BLEU-3, dan BLEU-4) dengan himpunan kandidat dari kalimat referensi. Untuk mendapatkan nilai BLEU yang tinggi, hasil kalimat perlu memiliki kata dan urutan kata yang sama dengan kalimat *ground truth*. Nilai BLEU maksimum yang dapat dicapai adalah 1 (atau kadang-kadang setara dengan

100), yang diperoleh ketika sebuah kalimat hasil sama persis dengan kalimat referensi. Nilai BLEU dapat dihitung menggunakan rumus II.1.

$$BLEU = BP \cdot \exp \left(\sum_{n=1}^N w_n \log p_n \right) \quad (II.1)$$

Pada rumus II.1, N merupakan jumlah dari n -gram. w_n merupakan bobot dari sebuah n -gram. p_n merupakan *precision* dari n -gram yang mengukur persentase n -gram pada kalimat kandidat yang juga terdapat pada kalimat referensi. Kemudian, BP merupakan *brevity penalty* yang memberikan poin pengurangan untuk kalimat yang terlalu pendek dan dapat dihitung menggunakan rumus II.2.

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{(1-\frac{r}{c})} & \text{if } c \leq r \end{cases} \quad (II.2)$$

Pada rumus II.2, c merupakan panjang kalimat kandidat dan r merupakan panjang kalimat referensi.

METEOR (Banerjee & Lavie, 2005) merupakan metrik yang dibuat untuk mengatasi beberapa masalah BLEU seperti kebutuhan untuk pencocokan kata yang tepat. Untuk mengatasi kekurangan tersebut, metrik METEOR melakukan pencocokan secara semantik dengan memanfaatkan *WordNet* untuk mencocokkan kata-kata di berbagai tingkatan, menggunakan pencocokan sinonim dan parafrase. Skor METEOR kemudian dihitung menggunakan penyesuaian antara output yang dihasilkan mesin dan kalimat referensi *ground truth* yang sesuai. Pada awalnya, himpunan unigram dari kalimat yang dihasilkan dan kalimat referensi digunakan untuk menghitung penyesuaian. Jika banyak opsi yang tersedia saat penyesuaian antara kalimat yang dihasilkan dan referensi, pengaturan dengan kemiripan paling sedikit kemudian dipilih. Nilai METEOR kemudian dihitung setelah proses penyesuaian tersebut. Nilai METEOR dapat dihitung menggunakan rumus II.3.

$$METEOR = Fmean * (1 - Penalty) \quad (II.3)$$

Pada rumus II.3, $Fmean$ merupakan kombinasi dari *precision* dan *recall* yang dihitung menggunakan *harmonic-mean*. Rumus $Fmean$ dapat dilihat lebih lanjut

pada rumus II.4. Pada rumus II.4, P merupakan *precision* yang dihitung dengan jumlah unigram yang beririsan pada kalimat kandidat dan kalimat referensi dibagi dengan jumlah unigram pada kalimat kandidat. Kemudian, R merupakan *recall* yang dihitung dengan jumlah unigram yang beririsan pada kalimat kandidat dan kalimat referensi dibagi dengan jumlah unigram pada kalimat referensi.

$$F_{mean} = \frac{10PR}{R + 9P} \quad (II.4)$$

Kemudian, *Penalty* merupakan faktor pengurangan poin saat memiliki nilai *precision* dan *recall* yang rendah. *Penalty* dapat dihitung dengan rumus II.5.

$$Penalty = 0.5 \times \left(\frac{c}{um}\right)^3 \quad (II.5)$$

Pada rumus II.5. c merupakan jumlah *chunks* dan um merupakan jumlah unigram yang sesuai antara kalimat kandidat dan kalimat referensi. *Chunks* merupakan jumlah kumpulan kata yang berurutan pada kalimat kandidat yang sesuai dengan kalimat referensi. Misalkan jika terdapat kalimat kandidat “presiden sedang berbicara kepada masyarakat” dan kalimat referensi “presiden sedang berbicara di depan umum”, maka terdapat tiga jumlah *chunks*, yaitu “presiden sedang berbicara”.

ROUGE (Lin, 2004) merupakan metrik yang dirancang untuk mengevaluasi rangkuman teks. Berbeda dengan BLEU yang menghitung presisi n-gram, ROUGE menghitung skor *recall* dari kalimat yang dihasilkan sesuai dengan kalimat referensi. Varian ROUGE yang paling banyak digunakan adalah ROUGE-L, yang didasarkan pada *subsequence* umum terpanjang. Varian lainnya termasuk ROUGE-W (*Subsequence* Umum Terpanjang dengan bobot) dan ROUGE-S (*Skip-Bigram Co-Occurences Statistics*). Salah satu keunggulan ROUGE-L dibandingkan BLEU dan METEOR adalah bahwa metrik ROUGE memeriksa *subsequences* dalam sebuah kalimat. Nilai ROUGE-L dapat dihitung dengan rumus II.6.

$$ROUGE - L = \frac{LCS(C, R)}{L(C, R)} \quad (II.6)$$

Pada rumus II.6, $LCS(C,R)$ merupakan jumlah *subsequence* umum terpanjang yang terdapat pada kalimat kandidat dan kalimat referensi. Kemudian, $L(C,R)$ merupakan rata-rata panjang dari kalimat kandidat dan kalimat referensi.

CIDEr (Vedantam dkk., 2015) menghitung konsensus antara kalimat yang dihasilkan dan satu set referensi kalimat dengan melakukan teknik pemangkasan bahasa yang berbeda, seperti *stemming* dan membangun satu set *n-gram*. *N-gram* yang sering berada di kalimat referensi dari semua data diberi bobot lebih rendah, karena dianggap kurang informatif terhadap konten visual yang ada dan bias terhadap konten tekstual dari kalimat. Berat untuk setiap *n-gram* dihitung menggunakan *Term Frequency* (TF) - *Inverse Document Frequency* (IDF) (TF-IDF), yang mana TF memberi bobot lebih tinggi pada *n-gram* yang lebih sering muncul di kalimat referensi, sedangkan IDF memberikan bobot yang lebih rendah pada *n-gram* yang umumnya muncul di seluruh *dataset*. Bobot TF-IDF dihitung dengan menggunakan rumus II.7.

$$g_k(s_{ij}) = \frac{h_k(s_{ij})}{\sum_{\omega_l \in \Omega} h_l(s_{ij})} \log \left(\frac{|I|}{\sum_{l_p \in I} \min(1, \sum_q h_k(s_{pq}))} \right) \quad (II.7)$$

Pada rumus II.7, jumlah kemunculan *n-gram* ω_k pada sebuah kalimat referensi s_{ij} dituliskan sebagai $h_k(s_{ij})$ ataupun $h_k(c_i)$ untuk kalimat kandidat. Kemudian, Ω merupakan kosakata dari keseluruhan *n-gram* dan I merupakan keseluruhan gambar pada *dataset*. Pada rumus II.7, bagian awal menghitung TF untuk setiap *n-gram* ω_k , dan bagian kedua menghitung IDF dengan menghitung nilai logaritmik dari jumlah gambar pada *dataset* dibagi dengan jumlah gambar yang mana ω_k muncul di salah satu kalimat referensi.

Kemudian, nilai CIDEr dapat dihitung dengan menggunakan rata-rata *cosine similarity* antara kalimat kandidat dan kalimat referensi yang dapat dilihat lebih lanjut pada rumus II.8 & II.9.

$$CIDEr(c_i, S_i) = \sum_{n=1}^N w_n CIDEr_n(c_i, S_i) \quad (II.8)$$

$$CIDEr_n(c_i, S_i) = \frac{1}{m} \sum_j \frac{g^n(c_i) \times g^n(s_{ij})}{||g^n(c_i)|| \times ||g^n(s_{ij})||} \quad (II.9)$$

Rumus II.8 menghitung nilai CIDEr pada jumlah *n-gram* ke N dengan menggunakan $w_n = 1/N$. Kemudian, pada rumus II.9 dihitung nilai CIDEr pada panjang n tertentu, yang mana $g^n(c_i)$ merupakan vektor yang terbentuk dari $g_k(c_i)$ pada seluruh *n-gram* sepanjang n , begitu juga dengan $g^n(s_{ij})$.

Untuk menghapus ketidakcocokan antara *human evaluation* dan skor CIDEr, digunakan varian dari CIDEr, yaitu CIDEr-D yang menambahkan variasi dari kata. Penambahan variasi kata dilakukan dengan tidak melakukan stemming dan memastikan kata-kata dengan tingkat keyakinan yang tinggi tidak diulang dalam kalimat dengan menggunakan *Gaussian Error* pada perbedaan panjang antara kalimat hasil dan kalimat referensi.

SPICE (Anderson dkk., 2016) merupakan metrik yang mengukur kesamaan antara graf kalimat yang dihasilkan dengan graf kalimat referensi yang dibuat manusia. Graf kemudian di-*encode* terhadap objek dan hubungannya dengan menggunakan *dependency parsing*. Hal tersebut membuat SPICE sangat bergantung pada metode *parsing* yang dilakukan. Mirip dengan METEOR, SPICE menggunakan *WordNet* untuk menemukan dan memperlakukan sinonim sebagai *positive matches* saat menghitung nilai F1 antara kalimat hasil dan kalimat referensi. Nilai SPICE dapat dihitung menggunakan rumus II.10.

$$SPICE(c, S) = \frac{2 \times P(c, S) \times R(c, S)}{P(c, S) + R(c, S)} \quad (II.10)$$

Pada rumus II.10, P merupakan *precision* yang dihitung dengan jumlah *tuple* kata yang beririsan pada kalimat kandidat c dan kalimat referensi S , dibagi dengan jumlah *tuple* kata pada kalimat kandidat. Kemudian, R merupakan *recall* yang dihitung dengan jumlah *tuple* kata yang beririsan pada kalimat kandidat c dan kalimat referensi S , dibagi dengan jumlah *tuple* kata pada kalimat referensi.

II.4.2 Human Evaluation

Human evaluation merupakan evaluasi yang dilakukan oleh sekumpulan orang dengan mengevaluasi kualitas teks yang dihasilkan. *Human evaluation* dirancang karena kurangnya korelasi antara kalimat yang dihasilkan oleh model dengan kalimat referensi dari manusia, sehingga memerlukan evaluasi yang dilakukan oleh manusia untuk menilai hasil kalimat. Untuk melakukan *human evaluation*, orang atau *evaluator* yang menilai hasil kalimat diberikan beberapa petunjuk dan pertanyaan terkait kalimat yang dihasilkan oleh model.

Salah satu metrik pada *human evaluation* yang dikembangkan pada *image captioning* yaitu THUMB (Kasai dkk., 2021). Pada metrik ini digunakan dua buah pengukuran nilai yaitu menggunakan *precision* dan *recall*. *Precision* digunakan mengukur seberapa tepat kalimat hasil dalam mendeskripsikan objek yang terdapat pada gambar. Kemudian, nilai *recall* merupakan nilai yang mengukur seberapa informatif kalimat yang dihasilkan. Kedua nilai ini memiliki skala maksimal 5 dan rata-rata dari kedua nilai kemudian dikurangi dengan beberapa nilai penalti yang diberikan terhadap kalimat. Beberapa poin penalti yang dipertimbangkan adalah terkait *Fluency* atau ketepatan ejaan dan tata bahasa kalimat, *Conciseness* atau keringkasan dari kalimat, serta *Inclusive Language* atau kesubjektivan dari kalimat yang dihasilkan.

II.5 Perplexity

Perplexity merupakan salah satu metrik yang dapat digunakan untuk mengevaluasi sebuah *language model*. *Perplexity* mengukur seberapa baik sebuah distribusi peluang atau distribusi model dalam memprediksi sebuah sampel teks (Shi dkk., 2022). *Perplexity* didefinisikan sebagai nilai eksponensial dari rata-rata negatif logaritmik dari probabilitas sebuah urutan token, dengan eksponen berbasis e . Jika kita memiliki sebuah urutan token $X = (x_0, x_1, \dots, x_t)$, maka *perplexity* dapat dihitung menggunakan rumus II.11.

$$PPL(X) = \exp \left\{ -\frac{1}{t} \sum_i^t \log p_\theta(x_i|x_{<i}) \right\} \quad (II.11)$$

Pada rumus II.11, $\log p_\theta(x_i|x_{<i})$ merupakan probabilitas logaritmik sebuah model menghasilkan token ke- i berdasarkan token-token sebelumnya ($x_{<i}$), dengan θ menggambarkan parameter dari sebuah *language model*. Secara intuitif, *perplexity* dapat dilihat sebagai evaluasi kemampuan model dalam memprediksi secara *uniform*, berdasarkan kumpulan token dalam sebuah korpus.

II.6 XGLM (Lin dkk., 2021)

XGLM merupakan sebuah *multilingual generative language model* yang dilatih dengan menggunakan bahasa selain Inggris. Pembuatan XGLM didasari oleh *large autoregressive language model*, seperti GPT-3, yang hanya menggunakan sedikit jumlah data non-Inggris, yaitu sekitar 7%. *Dataset* yang digunakan pada *pre-training* merupakan CC100-X, sebuah *dataset* multilingual yang berisi 68 gambar Common Crawl (CC) dan 134 bahasa. Dari *dataset* tersebut, model XGLM kemudian dilatih pada korpus yang berisi 500B token yang berasal dari 30 bahasa yang berbeda, salah satunya adalah Bahasa Indonesia. Pada proses *pre-training* dilakukan *upsampling* pada data yang termasuk dalam *low-resource language*. Seluruh bahasa diproses dengan *vocabulary* gabungan yang berisi 250k token yang diproses menggunakan *unigram language modelling* pada kakas *SentencePiece*.

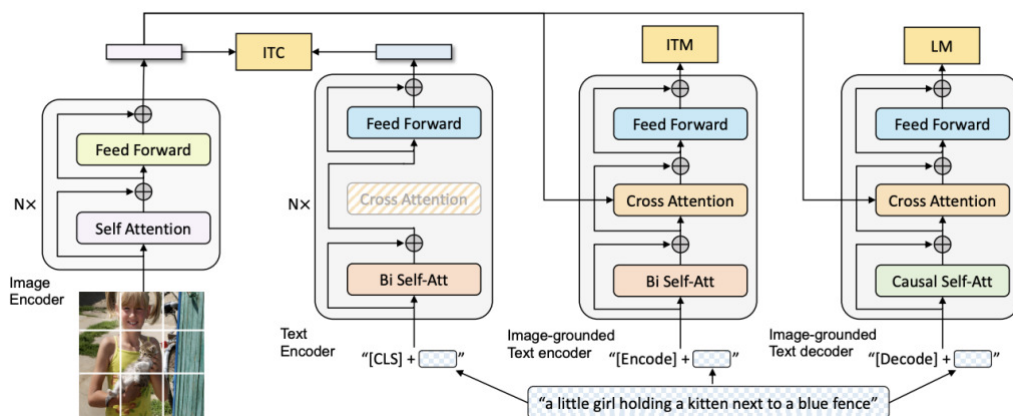
Model yang digunakan pada XGLM merupakan sebuah model *decoder-only causal language model* yang memanfaatkan arsitektur transformer, seperti yang digunakan pada model GPT-3. Terdapat 4 buah ukuran model pada XGLM yaitu 564M, 1.7B, 2.9B, dan 7.5B parameter. Setelah dilatih, model-model XGLM kemudian diukur kemampuannya dalam *multilingual in-context learning*, yaitu pada *zero-shot* dan *few-shot learning*. Hasilnya, model XGLM dengan parameter terbesar (7.5B) memiliki kemampuan *cross-lingual* yang baik dan mampu mencapai *state-of-the-art* untuk *few-shot learning* pada lebih dari 20 bahasa, termasuk bahasa yang *low-resource* dan *mid-resource*, pada permasalahan *common-sense reasoning*, *natural language inference*, dan *machine translation*.

II.7 Penelitian Terkait

Terdapat berbagai penelitian yang berkaitan dengan tugas akhir ini. Penelitian-penelitian tersebut meliputi penelitian model-model VL seperti BLIP (Li dkk., 2022), GIT (Wang dkk., 2022a), dan OFA (Wang dkk., 2022b). Kemudian, belum terdapat model VL yang dilatih khusus menggunakan Bahasa Indonesia ataupun memiliki versi *multilingual* yang dapat mengatasi Bahasa Indonesia. Salah satu model *image captioning* Bahasa Indonesia yang telah diteliti sebelumnya yaitu model *image captioning* Bahasa Indonesia menggunakan SCN dan *Soft Attention*.

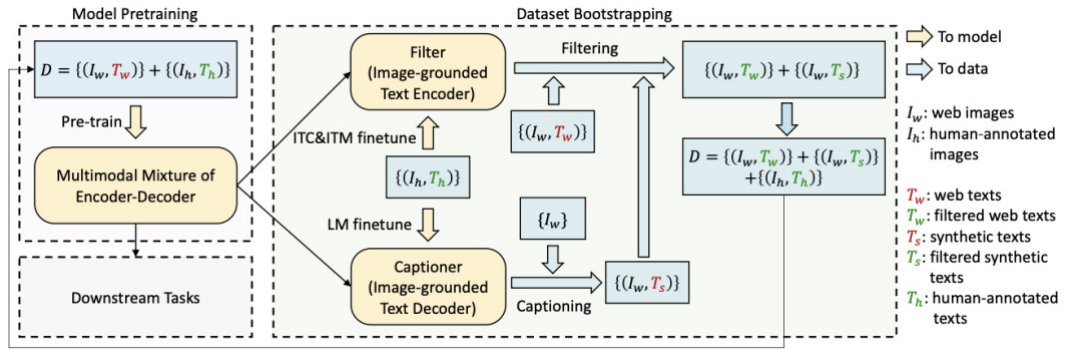
II.7.1 BLIP (Li dkk., 2022)

Bootstrapping Language-Image Pre-training (BLIP) merupakan sebuah model VL yang dirancang karena adanya permasalahan terkait model-model VL yang hanya bagus dalam satu *task* saja, hanya dalam *task* pemahaman ataupun hanya dalam *task* menghasilkan kalimat. Selain itu, model BLIP juga dibuat karena model-model VL yang ada dilatih menggunakan data dari *web* yang *noisy* dan kurang bagus untuk pembelajaran *vision-language*. Untuk mengatasi permasalahan tersebut, BLIP menggunakan *Multimodal mixture of Encoder-Decoder* (MED), sebuah komponen yang dapat digunakan sebagai *unimodal encoder*, *image-grounded text encoder*, ataupun *image-grounded text decoder*. Penggunaan komponen tersebut membuat BLIP mampu untuk mempelajari *multitask pretraining* dan *transfer learning* secara lebih efektif.



Gambar II.5. Arsitektur BLIP (Li dkk., 2022)

Dalam arsitekturnya, BLIP menggunakan ViT sebagai *image encoder*, yang membagi gambar masukan dan melakukan *encoding* terhadap gambar masukan. Kemudian, terdapat tiga MED yaitu *Text Encoder* yang meringkas kalimat, *Image-grounded Text Encoder* yang menerima informasi visual pada *cross attention layer* dan *output* dari *image-grounded text encoder* digunakan sebagai representasi pasangan gambar-teks, serta *Image-grounded text decoder* yang menggunakan *self-attention layers* untuk menghasilkan kalimat.



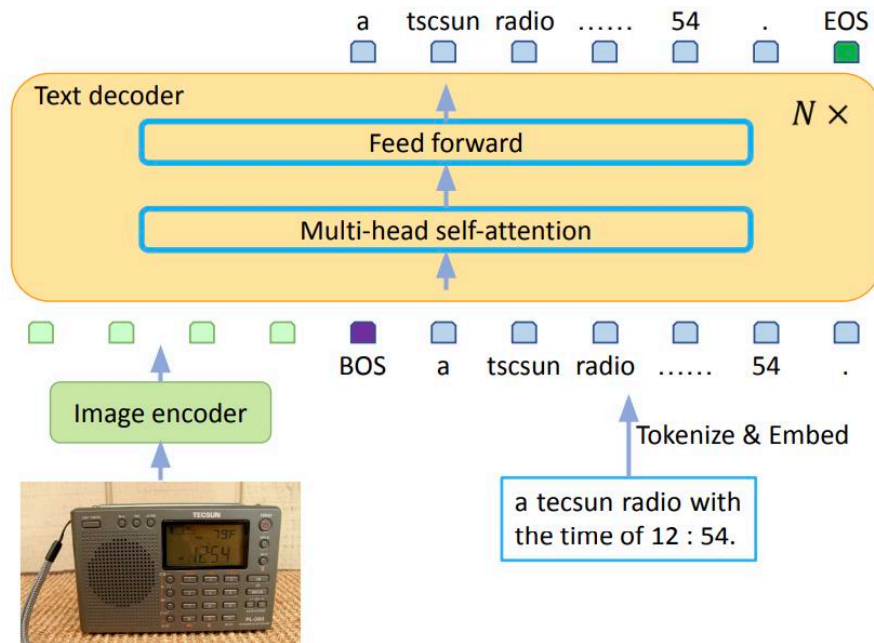
Gambar II.6. Pre-training BLIP (Li dkk., 2022)

Selain itu, BLIP juga dilatih menggunakan *Captioning and Filtering* (CapFilt) sebuah metode *dataset bootstrapping* yang dibuat untuk mempelajari pasangan gambar dan teks yang *noisy*. Komponen MED yang telah ada dilatih menjadi dua modul, yaitu *captioner* yang berfungsi untuk menghasilkan teks sintetik jika diberikan gambar, dan *filter* untuk menghapus *noise* pada teks referensi dan teks sintetik.

II.7.2 GIT (Wang dkk., 2022a)

Generative Image-to-text Transformer (GIT) merupakan sebuah model VL yang dirancang untuk memiliki arsitektur yang sederhana namun efektif untuk *downstream task* pada *vision-language*, seperti *image/video captioning* dan

question answering. Arsitektur pada GIT hanya terdiri dari satu buah *image encoder* dan satu buah *text decoder*.



Gambar II.7. Arsitektur GIT (Wang dkk., 2022a)

Image encoder yang digunakan merupakan *image encoder* yang telah di *pretrain* dengan *contrastive objective*. Masukan *image* diproses oleh *image encoder* menjadi sebuah fitur yang kemudian di *flatten* menjadi sebuah *list of features*. Fitur gambar kemudian diproyeksikan pada dimensi D . *Text decoder* yang digunakan merupakan sebuah *transformer* yang terdiri dari beberapa blok dan setiap blok terdiri dari satu *self-attention layer* dan *feed-forward layer*. Teks kemudian di tokenisasi dan di *embed* pada dimensi D . Fitur hasil *embedding* gambar dan teks kemudian digabungkan menjadi masukan dari *text decoder* dan ditambahkan token [BOS] untuk memisahkan gambar dan teks. Kemudian *text decoder* menggunakan representasi gabungan tersebut untuk melakukan *decode* secara *auto regressive* hingga mencapai token [EOS] atau mencapai batas kalimat.

Pada proses *pre-training*, untuk setiap pasangan gambar-teks, digunakan sebuah *language modelling loss* seperti yang tertera pada rumus II.12.

$$l = \frac{1}{N+1} \sum_{i=1}^{N+1} CE(y_i, p(y_i | I, \{y_j, j = 0, \dots, i-1\})), \quad (II.12)$$

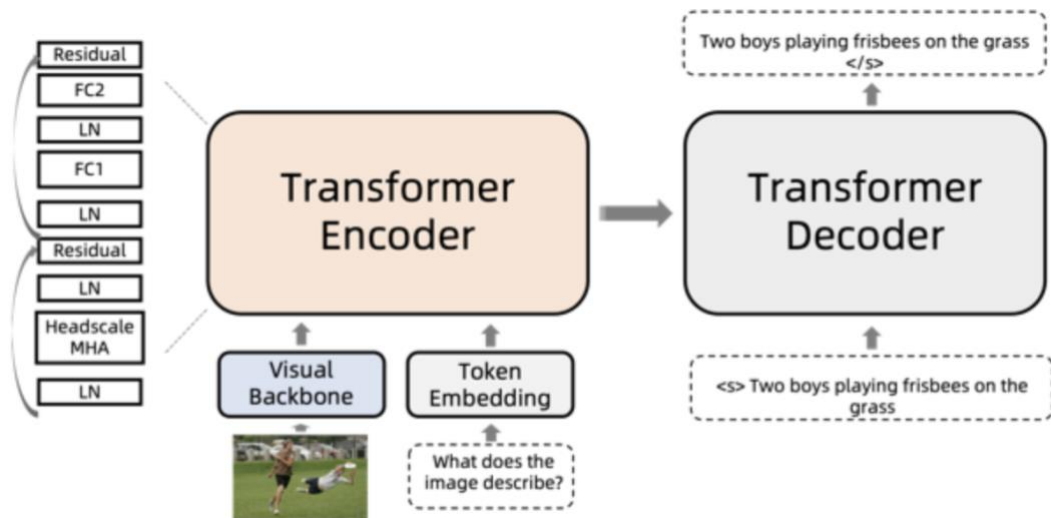
Pada rumus II.12, I merupakan gambar, $y_i, i \in \{1, \dots, N\}$ merupakan token teks dengan y_0 adalah [BOS] dan y_{N+1} adalah [EOS], dan CE merupakan *cross entropy* dengan pemerataan label 0.1. Pada *language modelling loss* yang digunakan, dihitung rata-rata *cross entropy* dari kalimat referensi atau *ground truth* dengan kalimat kandidat yang merupakan $p(y_i | I, \{y_j, j = 0, \dots, i-1\})$ pada rumus II.12.

II.7.3 OFA (Wang dkk., 2022b)

One For All (OFA) merupakan *framework Seq2seq* yang dirancang untuk membuat model *omnipotent* yang memiliki tiga karakteristik, yaitu 1. *Task-Agnostic* (TA); representasi yang mendukung berbagai jenis tugas baik pada *pretraining* atau *finetuning*. 2. *Modality-Agnostic* (MA); mendukung *input* dan *output* yang sama pada tiap *task* untuk menangani *modality* yang berbeda. dan 3. *Task comprehensiveness* (TC); memiliki tugas yang cukup bervariasi untuk memiliki kemampuan generalisasi yang kuat.

Dalam memproses data, OFA menggunakan *ResNet* yang telah di *pre-trained* untuk mengubah gambar menjadi *patch features* berukuran *hidden size*. Kemudian, untuk memproses bagian teks, OFA menggunakan tokenisasi *byte-pair encoding* (BPE) untuk mengubah *sequence* teks menjadi *subword sequence*. Kemudian dilakukan *embedding* dari hasil tokenisasi tersebut menjadi fitur. Untuk memproses *modality* yang berbeda tanpa spesifik terhadap *task* tertentu, OFA merepresentasikan data dalam sebuah ruang representasi yang universal serta melakukan diskritisasi terhadap teks, gambar, dan objek kemudian merepresentasikannya dalam token di sebuah *vocabulary* yang universal. Pada representasi gambar, OFA juga menggabungkan area *bounding box* serta label dari objek ke dalam sebuah

vocabulary yang sama. *Bounding box* didiskritisasi menjadi sebuah koordinat integer dan label diproses dengan menggunakan tokenisasi BPE.



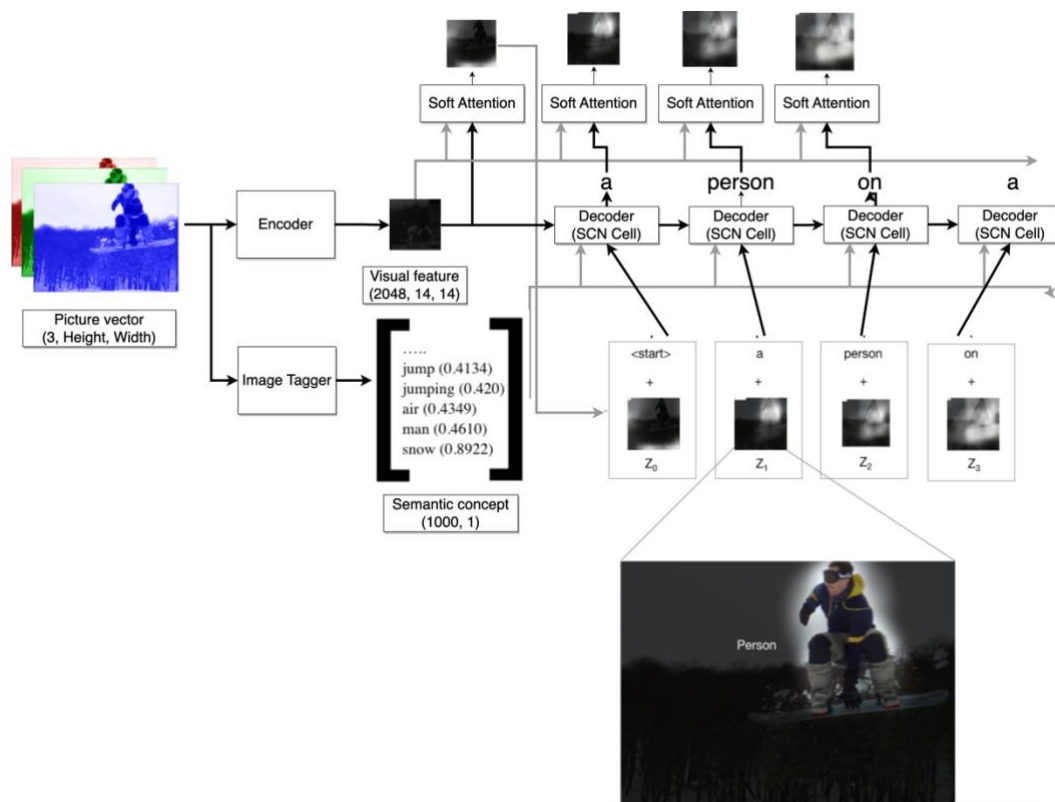
Gambar II.8. Arsitektur OFA (Wang dkk., 2022b)

Arsitektur dari OFA menggunakan *transformer* dalam sebuah *framework encoder-decoder*. *Encoder* pada *transformer* terdiri atas *self-attention* dan *feed forward*, sedangkan pada *decoder* terdapat tambahan *layer* berupa *cross-attention* untuk menghubungkan *decoder* dengan representasi *output* dari *encoder*. Untuk menstabilkan proses *training* dan konvergensi, pada OFA ditambahkan *head scaling* pada *self-attention*, sebuah *post-attention layer normalization*, dan sebuah *layer normalization* setelah *feed forward*. Kemudian, untuk informasi terkait posisi, digunakan *absolute position embedding* pada teks dan gambar yang terpisah dari *embedding* terhadap gambar dan token. Selain itu, digunakan juga posisi relatif pada teks dan gambar.

II.7.4 SCN & Soft Attention Indo (Sinurat, 2019)

Pada kasus *image captioning* bahasa indonesia, Sinurat (2019) menggunakan pendekatan *Semantic Compositional Network* (SCN) yang diusulkan oleh Gan dkk.

(2017). Pada pendekatan SCN dibutuhkan faktor visual lainnya seperti *tags* objek-objek pada gambar yang juga berfungsi sebagai masukan pada model *image captioning*. Faktor visual *tag* pada penelitian ini diperoleh dengan menggunakan ResNet yang dikembangkan oleh He dkk., (2016). Hasil dari bagian visual ini merupakan bobot yang didapat dari *ResNet* serta probabilitas setiap *tag* yang terdeteksi. Faktor visual tersebut kemudian menjadi masukan bagi *Long-Short Term Memory* (LSTM). Setiap *semantic tag* yang dideteksi pada gambar kemudian digunakan untuk mengubah matriks bobot pada LSTM konvensional menjadi LSTM yang *ensemble* dan matriks bobotnya juga mempertimbangkan probabilitas *semantic tag* yang terdeteksi.



Gambar II.9. Arsitektur SCN & Soft Attention Indo (Sinurat, 2019)

Model SCN tersebut kemudian dimodifikasi dengan ditambahkan komponen *Soft Attention* yang dikemukakan oleh Xu dkk., (2015). Mekanisme *soft attention* digunakan karena kelebihanannya yang *smooth* dan *differentiable* pada lingkungan atensi yang deterministik.

BAB III

ANALISIS MASALAH DAN RANCANGAN SOLUSI

Pada bagian ini dijelaskan lebih lanjut mengenai hasil analisis terhadap persoalan *Image Captioning* Bahasa Indonesia dengan menggunakan *Vision-Language Model*, beberapa analisis terkait alternatif solusi yang dapat dilakukan untuk menyelesaikan persoalan, serta deskripsi rancangan solusi yang dilakukan untuk tugas akhir ini.

III.1 Analisis Persoalan

Berdasarkan rumusan masalah yang telah dipaparkan, terdapat beberapa penelitian terdahulu terkait *image captioning* berbahasa Indonesia. Penelitian-penelitian *image captioning* Bahasa Indonesia terdahulu dapat dilihat lebih lanjut pada tabel III.1.

Tabel III.1. Perbandingan Penelitian *Image Captioning* Bahasa Indonesia

Penelitian	Pendekatan Penelitian	Dataset	Jml Data
Nugraha & Arifianto, 2019	Menggunakan model Inception-V3 sebagai <i>encoder</i> dan menggunakan Gated Recurrent Unit (GRU) sebagai <i>decoder</i> .	Flickr30k yang diterjemahkan ke Bahasa Indonesia dengan <i>Google Translate</i> . Terdapat proses perbaikan data & penambahan data dengan <i>crowdsourcing</i> .	158,915
Sinurat, 2019	Menggunakan model ResNet yang dimodifikasi dengan <i>soft attention</i> sebagai <i>encoder</i> serta SCN sebagai <i>decoder</i> .	Menggunakan <i>dataset</i> COCO, Flickr30k, dan Flickr8k yang diterjemahkan ke Bahasa Indonesia dan dilakukan proses perbaikan terhadap 2% data (~12,000).	609.183
Mulyanto dkk., 2019	Menggunakan model CNN sebagai <i>encoder</i> dan menggunakan LSTM sebagai <i>decoder</i> .	Membuat <i>dataset</i> Bahasa Indonesia bernama FEEH-ID (tidak <i>open-source</i>). Gambar diambil dari Flickr dan <i>caption</i> menggunakan gabungan <i>Google Translate</i> dan buatan manusia.	40,495

Tabel III.1. Lanjutan

Penelitian	Pendekatan Penelitian	Dataset	Jml Data
Mulyawan dkk., 2021	Menggunakan model ResNet sebagai <i>encoder</i> dan mekanisme <i>self-attention</i> pada <i>decoder</i> .	Flickr30k yang diterjemahkan ke Bahasa Indonesia dengan <i>Google Translate</i> . Terdapat proses perbaikan data.	40,460

Dari segi penggunaan model, penelitian *image captioning* bahasa Indonesia yang telah dilakukan umumnya dilakukan dengan menggunakan *pre-trained* yang terpisah. Namun demikian, hasil-hasil *image captioning* yang saat ini menjadi *state-of-the-art* pada umumnya menggunakan model-model VL. Penggunaan model-model VL mampu memperoleh kinerja yang lebih baik karena model VL telah melewati proses *pre-training* pada kumpulan gambar dan teks berskala besar. Adanya *pre-training* pada data berskala besar tersebut membuat sebuah model VL mampu mempelajari representasi yang universal antar kedua modalitasnya (Du dkk., 2022). Representasi universal tersebut bermanfaat untuk mencapai kinerja yang kuat dalam *downstream task vision* dan *language*, salah satunya yaitu *image captioning* (Du dkk., 2022).

Dari segi penggunaan *dataset*, penelitian *image captioning* bahasa Indonesia yang ada dilakukan dengan *dataset* yang memiliki kualitas yang kurang baik. Penelitian *image captioning* bahasa Indonesia yang sudah ada umumnya dilakukan dengan menggunakan *dataset* Bahasa Inggris yang diterjemahkan ke dalam Bahasa Indonesia. Proses tersebut dilakukan dengan menerjemahkan secara otomatis ke dalam Bahasa Indonesia menggunakan bantuan sistem penerjemah. Setelah proses penerjemahan tersebut, dilakukan upaya perbaikan terhadap kalimat terjemahan. Namun demikian, jumlah perbaikan yang dilakukan masih minimal, sehingga masih terdapat keterbatasan terkait *dataset* Bahasa Indonesia yang memiliki kualitas baik. Proses perbaikan perlu untuk dilakukan karena terkadang sistem penerjemah tidak menerjemahkan bagian penting dari sebuah teks, yang kemudian dapat menyebabkan masalah, seperti adanya kemungkinan mengurangi makna dari kalimat serta adanya *noise* pada data (Dashtipour dkk., 2016).



caption awal:

The sink is on the island of a large kitchen.

hasil terjemahan:

Wastafel ada di pulau dapur besar.



caption yang kurang tepat:

tree are two woman standing in the rain under a pink umbrella.

hasil terjemahan ketika

menggunakan *caption* yang kurang tepat:

pohon adalah dua wanita berdiri di tengah hujan di bawah payung merah muda.

Gambar III.1. Contoh Hasil Terjemahan *Dataset*

Salah satu contoh kesalahan yang terdapat pada terjemahan dapat dilihat pada Gambar III.1. Kedua gambar tersebut merupakan dua buah contoh data MSCOCO yang diterjemahkan menggunakan sistem penerjemah *google translate*. Dari Gambar III.1, dapat dilihat bahwa terdapat hasil dari terjemahan yang kemudian memiliki arti yang berbeda setelah diterjemahkan secara otomatis. Pada gambar atas, penggunaan kata “*island*” diartikan secara otomatis menjadi kata “pulau”, yang mana penggunaan kata “pulau” kemudian membuat adanya makna yang berkurang dari deskripsi awal. Kata “*island*” pada konteks kalimat tersebut lebih cocok untuk diartikan sebagai “tempat di tengah-tengah jalan”.

Kemudian, kesalahan terjemahan lain yang mungkin terjadi terlihat seperti pada Gambar III.1 bagian bawah. Pada *caption* awal terdapat *noise* pada data berupa kesalahan pengetikan kata “*there*” menjadi “*tree*”. Hal ini mengakibatkan ketika diterjemahkan secara otomatis menggunakan sistem penerjemah, maka muncul kata “pohon” padahal pada gambar tidak terdapat objek pohon. Selain itu, kesalahan kata tersebut juga membuat kalimat hasil terjemahan kurang sesuai dan memiliki semantik yang kurang sesuai dengan tata Bahasa Indonesia.

III.2 Analisis Solusi

Berdasarkan masalah dan studi literatur yang telah dipaparkan pada bagian sebelumnya, selanjutnya dilakukan analisis terhadap solusi-solusi yang memungkinkan. Alternatif solusi yang kemudian dianalisis meliputi pemilihan *dataset*, dan pemilihan metrik evaluasi.

Berdasarkan *dataset* yang digunakan pada *image captioning*, belum terdapat *image captioning dataset* Bahasa Indonesia yang tersedia secara umum untuk digunakan. Untuk mengatasi permasalahan tersebut terdapat dua alternatif solusi yang dapat dilakukan. Alternatif solusi pertama adalah dengan merancang sebuah *dataset* dari awal dengan mengumpulkan gambar dan memberikan keterangan terhadap gambar-gambar tersebut. Namun demikian, alternatif solusi ini memakan sumber daya yang mahal, baik dari segi waktu maupun biaya. Oleh karena itu, solusi ini kurang sesuai untuk dapat dilakukan saat ini. Alternatif solusi kedua adalah mengikuti alternatif solusi yang telah dilakukan pada penelitian-penelitian sebelumnya, yaitu dengan menggunakan *dataset* berbahasa Inggris yang diterjemahkan ke Bahasa Indonesia. Pada alternatif solusi ini, sumber daya yang perlu dikeluarkan lebih sedikit dibandingkan dengan alternatif solusi pertama, terlebih apabila proses terjemahan dilakukan dengan menggunakan sistem penerjemah. Namun demikian, untuk memastikan *dataset* yang digunakan memiliki kualitas yang baik dan menghindari masalah terjemahan yang telah dibahas sebelumnya, maka perlu dilakukan perbaikan terhadap hasil terjemahan.

Tabel III.2. Perbandingan Karakteristik *Dataset Image Captioning*

Nama Dataset	Jml Gambar	Jml Capt (per Gambar)	Jml Pasangan Gambar-Teks	Dianotasi Manusia
SBU1M	1,000,000	1	1,000,000	X
Flickr8k	8,000	5	40,000	√
Flickr30k	31,014	5	155,070	√
MSCOCO	123,287	5	616,435	√
CC3M	3,356,703	1	3,356,703	X

Kemudian, dari beberapa *dataset* yang dapat digunakan, perbandingan terkait *dataset* tersebut dapat dilihat pada tabel III.2. Untuk meminimalisir terjadinya kesalahan terjemahan akibat kalimat awal yang kurang baik, perlu dipastikan *dataset* yang digunakan memiliki kualitas awal yang baik. Oleh karena itu, alternatif *dataset* yang dipilih untuk digunakan adalah *dataset* yang telah dianotasi oleh manusia. Berdasarkan kriteria tersebut, *dataset* SBU1M dan CC3M kurang sesuai untuk digunakan pada penelitian tugas akhir ini. Kemudian, dari *dataset* yang tersisa, *dataset-dataset* tersebut merupakan *dataset* yang sering digunakan pada *task image captioning*. Namun demikian, *dataset* MSCOCO lebih banyak digunakan karena *dataset* MSCOCO difokuskan untuk menentukan atribut-atribut pada objek dan pemandangan yang ada sehingga mampu lebih mendetailkan hubungan antar objek dan memberikan deskripsi semantik yang baik (Lin dkk., 2014). Selain itu, *dataset* MSCOCO juga memiliki jumlah data yang lebih banyak jika dibandingkan *dataset* lainnya yang juga dianotasi oleh manusia. Penggunaan *dataset* yang lebih banyak mampu membuat model mempelajari variasi yang lebih banyak pada data gambar dan-teks. Berdasarkan analisis tersebut, pada tugas akhir ini MSCOCO dipilih sebagai *dataset* yang digunakan.

Lalu, berdasarkan metrik-metrik yang telah dijelaskan pada bagian sebelumnya, dapat digunakan beberapa alternatif metrik untuk mengevaluasi hasil eksperimen dari *image captioning*. Perbandingan setiap metrik yang dapat digunakan pada *image captioning* dapat dilihat lebih lanjut pada tabel III.3.

Tabel III.3. Perbandingan Metrik *Image Captioning*

Metrik	Deskripsi	Kelebihan & Kekurangan
BLEU	Mengukur <i>precision</i> antara <i>caption</i> kandidat dan <i>caption</i> referensi berdasarkan kecocokan <i>n-gram</i> .	<p>Kelebihan: Mudah dihitung & dipahami. Semakin tinggi <i>n-gram</i> maka semakin sesuai <i>caption</i> kandidat dengan referensi.</p> <p>Kekurangan: Tidak mempertimbangkan kemiripan semantik atau urutan kata. Semakin tinggi <i>n-gram</i>, maka semakin sulit mendapatkan nilai yang tinggi.</p>
METEOR	Menggunakan <i>harmonic mean</i> untuk menyeimbangkan <i>precision</i> dan <i>recall</i> pada <i>n-gram</i> serta memberikan penalty untuk urutan kata yang berbeda.	<p>Kelebihan: Mempertimbangkan sinonim dan urutan kata.</p> <p>Kekurangan: Tidak sepenuhnya menangkap makna semantik.</p>
ROUGE	Mengukur <i>recall</i> antara <i>caption</i> kandidat dan <i>caption</i> referensi berdasarkan kecocokan <i>n-gram</i> .	<p>Kelebihan: Bermanfaat agar beberapa variasi teks diterima.</p> <p>Kekurangan: Tidak sepenuhnya menangkap makna semantik.</p>
CIDEr	Mempertimbangkan konsensus tiap <i>caption</i> dan <i>n-gram</i> yang lebih langka untuk mendorong hasil <i>caption</i> yang lebih beragam & informatif.	<p>Kelebihan: Mendorong keragaman dalam teks yang dihasilkan.</p> <p>Kekurangan: Belum sepenuhnya menangkap kesamaan semantik atau kualitas teks yang dihasilkan.</p>
SPICE	Menggunakan <i>scene graphs</i> untuk merepresentasikan konten pada gambar dan mengevaluasi seberapa baik hubungan antar objek dalam gambar.	<p>Keuntungan: Mirip dengan penilaian manusia, berfokus pada konten semantik teks.</p> <p>Kekurangan: Memerlukan <i>scene graphs</i>, yang mungkin tidak tersedia di semua kumpulan data.</p>

Tabel III.3. Lanjutan

Metrik	Deskripsi	Kelebihan & Kekurangan
<i>Human Evaluation</i>	Menggunakan annotator untuk menilai hasil kalimat melalui beberapa petunjuk dan pertanyaan terkait kalimat yang dihasilkan oleh model.	Keuntungan: Mampu memahami dengan baik semantik kalimat hasil. Kekurangan: Memakan banyak sumber daya, waktu, dan bergantung terhadap keragaman annotator.

Berdasarkan Tabel III.3, dapat dilihat bahwa secara umum metrik-metrik otomatis belum mampu untuk menangkap makna semantik dari suatu kalimat secara keseluruhan. Metrik SPICE, yang merupakan salah satu metrik otomatis yang dapat mengukur semantik dari kalimat hasil, sulit untuk dapat diterapkan. Hal tersebut karena adanya ketergantungan pada implementasi, seperti bergantung pada *scene graphs* yang terdapat dalam data. Oleh karena itu, untuk mengukur seberapa baik semantik dari kalimat hasil, digunakan sebuah *human evaluation*. Namun demikian, untuk mengatasi sumber daya waktu dan biaya yang besar, maka *human evaluation* hanya dilakukan pada sampel dari *dataset*.

Kemudian, metrik yang digunakan lainnya adalah metrik untuk mengukur kalimat hasil agar dapat memiliki tata bahasa yang benar. Metrik yang kemudian dipilih dari metrik-metrik yang ada adalah metrik BLEU yang banyak diterapkan pada *image captioning*. Banyaknya pemakaian dari BLEU didasari oleh implementasi dari BLEU yang relatif mudah untuk dihitung dan dipahami. Selain itu, berdasarkan jumlah *n-gram* yang digunakan terdapat juga variasi dari BLEU, seperti BLEU-1, BLEU-2, BLEU-3, dan BLEU-4. Penggunaan semua *n-gram* BLEU dalam metrik evaluasi membantu memberikan penilaian yang menyeluruh terhadap teks yang dihasilkan, yaitu dengan mempertimbangkan berbagai tingkat kompleksitas dan koherensi bahasa. Penggunaan keseluruhan *n-gram* BLEU merupakan pendekatan yang banyak digunakan dan dapat memberikan pemahaman yang lebih beragam jika dibandingkan dengan hanya menggunakan satu varian *n-gram*.

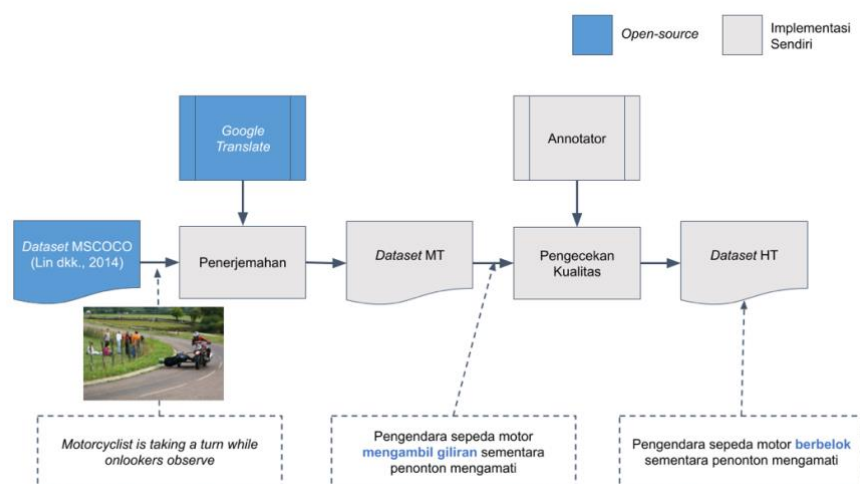
Selain itu, untuk menunjang hasil dari evaluasi metrik BLEU, digunakan juga metrik CIDEr untuk mendorong adanya keragaman dalam teks yang dihasilkan. Metrik CIDEr juga digunakan sebagai komplementer dari BLEU untuk mengatasi keterbatasan utama dari metrik BLEU, yaitu hanya mencocokkan kata per kata dan tidak mempertimbangkan keragaman.

III.3 Deskripsi Rancangan Solusi

Berdasarkan analisis permasalahan & analisis solusi yang dilakukan, solusi yang diterapkan untuk menjawab permasalahan yang telah dirumuskan adalah dengan membuat *image captioning dataset* Bahasa Indonesia dengan menerjemahkan *dataset* yang terdapat pada Bahasa Inggris ke dalam Bahasa Indonesia, serta melakukan eksperimen dengan model-model VL yang telah ditentukan. Langkah-langkah pada tiap proses tersebut dibahas lebih lanjut pada subbab berikut.

III.3.1 Pembuatan *Dataset*

Proses pembangunan *dataset* dilakukan dengan menerjemahkan 616,435 data MSCOCO Bahasa Inggris ke dalam Bahasa Indonesia. Proses pembuatan *dataset* memerlukan waktu yang cukup lama karena jumlah data yang tidak sedikit, serta adanya proses pengecekan oleh manusia. Alur dari proses pembangunan *dataset* dapat dilihat lebih lanjut pada Gambar III.2.



Gambar III.2. Alur Pembangunan *Dataset*

Pada gambar III.2, dapat dilihat bahwa tahap pertama pada pembuatan *dataset* dilakukan dengan menerjemahkan dataset MSCOCO yang berjumlah 616,435 data. Penerjemahan *dataset* ke Bahasa Indonesia dilakukan dengan menggunakan bantuan sistem penerjemah. Sistem penerjemah yang digunakan untuk menerjemahkan *dataset* pada tugas akhir ini adalah *Google Translate*. *Google translate* digunakan karena kemudahan terhadap aksesnya, tanpa perlu untuk menyiapkan model *machine translation*. Hasil dari proses tersebut merupakan sebuah *dataset* yang kemudian disebut sebagai *machine translated (MT) dataset*.

Kemudian, untuk memperkaya kualitas dan kuantitas *dataset* maka dilakukan pengecekan kualitas terhadap hasil terjemahan. Pada proses pengecekan kualitas dilakukan perbaikan terhadap terjemahan deskripsi yang tidak sesuai secara tata bahasa ataupun secara makna kalimat. Proses pengecekan hasil terjemahan ini dilakukan dengan menggunakan bantuan 4 orang *annotator* agar proses dapat dilakukan lebih cepat. Kriteria utama dari *annotator* yang digunakan merupakan orang yang telah mempelajari Bahasa Inggris untuk setidaknya 12 tahun. Hal tersebut diperlukan karena *dataset* yang ingin dicek kualitasnya merupakan *dataset* yang berasal dari Bahasa Inggris. Lalu, karena keterbatasan waktu dan juga biaya, proses pengecekan kualitas hanya dilakukan pada 10% dari *dataset* MSCOCO, yaitu berjumlah sekitar 60,000. Jumlah 60,000 data tersebut ditentukan untuk melebihi penelitian yang telah dilakukan sebelumnya oleh Sinurat (2019), yang berjumlah 12,000 data. Jumlah 60,000 itu juga ditentukan dengan pertimbangan bahwa jumlah tersebut sudah cukup banyak untuk kemudian dapat dibagi kembali menjadi bagian *train*, *val*, dan *test*.

Hasil *dataset* dari proses pengecekan kualitas ini kemudian disebut sebagai *human translated (HT) dataset*. Setelah proses pengecekan selesai, *dataset* dibentuk lagi menjadi format *dataset* MSCOCO. Hasil dari proses pembuatan *dataset* ini adalah tersedianya dua buah *dataset* Bahasa Indonesia, yaitu *MT dataset* berbahasa Indonesia dan *HT dataset* berbahasa Indonesia.

III.3.2 Eksperimen

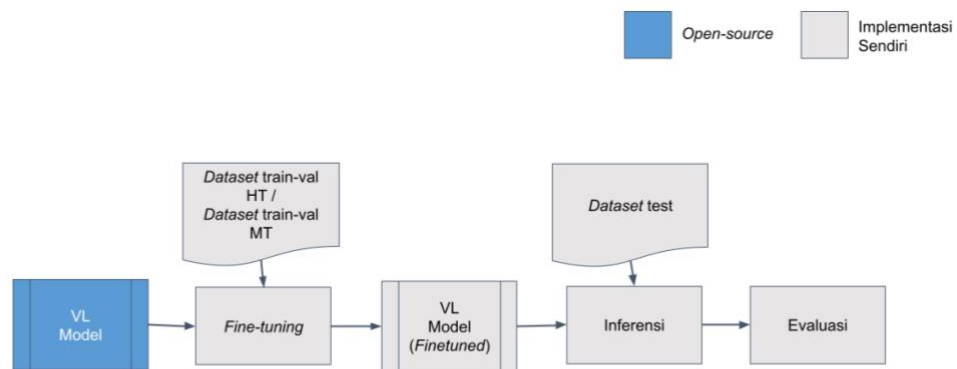
Setelah proses pembuatan *dataset* selesai, proses yang dilakukan selanjutnya adalah eksperimen. Eksperimen pada penelitian ini dilakukan dengan tujuan untuk melakukan adaptasi model-model VL pada *image captioning* berbahasa Indonesia. Adaptasi tersebut diperoleh dengan menggunakan skema *transfer learning* saat melatih model VL pada *image captioning dataset* berbahasa Indonesia. Proses *transfer learning* dilakukan dengan memanfaatkan variasi kualitas dan kuantitas *dataset* yang terbentuk pada tahapan sebelumnya. Beberapa skema model yang diujikan, yaitu sebagai berikut.

1. Model *pre-train* awal. Pada model ini, model *pre-train* awal VL digunakan untuk melakukan prediksi terhadap bagian *test* dari *image captioning dataset* berbahasa Indonesia. Eksperimen dengan model ini dilakukan untuk mengevaluasi seberapa baik kemampuan awal model VL sebelum melalui proses *fine-tuning* menggunakan *dataset* Bahasa Indonesia.
2. Model yang dikenai *finetune* menggunakan MT *dataset*. Eksperimen dengan model ini dilakukan untuk mengevaluasi seberapa baik hasil adaptasi dari model VL setelah dikenai *finetune* menggunakan *dataset* yang diterjemahkan oleh mesin, yang mungkin memiliki beberapa kesalahan pada tata Bahasa atau makna kalimat.
3. Model yang dikenai *finetune* menggunakan HT *dataset*. Eksperimen dengan model ini dilakukan untuk mengevaluasi seberapa baik hasil adaptasi dari model VL setelah dikenai *finetune* menggunakan *dataset* yang diterjemahkan oleh manusia, yang mana data yang digunakan memiliki tata Bahasa dan makna kalimat yang lebih baik jika dibandingkan dengan yang diterjemahkan menggunakan sistem penerjemah.
4. Model yang dikenai *finetune* menggunakan keseluruhan MT *dataset*. Eksperimen dengan model ini dilakukan untuk mengevaluasi seberapa baik hasil adaptasi dari model VL setelah dikenai *finetune* dengan menggunakan MT *dataset* yang memiliki jumlah yang jauh lebih banyak. Skema model ini dapat digunakan untuk membandingkan kinerja adaptasi model dengan

perbedaan jumlah data yang signifikan, yang mana jumlah data tersebut menjadi variabel tambahan yang dapat memengaruhi hasil.

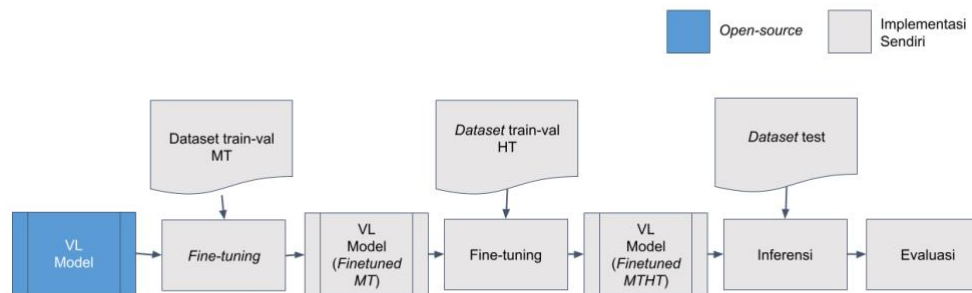
5. Model yang dikenai *finetune* menggunakan kedua *dataset* Bahasa Indonesia, MT *dataset* dan HT *dataset*. Eksperimen dengan model ini dilakukan untuk melihat seberapa baik hasil adaptasi dari model VL setelah dikenai *finetune* dengan menggunakan kedua jenis *dataset* serta menganalisis pengaruh penambahan HT *dataset* pada model.

Alur dari eksperimen tersebut dapat dilihat lebih lanjut pada Gambar III.3 dan III.4.



Gambar III.3. Alur *Fine-tuning* HT / MT Dataset

Gambar III.3 merupakan alur yang digunakan saat melakukan *fine-tuning* pada satu jenis *dataset*, seperti yang terdapat pada skema model ke 2, 3, dan 4. Model-model VL yang tersedia digunakan dan dikenai proses *fine-tuning* terhadap data *image captioning* Bahasa Indonesia. Hasil model yang dikenai proses *fine-tuning* kemudian digunakan pada tahapan selanjutnya, yaitu melakukan inferensi. Inferensi dilakukan pada bagian *test* yang berasal dari HT *dataset*. Hasil proses inferensi tersebut kemudian dievaluasi dengan menggunakan metrik evaluasi yang telah ditentukan, yaitu menggunakan metrik BLEU, CIDEr.



Gambar III.4. Alur *Fine-tuning* dengan kombinasi HT & MT Dataset

Gambar III.4. merupakan alur yang digunakan saat melakukan *finetuning* pada kombinasi dua buah jenis *dataset*, yaitu HT *dataset* & MT *dataset*. Perbedaan alur ini dibandingkan dengan alur sebelumnya yaitu terdapat dua buah tahapan *training*, yang mana *training* pertama dilakukan menggunakan MT *dataset* dan *training* kedua dilakukan menggunakan HT *dataset*. Hasil setelah kedua proses *training* tersebut kemudian digunakan pada tahapan inferensi yang menggunakan *test dataset* yang sama dengan alur sebelumnya. Hasil proses inferensi kemudian dievaluasi juga dengan menggunakan metrik evaluasi yang telah ditentukan.

BAB IV

IMPLEMENTASI, EKSPERIMEN, DAN EVALUASI

Pada bagian ini dijelaskan lebih lanjut terkait hasil solusi yang telah dilakukan. Penjelasan terkait hasil solusi meliputi implementasi yang dilakukan, eksperimen yang dilakukan, serta evaluasi dan analisis terhadap hasil dari eksperimen.

IV.1 Pembuatan *Image Captioning Dataset Bahasa Indonesia*

Tahapan awal dalam proses pembuatan *image captioning dataset* berbahasa Indonesia dimulai dengan melakukan terjemahan pada dataset MSCOCO (Lin dkk., 2014) ke dalam Bahasa Indonesia menggunakan sistem penerjemah. Untuk menerjemahkan *dataset* ke Bahasa Indonesia digunakan bantuan *library googletrans*¹ yang terdapat pada *Python*. *Library googletrans* ini berfungsi sebagai implementasi dari API *Google Translate* yang memungkinkan mendeteksi dan merubah teks dari satu bahasa ke bahasa lain. Setelah seluruh *caption* telah diterjemahkan, tahap berikutnya adalah melakukan pengecekan terhadap hasil terjemahan serta melakukan perbaikan terhadap deskripsi yang tidak tepat.

Untuk mengatasi keterbatasan waktu dan sumber daya *annotator*, maka proses pengecekan hanya dilakukan pada sampel dari *dataset* yang berjumlah sekitar 10% (60,004) dari total keseluruhan *dataset* MSCOCO yang berjumlah sekitar 600,000 pasangan gambar dan teks. Dari hasil pengecekan kualitas terjemahan, ditemukan sebanyak 8,172 perubahan yang didapatkan dari total 60,004 data yang diperiksa. Selama proses perbaikan oleh *annotator*, dilakukan juga pencatatan yang lebih rinci terkait jenis perubahan yang dilakukan pada masing-masing kalimat. Jenis perubahan yang dilakukan mengadaptasi klasifikasi jenis perbaikan yang telah digunakan sebelumnya pada penelitian NusaX (Winata dkk., 2022) yang mencakup kategori *Typos and Mechanics*, *Translation*, *Word Edit*, dan *Major Changes*.

¹ <https://pypi.org/project/googletrans/>

Informasi lebih rinci terkait statistik perubahan data tersebut dapat dilihat pada tabel IV.1.

Tabel IV.1. Statistik Jenis Perubahan

Kategori	Jumlah	% dari Total Data (60,004)
<i>Typos & Mechanic</i>	6444	10.74%
<i>Translation</i>	592	0.99%
<i>Word Edit</i>	1073	1.79%
<i>Major Changes</i>	63	0.1%
Total	8172	13.62%

Pada tabel IV.1, dapat dilihat bahwa total perubahan yang dilakukan hanya 13.62% dari total keseluruhan *dataset* yang diperiksa. Persentase yang relatif kecil ini dapat disebabkan oleh kualitas terjemahan awal dari sistem penerjemah *google translate* yang sudah cukup baik, sehingga hanya terdapat sedikit perbaikan yang diperlukan. Selain itu, *dataset* MSCOCO memiliki deskripsi yang cukup baik dalam bahasa aslinya, sehingga sistem penerjemah mampu melakukan terjemahan dengan lebih akurat.

Pada tabel IV.1, kategori *typos & mechanic* diberikan pada perubahan kalimat yang mengubah kata, tetapi tidak mengubah makna dari kalimat. Kemudian, kategori *translation* diberikan pada perubahan kalimat yang masih berbahasa inggris. Lalu, kategori *word edit* diberikan pada perubahan kalimat yang mengubah kata, sehingga terjadi perubahan makna dari kalimat awal. Terakhir, kategori *major changes* diberikan karena adanya kesalahan yang besar pada kalimat awal. Contoh-

contoh dan juga penjelasan untuk setiap kategori kesalahan yang terdapat pada tabel IV.1 dapat dilihat lebih lanjut pada Gambar IV.1, IV.2, IV.3, IV.4.



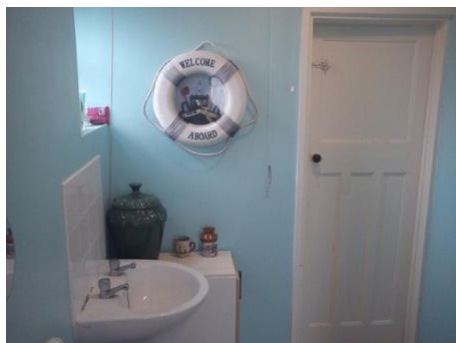
Caption awal: *Ma motorcycle parked on the gravel in front of a garage.*

Caption hasil terjemahan: Sepeda motor Ma diparkir di atas kerikil di depan garasi.

Caption hasil pengecekan: Sepeda motor diparkir di atas kerikil di depan garasi.

Gambar IV.1. Contoh kategori kesalahan *Typos & Mechanic*

Pada contoh di Gambar IV.1, pada *caption* awal terdapat kata “Ma” yang kemudian mengakibatkan terdapat kata “Ma” pada *caption* hasil terjemahan. Kata “Ma” disini dianggap sebagai sebuah kesalahan pengetikan. Untuk kasus kesalahan ini, annotator dapat menghilangkan kata “Ma” pada hasil karena penggunaan kata “Ma” disini tidak memiliki arti atau mengubah makna tertentu pada kalimat.



Caption awal:

A bathroom with walls that are painted baby blue.

Caption hasil terjemahan:

Kamar mandi dengan dinding yang dicat *baby blue*.

Caption hasil pengecekan:

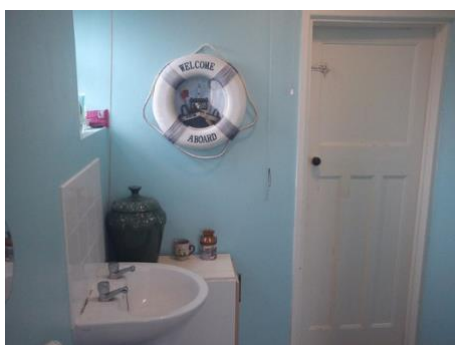
Kamar mandi dengan dinding yang dicat biru muda.

Gambar IV.2. Contoh kategori kesalahan *Translation*

Pada contoh di Gambar IV.2, pada *caption* hasil terjemahan terdapat kata “*baby blue*” yang mana kata tersebut merupakan kata yang masih berasal dari *caption* berbahasa Inggris dan dapat diartikan lebih lanjut. Karena adanya kekurangan dari terjemahan tersebut, pada kasus kategori kesalahan *Translation* annotator dapat menerjemahkan kembali kata yang masih berbahasa Inggris, yang mana pada contoh Gambar IV.2, kata “*baby blue*” diterjemahkan menjadi “biru muda”.

Caption awal:

This is a blue and white bathroom with a wall sink and a lifesaver on the wall.



Caption hasil terjemahan:

Ini adalah kamar mandi biru dan putih dengan wastafel dinding dan penyelamat di dinding.

Caption hasil pengecekan:

Ini adalah kamar mandi biru dan putih dengan wastafel dinding dan ban penyelamat di dinding.

Gambar IV.3. Contoh kategori kesalahan *Word Edit*

Pada contoh di Gambar IV.3, pada *caption* hasil terjemahan terdapat kata “*penyelamat*” yang mana kata tersebut merupakan terjemahan dari kata “*lifesaver*” yang berasal dari *caption* berbahasa Inggris. Pada kasus kategori kesalahan *Word*

Edit, annotator melakukan perubahan dengan menambahkan atau mengurangi kata agar menghasilkan kalimat yang lebih baik secara semantik. Dalam kasus ini, ditambahkan kata “ban” di depan kata “penyelamat”, sehingga *caption* hasil lebih sesuai dengan makna yang digambarkan.

Caption awal:

There is no picture or image sorry sorry.



Caption hasil terjemahan:

Tidak ada gambar atau gambar maaf maaf.

Caption hasil pengecekan:

Beberapa pengendara sepeda motor berkumpul di parkir.

Gambar IV.4. Contoh kategori kesalahan *Major Changes*

Pada contoh di Gambar IV.4, terjadi sebuah kesalahan pada proses *captioning* yang terjadi pada pembuatan *dataset* MSCOCO, sehingga *caption* awal yang terdapat pada *dataset* merupakan “*There is no picture or image sorry sorry*”. *Caption* awal tersebut tidak menggambarkan gambar sama sekali dan ketika di terjemahkan secara otomatis juga menghasilkan *caption* bahasa Indonesia yang kurang mendeskripsikan gambar. Pada kasus kategori kesalahan *Major Changes*, annotator melakukan perubahan yang dapat mengubah secara signifikan *caption* hasil terjemahan dengan *caption* hasil pengecekan.

Selain analisis terhadap kategori perubahan kalimat, dilakukan juga perhitungan *perplexity* terhadap pengecekan kualitas *dataset* menggunakan XGLM (Lin dkk.,

2021). Perhitungan *perplexity* dilakukan untuk mengecek kualitas dari *dataset*, yang mana *dataset* dengan *perplexity* yang lebih rendah lebih baik karena *language model* lebih mampu memprediksi hal tersebut. *Perplexity* dihitung menggunakan bantuan *library* dari *huggingface*². Hasil dari perhitungan *perplexity* tersebut didapatkan bahwa *dataset* setelah pengecekan mendapatkan nilai 419,67 yang mana nilai ini lebih kecil dibandingkan dengan *dataset* sebelum pengecekan yang bernilai 426,41. Hal ini berarti bahwa sebelum melalui proses pengecekan terdapat beberapa kesalahan terjemahan yang mengakibatkan kalimat tidak natural seperti kalimat manusia dan *language model* yang digunakan menjadi lebih sulit untuk memprediksi kalimat. Hal ini berbanding terbalik dengan data yang telah dicek oleh manusia. Data yang dicek oleh manusia memiliki nilai yang lebih rendah karena adanya beberapa perbaikan terhadap kalimat yang kurang tepat, sehingga kalimat yang digunakan menjadi lebih natural dan lebih mampu untuk diprediksi oleh *language model*. Namun demikian, nilai *perplexity* yang tidak jauh berbeda dengan data awal dapat disebabkan karena proses pengecekan dilakukan dengan menggunakan data MT sebagai referensi pada perbaikan, sehingga perubahan yang dilakukan relatif minimal.

Setelah *dataset* selesai, langkah selanjutnya adalah membagi *dataset*. *Dataset* hasil pengecekan dibagi menjadi *dataset train*, *val*, dan *test*, dengan rincian data *training* berjumlah 30,002; data *validation* berjumlah 9,000; dan data *test* berjumlah 21,002. *Dataset* ini kemudian digunakan pada eksperimen.

IV.2 Desain Eksperimen

Desain eksperimen mencakup beberapa hal, yaitu tujuan eksperimen, rancangan eksperimen, lingkungan eksperimen, metrik evaluasi, prosedur pelatihan, dan prosedur evaluasi. Berikut merupakan penjelasan untuk tiap bagian tersebut.

1. Tujuan Eksperimen

² <https://huggingface.co/spaces/evaluate-metric/perplexity>

Eksperimen yang dilakukan bertujuan untuk mencari model VL dengan adaptasi terbaik pada *image captioning* Bahasa Indonesia serta menganalisis dampak penggunaan data yang melewati proses pengecekan kualitas oleh manusia terhadap kinerja model VL.

2. Skenario Eksperimen

Rancangan eksperimen dilakukan dengan menggunakan variasi kualitas dan kuantitas *dataset* yang digunakan untuk *finetuning* pada model-model VL. Beberapa skema model yang diujikan, yaitu model *pre-train* dari VL, model yang dikenai *finetune* dengan HT *dataset*, model yang dikenai *finetune* dengan MT *dataset*, dan model yang dikenai *finetune* dengan kombinasi MT & HT *dataset*. Skema-skema model tersebut dilakukan pada model BLIP, OFA, dan GIT. Hasil terbaik dari model-model tersebut kemudian dibandingkan dengan model SCN & Soft Attention Indo yang dijadikan sebagai *baseline* dari eksperimen.

3. Lingkungan eksperimen

Lingkungan eksperimen yang diimplementasikan dalam penelitian ini dapat dilihat lebih jelas pada Tabel IV.2. Lingkungan eksperimen yang terdapat pada Tabel IV.2 digunakan untuk melatih tiga buah model VL, yaitu BLIP, OFA, dan juga GIT.

Tabel IV.2. Rincian Lingkungan Eksperimen

Lingkungan	Nilai
Perangkat	GPU NVIDIA A100-SXM4-40GB
Jumlah GPU	4
Ukuran memori	160GB (40GB per GPU)
Versi Python	3.8
Versi CUDA	11.7

4. Metrik Evaluasi

Metrik evaluasi yang digunakan pada eksperimen ini adalah BLEU, CIDEr, dan *Human Evaluation*. Metrik BLEU dan CIDEr memiliki tujuan utama untuk mengevaluasi tata Bahasa dari kalimat hasil, sedangkan *human evaluation* dilakukan untuk mengevaluasi semantik dari kalimat hasil.

IV.3 Hasil Eksperimen

Pada eksperimen menggunakan model BLIP, digunakan *learning rate* sebesar $1e-5$, $5e-5$, dan $1e-6$ dengan *weight decay* sebesar 0,05. *Training* dilakukan selama 3 epoch dengan ukuran *batch* yang digunakan pada *training* sebesar 32. Kemudian, pada proses inferensi, digunakan metode *beam search* dengan 3 buah *beam* serta panjang kalimat terpanjang sebesar 70. Hasil dari eksperimen pada model BLIP dapat dilihat lebih lanjut pada Tabel IV.3.

Tabel IV.3. Hasil Eksperimen terhadap Model BLIP

Jenis Eksperimen	Metrik Evaluasi				
	BLEU-1	BLEU-2	BLEU-3	BLEU-4	CIDEr
Pretrained BLIP (English)	0.022	0.005	0.002	0.001	0.022
Finetuned BLIP w/MT	0.514	0.373	0.26	0.181	1.151
Finetuned BLIP w/HT	0.541	0.394	0.277	0.196	1.249
Finetuned BLIP w/MT All	0.569	0.421	0.304	0.222	1.39
Finetuned BLIP w/HT + MT All	0.579	0.433	0.315	0.232	1.435

‘Pretrained BLIP’ merupakan *checkpoint* model BLIP yang sebelumnya telah di *pretrain* dengan menggunakan data pasangan gambar dan teks berbahasa Inggris. ‘Finetune BLIP w/MT’ merupakan model BLIP yang di *finetune* menggunakan data *machine translated* yang memiliki jumlah yang sama dengan data *human translated*. ‘Finetune BLIP w/HT’ merupakan model BLIP yang di *finetune* menggunakan data *human translated*. ‘Finetune BLIP w/MT All’ merupakan model BLIP yang di *finetune* menggunakan keseluruhan data *machine translated*. ‘Finetune BLIP w/HT + MT All’ merupakan model BLIP yang di *finetune* menggunakan data *human translated* dan *machine translated*.

Berdasarkan hasil eksperimen yang terdapat pada tabel IV.3, didapatkan bahwa model BLIP memiliki kinerja yang paling baik ketika di *finetune* menggunakan data *machine translated* dan *human translated*. Untuk tiap metriknya, nilai tertinggi dari setiap metrik terdapat pada saat model di *finetune* menggunakan data *machine translated* dan *human translated*.

Selanjutnya, pada eksperimen menggunakan model OFA, digunakan *learning rate* sebesar $1e-5$, $5e-5$, dan $1e-6$ dengan *weight decay* sebesar 0,01. *Training* dilakukan dengan jumlah epoch yang sama, yaitu 3 epoch, dengan ukuran *batch* yang digunakan pada *training* sebesar 32. Kemudian, pada proses inferensi, digunakan metode *beam search* dengan 5 buah *beam* serta panjang kalimat terpanjang sebesar 16 kata. Hasil dari eksperimen pada model OFA dapat dilihat lebih lanjut pada Tabel IV.4.

Tabel IV.4. Hasil Eksperimen terhadap Model OFA

Jenis Eksperimen	Metrik Evaluasi				
	BLEU-1	BLEU-2	BLEU-3	BLEU-4	CIDEr
Pretrained OFA (English)	0.021	0.004	0.001	0	0.018
Finetuned w/MT OFA	0.403	0.268	0.162	0.104	0.062

Tabel IV.4. Lanjutan

Jenis Eksperimen	Metrik Evaluasi				
	BLEU-1	BLEU-2	BLEU-3	BLEU-4	CIDEr
Finetuned OFA w/HT	0.431	0.296	0.192	0.127	0.815
Finetuned OFA w/MT All	0.506	0.366	0.254	0.18	1.125
Finetuned OFA w/HT + MT All	0.507	0.369	0.258	0.184	1.151

‘Pretrained OFA’ merupakan *checkpoint* model OFA yang sebelumnya telah di *pretrain* dengan menggunakan data pasangan gambar dan teks berbahasa Inggris. ‘Finetune OFA w/MT’ merupakan model OFA yang di *finetune* menggunakan data *machine translated* yang memiliki jumlah yang sama dengan data *human translated*. ‘Finetune OFA w/HT’ merupakan model OFA yang di *finetune* menggunakan data *human translated*. ‘Finetune OFA w/MT All’ merupakan model OFA yang di *finetune* menggunakan keseluruhan data *machine translated*. ‘Finetune OFA w/HT + MT All’ merupakan model OFA yang di *finetune* menggunakan data *human translated* dan *machine translated*.

Berdasarkan hasil eksperimen yang terdapat pada tabel IV.4, sama seperti eksperimen yang dilakukan pada model BLIP, didapatkan bahwa model OFA juga memiliki kinerja yang paling baik ketika di *finetune* menggunakan data *machine translated* dan *human translated*. Kemudian untuk tiap metriknya, nilai tertinggi dari setiap metrik juga terdapat pada saat model di *finetune* menggunakan data *machine translated* dan *human translated*.

Terakhir, pada eksperimen menggunakan model GIT, digunakan *learning rate* sebesar $5e-5$ dengan *weight decay* sebesar 0,01. *Training* dilakukan dengan jumlah epoch yang sama, yaitu 3 epoch, dengan ukuran *batch* yang digunakan pada *training* sebesar 16. Kemudian, pada proses inferensi, digunakan metode *beam*

search dengan 5 buah *beam* serta panjang kalimat terpanjang sebesar 50 kata. Hasil dari eksperimen pada model GIT dapat dilihat lebih lanjut pada Tabel IV.5.

Tabel IV.5. Hasil Eksperimen terhadap Model GIT

Jenis Eksperimen	Metrik Evaluasi				
	BLEU-1	BLEU-2	BLEU-3	BLEU-4	CIDEr
Pretrained GIT (English)	0.02	0.003	0.001	0	0.008
Finetuned GIT w/MT	0.401	0.259	0.164	0.105	0.639
Finetuned GIT w/HT	0.461	0.313	0.205	0.136	0.859
Finetuned GIT w/MT All	0.519	0.365	0.251	0.174	1.075
Finetuned GIT w/HT + MT All	0.541	0.387	0.268	0.187	1.19

‘Pretrained GIT’ merupakan *checkpoint* model GIT yang sebelumnya telah di *pretrain* dengan menggunakan data pasangan gambar dan teks berbahasa Inggris. ‘Finetune GIT w/MT’ merupakan model GIT yang di *finetune* menggunakan data *machine translated* yang memiliki jumlah yang sama dengan data *human translated*. ‘Finetune GIT w/HT’ merupakan model GIT yang di *finetune* menggunakan data *human translated*. ‘Finetune GIT w/MT All’ merupakan model GIT yang di *finetune* menggunakan keseluruhan data *machine translated*. ‘Finetune GIT w/HT + MT All’ merupakan model GIT yang di *finetune* menggunakan data *human translated* dan *machine translated*.

Berdasarkan hasil eksperimen yang terdapat pada tabel IV.5, didapatkan bahwa model GIT memiliki kinerja yang paling baik ketika di *finetune* dengan menggunakan data *human translated* dan *machine translated*. Kemudian untuk tiap

metriknya, nilai tertinggi dari setiap metrik juga terdapat pada saat model di *finetune* menggunakan data *machine translated* dan *human translated*.

IV.4 Evaluasi

Berikut ini adalah hasil terbaik dari eksperimen pada setiap model VL. Hasil model *image captioning* kemudian dibandingkan dengan model SCN + Soft Attention Indo yang dijadikan sebagai *baseline*. Hasil perbandingan tersebut dapat dilihat lebih lanjut pada tabel IV.6.

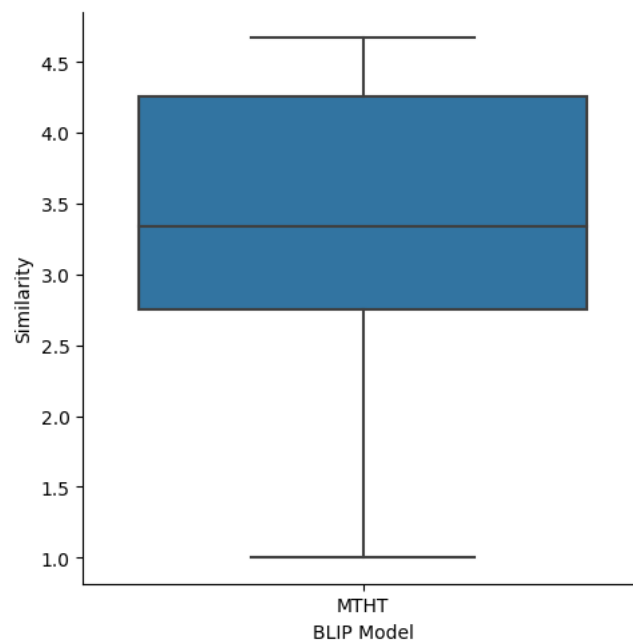
Tabel IV.6. Hasil *Image Captioning* Bahasa Indonesia

Model	Metrik Evaluasi				
	BLEU-1	BLEU-2	BLEU-3	BLEU-4	CIDEr
SCN + Soft Attention	0.474	0.321	0.214	0.145	0.942
BLIP (Finetuned w/HT + MT All)	0.579	0.433	0.315	0.232	1.435
OFA (Finetuned w/HT + MT All)	0.507	0.369	0.258	0.184	1.151
GIT (Finetuned w/HT + MT All)	0.541	0.387	0.268	0.187	1.19

Berdasarkan tabel IV.6, didapatkan bahwa model BLIP yang dikenai *finetune* pada kombinasi data HT & MT memiliki hasil yang terbaik untuk *image captioning* Bahasa Indonesia. Kemudian didapatkan juga bahwa model-model VL yang digunakan mampu untuk mendapatkan kinerja yang lebih baik pada semua skor pada metrik otomatis jika dibandingkan dengan model *image captioning* Bahasa Indonesia SCN + Soft Attention yang dijadikan sebagai *baseline*.

Kemudian, dilakukan juga beberapa penilaian manual yang dilakukan untuk mengevaluasi makna semantik dari kalimat. Penilaian dilakukan oleh 3 orang evaluator yang memberikan nilai berdasarkan beberapa panduan pada Lampiran C. Karena adanya keterbatasan waktu dan sumber daya evaluator, penilaian manual

hanya dilakukan untuk model yang memperoleh hasil terbaik pada eksperimen. Penilaian manual pertama dilakukan untuk membandingkan kemiripan makna *caption* hasil dengan kalimat *ground truth*. Evaluasi dilakukan terhadap 30 sampel hasil prediksi model BLIP yang dikenai *finetune* menggunakan kombinasi data HT & MT. Penilaian dari ketiga evaluator kemudian dirata-rata dan distribusi penilaian terhadap 30 sampel dapat dilihat lebih jelas pada Gambar IV.5.

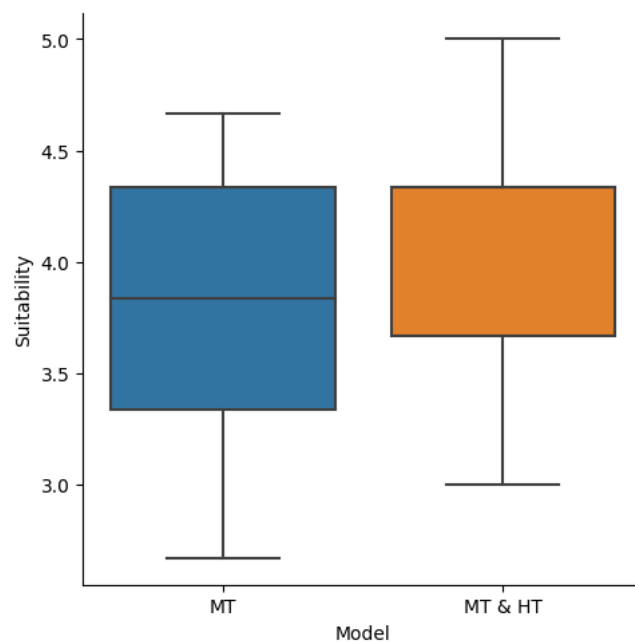


Gambar IV.5. Hasil *Human Evaluation* Kategori ‘*Similarity*’

Pada Gambar IV.5, sumbu Y merupakan penilaian ‘*Similarity*’ yang mengukur seberapa sesuai makna dari *caption* kandidat dibandingkan dengan *caption* referensi atau *ground truth*. Penilaian ‘*Similarity*’ memiliki nilai yang berada pada *range* 1-5 dan diberikan berdasarkan rubrik yang terdapat pada Gambar C.1. Kemudian, sumbu X memperlihatkan jenis dari model yang digunakan. Dari Gambar IV.5, dapat dilihat bahwa menurut evaluator, rata-rata kemiripan *caption* kandidat dan *caption* referensi berada di sekitar angka 3 dan 4. Hal tersebut berarti hasil *caption* dari model memiliki makna yang sesuai dengan *ground truth*, tetapi

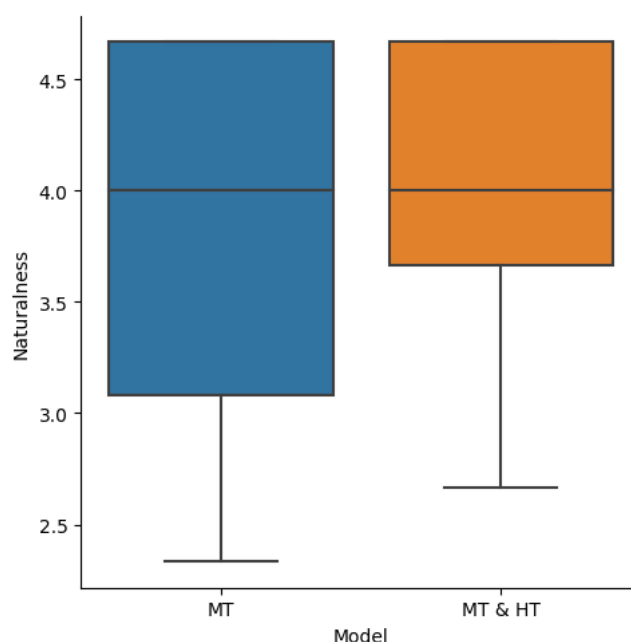
terdapat beberapa perbedaan kata-kata pada kedua kalimat. Kemudian, dapat dilihat juga bahwa cukup sulit bagi model untuk menghasilkan *caption* yang sangat mirip atau sangat berbeda terhadap *ground truth*.

Kemudian, evaluasi manual selanjutnya dilakukan untuk mengevaluasi bagaimana dampak dari penambahan data HT pada kinerja model. Pada penilaian ini, diambil 30 sampel kalimat yang sama dengan kalimat yang digunakan pada penilaian sebelumnya. Sampel kalimat diambil dari hasil data uji yang didapatkan dari model BLIP yang dikenai *finetune* pada data MT & model BLIP yang dikenai *finetune* dengan menggunakan kombinasi data MT & data HT. Penilaian manual dilakukan untuk membandingkan perubahan terkait kesesuaian dan kenaturalan dari kalimat ketika ditambahkan data HT. Sama seperti penilaian sebelumnya, penilaian yang didapat dari ketiga evaluator dirata-rata dan distribusi penilaiannya dapat dilihat lebih jelas pada Gambar IV.6 dan Gambar IV.7.



Gambar IV.6. Hasil *Human Evaluation* Kategori ‘*Suitability*’

Pada Gambar IV.6, sumbu Y merupakan penilaian ‘*Suitability*’ yang mengukur seberapa sesuai *caption* kandidat dalam mendeskripsikan sebuah gambar. Penilaian ‘*Suitability*’ memiliki nilai yang berada pada *range* 1-5 dan diberikan berdasarkan rubrik yang terdapat pada Gambar C.2. Kemudian, sumbu X memperlihatkan jenis dari model yang digunakan. Dari Gambar IV.6, dapat dilihat bahwa model yang dilatih menggunakan kombinasi data MT & HT lebih sesuai dalam mendeskripsikan objek-objek yang terdapat pada gambar. Terlihat bahwa evaluator memberikan nilai yang lebih rendah pada *caption* yang dihasilkan oleh model yang dikenai *finetune* pada data MT. Kemudian, rata-rata evaluator memberikan nilai yang berada di nilai 4 untuk kedua jenis model. Hal tersebut berarti secara umum kedua jenis model mampu untuk menghasilkan *caption* yang lumayan akurat. Lalu, dapat dilihat pada model yang menggunakan kombinasi data MT & HT memiliki batas atas yang lebih tinggi yang dapat mencapai nilai 5. Hal tersebut berarti penambahan data HT mampu membuat model memberikan *caption* yang lebih akurat dan detail.



Gambar IV.7. Hasil *Human Evaluation* Kategori ‘Naturalness’

Kemudian, pada Gambar IV.7, sumbu Y merupakan penilaian ‘*Naturalness*’ yang mengukur seberapa natural hasil *caption* kandidat. Penilaian ‘*Naturalness*’ memiliki nilai yang berada pada *range* 1-5 dan diberikan berdasarkan rubrik yang terdapat pada Gambar C.3. Kemudian, sumbu X memperlihatkan jenis dari model yang digunakan. Dari Gambar IV.7, dapat dilihat bahwa rata-rata evaluator memberikan nilai yang berada di nilai 4 untuk kedua jenis model. Hal tersebut berarti secara umum kedua jenis model mampu untuk menghasilkan *caption* yang fasih, memiliki tata bahasa yang baik, dan terlihat alami. Lalu, dapat dilihat bahwa ketika hanya menggunakan data MT terdapat beberapa evaluator yang menilai *caption* memiliki kesalahan tata bahasa atau ungkapan yang aneh. Nilai tersebut juga terdapat pada model yang menggunakan kombinasi data MT & HT, tetapi dapat dilihat bahwa nilai batas bawah tersebut lebih rendah ketika menggunakan jenis model MT. Hal tersebut berarti model yang ditambah dengan HT lebih mampu memberikan *caption* yang lebih natural jika dibandingkan dengan hanya menggunakan data MT.

IV.4.1 Analisis Hasil Eksperimen

Berdasarkan hasil eksperimen yang telah dilakukan, beberapa poin analisis yang didapat yaitu sebagai berikut:

1. Secara keseluruhan, hasil eksperimen menunjukkan bahwa melakukan *finetune* model-model VL dengan data tambahan (*machine translated*, *human translated*, atau keduanya) mampu meningkatkan kinerja secara signifikan dibandingkan dengan hanya menggunakan model VL *pretrained* yang telah disediakan. Hal ini karena model-model *pretrained* VL yang digunakan tidak di *pretrain* menggunakan data Bahasa Indonesia, sehingga tidak mampu menghasilkan kalimat dalam Bahasa Indonesia yang memiliki tata bahasa dan makna yang sesuai.
2. Hasil eksperimen *finetune* model dengan data *machine translated* menunjukkan bahwa dengan melakukan *finetune* model terhadap data

terjemahan, model mampu beradaptasi untuk menangkap lebih banyak pola dan kalimat pada bahasa yang baru serta mampu menghasilkan kalimat dengan bahasa Indonesia yang baik dan benar. Namun demikian, kinerja tersebut masih lebih rendah jika dibandingkan dengan menggunakan data *human translated*. Hal tersebut karena penggunaan data bersih hasil kurasi manusia memiliki kalimat yang lebih baik dari segi tata bahasa dan juga makna kalimat, sehingga lebih sesuai dengan distribusi kalimat pada bahasa Indonesia yang baik dan benar.

3. Model yang di *finetune* menggunakan keseluruhan data *machine translated* memiliki kinerja yang lebih baik jika dibandingkan dengan menggunakan data *human translated* dan sebagian data *machine translated*. Hal ini menunjukkan bahwa peningkatan yang substansial dalam ukuran dataset yang digunakan membuat model lebih banyak menangkap variasi bahasa yang terdapat dalam bahasa Indonesia sehingga berdampak pada kemampuan model untuk menghasilkan kalimat yang memiliki tata bahasa dan makna yang lebih sesuai.
4. Model yang di *finetune* menggunakan kombinasi *dataset machine translated* dan *human translated* memiliki kinerja yang lebih baik jika dibandingkan dengan hanya menggunakan satu jenis *dataset*. Hal ini menunjukkan bahwa penambahan data bersih hasil kurasi manusia kedalam *noisy* data hasil terjemahan mesin dapat meningkatkan performa model dalam *image captioning* dikarenakan adanya peningkatan keragaman yang diikuti dengan peningkatan kualitas penulisan kalimat, sehingga mampu memberikan kalimat yang lebih detail serta lebih natural.
5. Hasil eksperimen yang menggunakan model BLIP secara umum memiliki kinerja yang lebih baik jika dibandingkan dengan menggunakan model-model lainnya. Hal ini dapat disebabkan karena terdapat proses *pretrain* & data tambahan pada model BLIP berupa *bootstrapping dataset* yang didapatkan dari proses *filtering* dan *captioning* pada *dataset pretrain*. Hal

tersebut membuat arsitektur model BLIP mampu memberikan kinerja yang lebih baik jika *dataset caption* yang digunakan semakin bervariasi.

IV.4.2 Analisis Error Hasil

Pada tahapan analisis eror, digunakan 100 sampel yang diambil dari prediksi model dengan hasil terbaik. Dari 100 sampel tersebut kemudian diambil kesalahan-kesalahan yang cukup umum terdapat pada kalimat yang dihasilkan pada model. Dengan adanya analisis eror, diharapkan mampu ditelusuri lebih lanjut terkait kelemahan pada model dan dapat mencari cara untuk mengatasi kekurangan tersebut.

Kesalahan pertama yang cukup umum adalah model yang dihasilkan kurang baik dalam mendeteksi tulisan yang terdapat pada tanda-tanda jalan. Beberapa contoh kesalahan yang dihasilkan dapat dilihat lebih lanjut pada lampiran D. Hal ini dapat disebabkan oleh *caption* dari data latih yang hanya mendeskripsikan sebagai objek tanda jalan tetapi tidak menuliskan tulisan yang terdapat pada tanda jalan tersebut. Hal tersebut dapat diatasi dengan memvariasikan *caption* yang terdapat pada data *training*, khususnya pada gambar yang menunjukkan tulisan pada tanda jalan sebagai fokus utamanya.

Kemudian, kesalahan lainnya adalah model cukup sering untuk menghasilkan kalimat yang menggunakan kata “di”, sehingga penggunaan kata “di” menjadi berlebihan dan membuat kalimat menjadi tidak natural. Penggunaan kata “di” yang berlebih ini dapat disebabkan oleh banyaknya penggunaan kata “di” pada pelatihan. Untuk menangani permasalahan ini terdapat beberapa hal yang dapat dilakukan seperti menambah variasi preposisi yang digunakan pada *caption* data *training*, ataupun melakukan pemrosesan pada *dataset* dengan mengurangi kata-kata ganti.

BAB V

KESIMPULAN DAN SARAN

Pada bagian ini dijelaskan lebih lanjut terkait kesimpulan pada hal-hal baru yang relevan terkait ketercapaian tujuan tugas akhir serta saran-saran pengembangan yang dapat dilakukan selanjutnya.

V.1 Kesimpulan

Berdasarkan eksperimen dan analisis yang telah dilakukan, beberapa kesimpulan yang diperoleh dari tugas akhir ini yaitu:

1. Model BLIP yang dikenai *finetune* dengan kombinasi data *machine translated* dan *human translated* memiliki kemampuan adaptasi bahasa paling baik pada *image captioning* Bahasa Indonesia. Model tersebut mencapai nilai BLEU 1,2,3,4 secara berturut-turut yaitu sebesar 57.9, 43.3, 31.5, 23.2 serta nilai CIDEr sebesar 143.5. Nilai rata-rata BLEU dan CIDEr tersebut meningkat sebesar 78% dan 52% dibandingkan dengan *baseline* yang tidak menggunakan model VL.
2. Penambahan *dataset* Bahasa Indonesia yang dicek kualitasnya oleh manusia memberikan hasil yang lebih baik jika dibanding dengan hanya menggunakan data terjemahan, karena penambahan *dataset* HT memberikan peningkatan keragaman yang diikuti dengan peningkatan kualitas penulisan kalimat sehingga mampu menghasilkan kalimat yang lebih detail dan natural.

V.2 Saran

Penelitian terkait *image captioning* Bahasa Indonesia masih dapat dikembangkan supaya menghasilkan kinerja yang lebih baik. Berikut merupakan beberapa saran yang dapat diberikan untuk penelitian selanjutnya:

1. Membuat *dataset* Bahasa Indonesia dari awal yang lebih sesuai dengan geografis negara Indonesia. Hal ini disebabkan karena penggunaan *dataset* MSCOCO memiliki beberapa objek yang tidak umum terdapat pada Indonesia, contohnya seperti trem.
2. Menggunakan model VL *multilingual* atau *monolingual* yang di *pretrain* menggunakan data berbahasa Indonesia. Pada penelitian ini model yang digunakan merupakan model *monolingual* yang di *pretrain* menggunakan *dataset* berbahasa Inggris.

DAFTAR REFERENSI

- Anderson, P., Fernando, B., Johnson, M., & Gould, S. (2016). Spice: Semantic propositional image caption evaluation. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part V 14* (pp. 382–398). Springer International Publishing.
- Bai, S., & An, S. (2018). A survey on automatic image caption generation. *Neurocomputing*, 311, 291–304.
- Baltrušaitis, T., Ahuja, C., & Morency, L. P. (2018). Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2), 423–443.
- Banerjee, S., & Lavie, A. (2005, June). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization* (pp. 65–72).
- Dashtipour, K., Poria, S., Hussain, A., Cambria, E., Hawalah, A. Y., Gelbukh, A., & Zhou, Q. (2016). Multilingual sentiment analysis: state of the art and independent comparison of techniques. *Cognitive computation*, 8(4), 757–771.
- Du, Y., Liu, Z., Li, J., & Zhao, W. X. (2022). A survey of vision-language pre-trained models. *arXiv preprint arXiv:2202.10936*.
- Farhadi, A., Hejrati, M., Sadeghi, M. A., Young, P., Rashtchian, C., Hockenmaier, J., & Forsyth, D. (2010). Every picture tells a story: Generating sentences from images. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5–11, 2010, Proceedings, Part IV 11* (pp. 15–29). Springer Berlin Heidelberg.
- Gan, Z., Gan, C., He, X., Pu, Y., Tran, K., Gao, J., ... & Deng, L. (2017). Semantic compositional networks for visual captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5630–5639).
- Gan, Z., Li, L., Li, C., Wang, L., Liu, Z., & Gao, J. (2022). Vision-language pre-training: Basics, recent advances, and future trends. *Foundations and Trends® in Computer Graphics and Vision*, 14(3–4), 163–352.
- Hodosh, M., Young, P., & Hockenmaier, J. (2013). Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47, 853–899.
- Hossain, M. Z., Sohel, F., Shiratuddin, M. F., & Laga, H. (2019). A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CsUR)*, 51(6), 1–36.

- Jelinek, F., Mercer, R. L., Bahl, L. R., & Baker, J. K. (1977). Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1), S63-S63. Acoustical Society of America.
- Kasai, J., Sakaguchi, K., Dunagan, L., Morrison, J., Bras, R. L., Choi, Y., & Smith, N. A. (2021). Transparent human evaluation for image captioning. *arXiv preprint arXiv:2111.08940*.
- Li, L. H., Yatskar, M., Yin, D., Hsieh, C. J., & Chang, K. W. (2019). Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*.
- Li, J., Li, D., Xiong, C., & Hoi, S. (2022, June). Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning* (pp. 12888-12900). PMLR.
- Lin, C. Y. (2004, July). Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out* (pp. 74-81).
- Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13* (pp. 740-755). Springer International Publishing.
- Lin, X. V., Mihaylov, T., Artetxe, M., Wang, T., Chen, S., Simig, D., ... & Li, X. (2021). Few-shot learning with multilingual language models. *arXiv preprint arXiv:2112.10668*.
- Long, S., Cao, F., Han, S. C., & Yang, H. (2022). Vision-and-language pretrained models: A survey. *arXiv preprint arXiv:2204.07356*.
- Lu, J., Batra, D., Parikh, D., & Lee, S. (2019). Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32.
- Mitchell, M., Dodge, J., Goyal, A., Yamaguchi, K., Stratos, K., Han, X., ... & Daumé III, H. (2012, April). Midge: Generating image descriptions from computer vision detections. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 747-756).
- Mogadala, A., Kalimuthu, M., & Klakow, D. (2021). Trends in integration of vision and language research: A survey of tasks, datasets, and methods. *Journal of Artificial Intelligence Research*, 71, 1183-1317.
- Mulyanto, E., Setiawan, E. I., Yuniarno, E. M., & Purnomo, M. H. (2019, June). Automatic Indonesian Image Caption Generation using CNN-LSTM Model and FEEH-ID Dataset. In *2019 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)* (pp. 1-5). IEEE.

- Mulyawan, R., Sunyoto, A., & Muhammad, A. H. (2022, August). Automatic Indonesian Image Captioning using CNN and Transformer-Based Model Approach. In *2022 5th International Conference on Information and Communications Technology (ICOIACT)* (pp. 355-360). IEEE.
- Nugraha, A. A., & Arifianto, A. (2019, July). Generating image description on Indonesian language using convolutional neural network and gated recurrent unit. In *2019 7th International Conference on Information and Communication Technology (ICoICT)* (pp. 1-6). IEEE.
- Ordonez, V., Kulkarni, G., & Berg, T. (2011). Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24.
- Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002, July). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics* (pp. 311-318).
- Sharma, P., Ding, N., Goodman, S., & Soricut, R. (2018, July). Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 2556-2565).
- Sharma, H., & Padha, D. (2023). A comprehensive survey on image captioning: from handcrafted to deep learning-based techniques, a taxonomy and open research issues. *Artificial Intelligence Review*, 1-43.
- Shi, W. Z., Zeng, F., Zhang, A., Tong, C., Shen, X., Liu, Z., & Shi, Z. (2022). Online public opinion during the first epidemic wave of COVID-19 in China based on Weibo data. *Humanities and Social Sciences Communications*, 9(1).
- Shin, A., Ishii, M., & Narihira, T. (2022). Perspectives and prospects on transformer architecture for cross-modal tasks with language and vision. *International journal of computer vision*, 130(2), 435-454.
- Singh, A., Hu, R., Goswami, V., Couairon, G., Galuba, W., Rohrbach, M., & Kiela, D. (2022). Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 15638-15650).
- Sinurat, R. A. O. (2019). Pembangkitan deskripsi gambar dalam bahasa indonesia dengan pendekatan semantic compositional networks.
- Stefanini, M., Cornia, M., Baraldi, L., Cascianelli, S., Fiameni, G., & Cucchiara, R. (2022). From show to tell: A survey on deep learning-based image captioning. *IEEE transactions on pattern analysis and machine intelligence*, 45(1), 539-559.

- Tan, H., & Bansal, M. (2019). Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*.
- Ushiku, Y., Harada, T., & Kuniyoshi, Y. (2012, October). Efficient image annotation for automatic sentence generation. In *Proceedings of the 20th ACM international conference on Multimedia* (pp. 549-558).
- Vedantam, R., Lawrence Zitnick, C., & Parikh, D. (2015). Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4566-4575).
- Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2016). Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *IEEE transactions on pattern analysis and machine intelligence*, 39(4), 652-663.
- Wang, C., Yang, H., Bartz, C., & Meinel, C. (2016, October). Image captioning with deep bidirectional LSTMs. In *Proceedings of the 24th ACM international conference on Multimedia* (pp. 988-997).
- Wang, J., Yang, Z., Hu, X., Li, L., Lin, K., Gan, Z., ... & Wang, L. (2022). Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100*.
- Wang, P., Yang, A., Men, R., Lin, J., Bai, S., Li, Z., ... & Yang, H. (2022, June). Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning* (pp. 23318-23340). PMLR.
- Wilie, B., Vincentio, K., Winata, G. I., Cahyawijaya, S., Li, X., Lim, Z. Y., ... & Purwarianti, A. (2020). IndoNLU: Benchmark and resources for evaluating Indonesian natural language understanding. *arXiv preprint arXiv:2009.05387*.
- Winata, G. I., Aji, A. F., Cahyawijaya, S., Mahendra, R., Koto, F., Romadhony, A., ... & Ruder, S. (2022). Nusax: Multilingual parallel sentiment dataset for 10 indonesian local languages. *arXiv preprint arXiv:2205.15960*.
- Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., ... & Bengio, Y. (2015, June). Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning* (pp. 2048-2057). PMLR.
- You, Q., Jin, H., Wang, Z., Fang, C., & Luo, J. (2016). Image captioning with semantic attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4651-4659).
- Young, P., Lai, A., Hodosh, M., & Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2, 67-78.

Zhang, J., Huang, J., Jin, S., & Lu, S. (2023). Vision-language models for vision tasks: A survey. *arXiv preprint arXiv:2304.00685*.

Lampiran A. Parameter Eksperimen pada BLIP, OFA, dan GIT

Tabel A.1. menunjukkan *hyperparameter* yang digunakan pada eksperimen terhadap model-model yang digunakan.

Tabel V.1. *Hyperparameter* eksperimen BLIP, OFA, dan GIT

Jenis Parameter	BLIP	OFA	GIT
<i>Learning rate</i>	1e-5, 5e-5, 1e-6	1e-5	1e-5
<i>Weight decay</i>	0.05	0.01	0.01
<i>Batch size</i>	32	16	16
Epoch	3	3	3
Teknik <i>searching</i>	<i>Beam search</i>	<i>Beam search</i>	<i>Beam search</i>
Maksimal panjang kalimat	70	16 token	50

Lampiran B. Panduan Proses Anotasi

Untuk memastikan agar proses anotasi memberikan hasil yang baik, dibuat sebuah panduan pada proses anotasi. Beberapa panduan proses anotasi yang digunakan oleh *annotator* dapat dilihat pada Gambar B.1.

QUALITY CHECK COCO ID TRANSLATION DATASET	
Tugas: Mengecek hasil translate kalimat bahasa inggris ke bahasa indo	
How?	
- Terdapat bagian 'caption' yang berisi kalimat bahasa inggris & bagian 'translated' yang berisi kalimat translate bahasa indonesia	
- Hal yang dilakukan adalah melakukan pengecekan terjemahan, jika ada yang kurang sesuai maka dibenarkan	
- Hasilnya perubahan kalimat diisi di bagian 'checked', jika tidak terdapat perubahan maka kalimat sama seperti bagian 'translated'	
- Jika perlu memerlukan informasi lebih lanjut terkait kalimat, maka dapat dilihat langsung pada <i>link</i> gambar	
- Di bagian 'tag' terdapat beberapa kategori yang dapat diisi:	
	" " - jika tidak terdapat perubahan
	"Typo" - jika perubahan berkaitan dengan typo pada kalimat indonesia
	"Translation" - jika perubahan berkaitan dengan kalimat inggris pada kalimat indonesia
	"Word" - jika perubahan berkaitan dengan parafrase kata atau kalimat
	"Major" - jika perubahan mengubah kalimat awal kalimat
- Jika kesulitan atau ada yang tidak dimengerti terkait suatu kalimat maka 'tag' dapat ditandai dahulu dengan '?'	
Nanti akan dibantu dilakukan pengecekan	

Gambar V.1. Panduan Perbaikan Terjemahan Dataset

Lampiran C. Rubrik *Human Evaluation*

Untuk memastikan agar proses *human evaluation* memiliki standarisasi yang sama, maka dibuat sebuah rubrik untuk menilai hasil kalimat. Beberapa rubrik komponen penilaian kalimat meliputi penilaian terkait *similarity*, *suitability*, dan *naturalness*. Komponen penilaian *similarity* merupakan penilaian yang dilakukan untuk membandingkan kemiripan *caption* kandidat dengan *caption* referensi. Komponen penilaian *suitability* merupakan penilaian yang mengukur seberapa sesuai *caption* dalam mendeskripsikan kalimat. Terakhir, komponen penilaian *naturalness*, merupakan pengukuran yang menilai kealamian dari kalimat yang dihasilkan. Rubrik-rubrik dari setiap komponen penilaian yang digunakan oleh *evaluator* dapat dilihat pada Gambar C.1, Gambar C.2, dan Gambar C.3.

Similarity	
Tujuan:	Membandingkan kemiripan makna caption hasil dengan kalimat awal (1: Tidak Mirip, 5: Sangat Mirip)
Nilai	Kriteria Penilaian
1	Kedua teks sama sekali berbeda dan tidak memiliki kesamaan yang berarti.
2	Ada beberapa kesamaan atau tumpang tindih di antara kedua teks, namun sebagian besarnya berbeda dan tidak terlalu mirip.
3	Kedua teks tersebut menunjukkan tingkat kesamaan rata-rata; terdapat tingkat tumpang tindih atau kesamaan yang moderat, namun tidak terlalu mirip.
4	Kedua teksnya sangat mirip, memiliki kesamaan atau isi yang signifikan, namun tetap ada perbedaan.
5	Kedua teks tersebut hampir identik atau sangat erat kaitannya, dengan sedikit perbedaan.

Gambar C.2. Rubrik Penilaian *Similarity*

Suitability	
Tujuan:	Mengukur seberapa sesuai caption dalam mendeskripsikan elemen-elemen pada gambar (1: Tidak Tepat, 5: Sangat Tepat)
Nilai	Kriteria Penilaian
1	Caption yang dihasilkan tidak akurat dan gagal menggambarkan elemen utama pada gambar
2	Caption yang dihasilkan sebagian akurat tetapi melewatkan detail penting dari gambar
3	Caption yang dihasilkan menjelaskan secara akurat beberapa aspek utama dari gambar, tetapi mungkin menghilangkan atau salah mengartikan elemen tertentu
4	Caption yang dihasilkan sebagian besar akurat dan memberikan deskripsi gambar yang baik, tetapi terdapat sedikit kekurangan
5	Caption yang dihasilkan sangat akurat , menangkap semua elemen penting dari gambar dengan sangat detail

Gambar C.3. Rubrik Penilaian *Suitability*

Naturalness	
Tujuan:	Mengukur seberapa natural caption dalam mendeskripsikan gambar (1: Tidak Natural, 5: Sangat Natural)
Nilai	Kriteria Penilaian
1	Caption yang dihasilkan aneh/janggal, tata bahasanya salah, dan tidak terlihat seperti bahasa alami manusia
2	Caption yang dihasilkan memiliki beberapa ungkapan yang aneh/janggal dan terdapat sedikit kesalahan pada tata bahasa
3	Caption yang dihasilkan koheren/berhubungan dan sebagian besar benar secara tata bahasa , tetapi mungkin memiliki beberapa frasa yang sedikit tidak wajar
4	Caption yang dihasilkan fasih , memiliki tata bahasa yang baik , dan terlihat alami , tetapi dapat sedikit diperbaiki/ditingkatkan
5	Caption yang dihasilkan sangat fasih, tata bahasanya sempurna , dan terlihat sangat alami seperti teks buatan manusia

Gambar C.4. Rubrik Penilaian *Naturalness*

Lampiran D. Contoh Hasil

Berikut merupakan beberapa hasil dari model *image captioning* pada beberapa *data set* uji.

Tabel D.2. Hasil *Image Captioning*

Gambar	Hasil Caption
	beberapa sepeda motor diparkir di depan sebuah toko.
	tanda satu arah dan satu arah di depan bangunan bata.

Gambar	Hasil <i>Caption</i>
	<p>seorang pria berdiri di belakang bar di sebuah restoran.</p>
	<p>sebuah mobil berhenti di lalu lintas di persimpangan.</p>