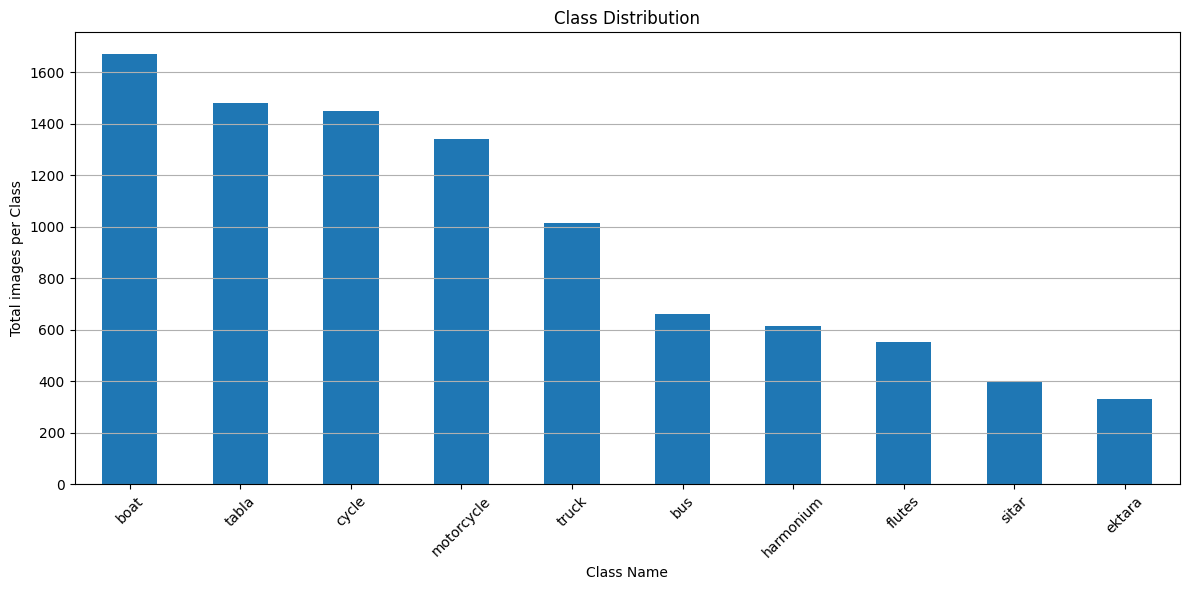
1.

Link Drive model = <https://drive.google.com/drive/folders/1vRjq0sRHr0cJCVWEGgC_72NhR3e1ubgx?usp=sharing>

# 1-a. Melakukan exploratory analysis terhadap data image, seperti melakukan visualisasi beberapa sample image, distribusi masing-masing kelas, mean dan standard deviasi per channel, dan histogram RGB.

EDA =

* visualisasi beberapa sample image, Melakukan pengambilan data gambar secara acak lalu menampilkan 10 gambar dengan plot
* Distribusi kelas di setiap file, File 2 Sesuai dengan kriteria soal, dimana memiliki 10 kelas
* distribusi masing-masing kelas, menghitung jumlah penyebaran gamabar per kelas masing masing



* mean dan standar deviasi per channel, menghitung mean dan standar deviasi dari 200 gambar sample untuk melihat representasi warna
* histogram RGB, Melihat penyebaran warna R G B pada sampel gambar

# 1-b. Pemodelan dengan menggunakan algortima berbasis CNN, dengan pengembangan dari scratch di mana anda akan mendesain arsitektur sendiri dan melakukan hyperparameter tuning seperti jumlah hidden layer, epoch, batch size, activation function, learning rate dan optimization function hingga menghasilkan hasil yang terbaik. Berikan penjelasan mengenai arsitektur yang Anda buat, serta analisa perfomance dari model tersebut.

**Data preparation**, menyesuaikan path data label dan image dataframe, agar sesuai dengan input untuk pytorch, melakukan transform resize data menjadi 128x128, Membagi dataset ke train and val loader (80:20)

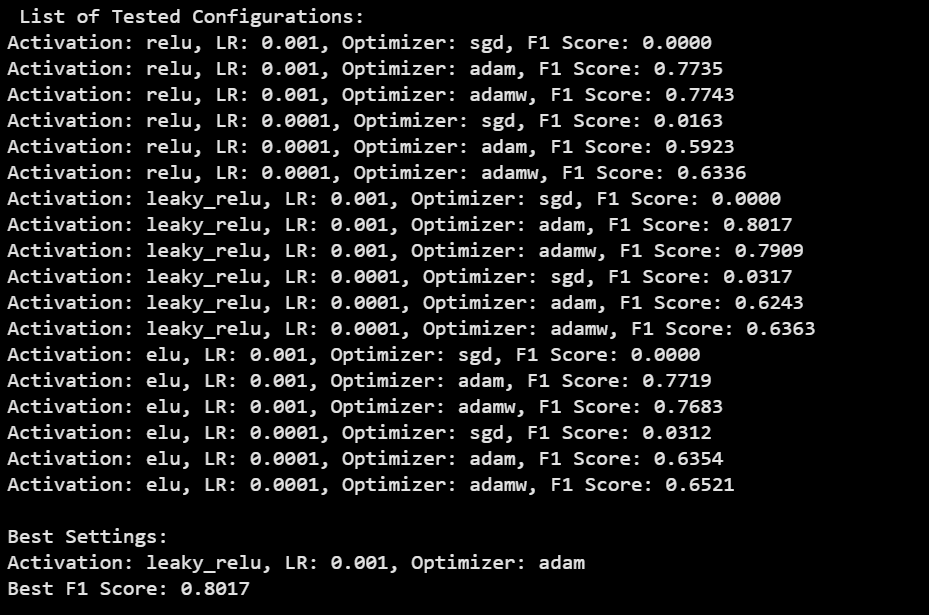
**Create scratch CNN,** membuat arsitektur CNN berupa:

* 3 layer blok konfulusi (16,32,64)
* Activation function = untuk mengubah hasil menjadi non linear
* Maxpooling layer = untuk reduce dimension
* Flatten layer, transform hasil konfulusi 3D ke 1D
* Fully Connected layer, proses dropout fitur dan layer prediksi kelas
* Sigmoid output, mengkonversi value probabilitas untuk menjadi jawaban prediksi label

**Membuat skema training & melakukan uji coba hyperparameter tuning**, dimana settingan berupa:

1. Epoch = 3
2. Batch Size = 32
3. Hidden layer sasma seperti konfigurasi sebelumnya
4. Kombinasi activation function, learning rate dan optimization function

Dengan hasil sebagai berikut:



Dengan mengambil kombinasi pengaturan terbaik yakni (leaky\_relu, LR: 0.001, Optimizer: adam)

Didapatkan hasil performance sebagai berikut,

Final Performance Metrics:

Accuracy : 0.7067

Precision : 0.8723

Recall : 0.7874

F1 Score : 0.8264

**Kesimpulan hasil,** akurasi sebesar 0.7067 mengindikasikan bahwa model baik dalam mengidentifikasi label, precision sebesar 0.8723 mengindikasikan bahwa model memiliki tingkat prediksi terhadap label dan berhasil memprediksi label tersebut sesuai dengan gambarnya melebihi 80%, Recall sebesar 0.7874, mengindikasikan bahwa total label yang berhasil diprediksi dengan benar mencapai 70 persen keatas. Dan F1 Score mengindikasikan bahwa kualitas model secara keseluruhan mampu bertahan di kisaran 80%

# 1-c. Mengembangkan model CNN dengan menggunakan transfer learning dengan melakukan hyperparameter tuning seperti arsitektur pada fully connected layer, epoch, batch size, activation function, learning rate dan optimization function. Berikan penjelasan mengenai arsitektur yang Anda gunakan, serta analisa perfomance dari model tersebut.

**Data preparation**, menyesuaikan path data label dan image dataframe, agar sesuai dengan input untuk pytorch, melakukan transform resize data menjadi 224x224 agar sesuai dengan input resnet18, Membagi dataset ke train and val loader (80:20)

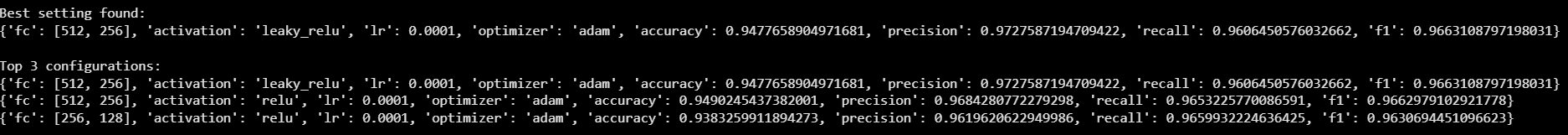
**Build the model using pretrained resnet18,** Pembuatan arsitektur Model transfer learning diamana model menggunakan pretrained resnet18, dengan konfigurasi dimana struktur dapat menerima variasi input Fully Connected layer, activation

**Define the training process,** Pembuatan skenario training dengan modifikasi bagian optimizer, dan epochs sehingga bisa menerima kombinasi hyperparameter nantinya.

**Define hyperparameter option and its loop for finding the best config**, proses looping dari proses train dengan kombunasi sebagai:

* Fully connected layer = [256], [512, 256], [256, 128]
* Fungsi aktivasi = relu, leaky\_relu
* Learning rate = 1e-3, 1e-4
* optimizer = adam, sgd
* epoch = 5 (dikarenakan limit komputasi epoch diturunkan untuk skenario looping)
* Batch size = 32

Dengan hasil kombinasi terbaik,



* Fully connected layer = [512, 256]
* Fungsi aktivasi = leaky\_relu
* Learning rate =1e-4
* optimizer = adamepoch = 5
* Batch size = 32

**Train model with best config**, melakukan training dengan penambahan epoch sebesar 10 sehingga sesuai dengan skenario scratch CNN, saving model dan test model performance, dengan hasil=

Accuracy : 0.9541

Precision : 0.9731

Recall : 0.9601

F1 Score : 0.9658

**Kesimpulan hasil,** Model dengan menggunakan transfer learning memiliki performa yang lebih baik, hal ini dikarenakan:

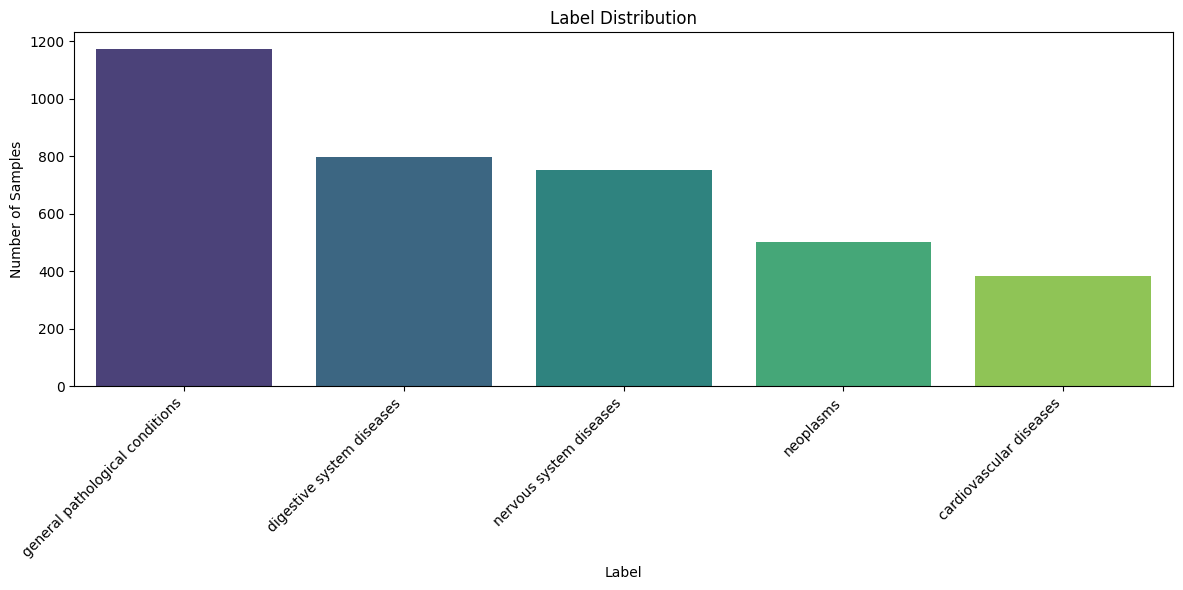
* Pre-trained Features, dimana pengunaan resnet18, memuat model lebih mengeneralisir proses identifikasi gambar
* Faster and More Stable Training, proses training dimulai dengan representasi fitur yang lebih kaya, sehingga proses training lebih efektif
* Perbedaan struktur, jumlah layer dan tingkat komputasi yang berbeda membuat transfer learning lebih unggul dalam banyak hal

2. Link Drive model = <https://drive.google.com/drive/folders/1vRjq0sRHr0cJCVWEGgC_72NhR3e1ubgx?usp=sharing>

# 2-a. Melakukan EDA pada teks, seperti memberikan informasi mengenai distribusi label, kata-kata yang sering muncul dalam data, dan lain-lain yang dapat Anda tambahkan.

EDA

* **Identifikasi data,** melihat struktur raw data
* **Visualisasi Distribusi penyebaran data,** Data memiliki sample sebesar 3609 entries, yang terdiri dari: general pathological conditions, 1172 sampel;digestive system diseases, 797 sampel; nervous system diseases, 753 sampel; neoplasms, 502 sampel; cardiovascular diseases, 385 sampel penyebaran kelas sesuai dengan soal.



* **Check word and character statistic,** melihat jumlah kata dalam 1 text dimana rata rata kata berkisar pada 100–250 kata
* **Word’s Frequency**, kata yang sering muncul adalah beberapa kata penghubung, dan kata yang berpaut pada data medis, seperti pasien.

# 2-b. Melakukan pre-processing, seperti cleaning, tokenization, atau stemming dan lemmatization sesuai dengan kebutuhan Anda dalam pemodelan.

Pre-Processing

* Drop column, hapus kolom awal karena hanya berisikan nomor urut data dan kurang informatif terhadap model
* Text Cleaning & Normalization, remove tanda baca, ekstra space dan lowercase
* Encode label, mengubah kolom label dari string ke integer kategorikal
* Splitting data
* Tokenization, untuk memecah kata ke dalam beberapa sub kata dengan direpresentasikan simbol angka/token, untuk melihat struktur kata
* Padding, untuk menyamakan setiap sequence kata guna input ke LSTM nantinya

2-c. Pengembangan model klasifikasi dengan menggunakan pendekatan LSTM dan LLM. Mohon untuk menjelaskan algoritma yang digunakan dalam pemodelan. Anda juga perlu melakukan hyperparameter tuning dalam mengembangkan model ini. Anda diberi kebebasan untuk memilih metode text representation untuk LSTM. Jelaskan perbandingan performa (accuracy, precision, recall, dan F1 score) dari LSTM dan LLM dalam model klasifikasi. Jelaskan analisa Anda.

## **LSTM**

Menggunakan tipe LSTM biderectional, dimana proses representasi kata dapat diliahat dari 2 arah depan - belakang dan belakang - depan,  
Melakukan deklarasi parameter seperti

* Vocab size: yang mennentukan jumlah kata unik
* Embedding dim: yang menentukan kedalaman dimensi suatu kata
* Max len: panjang maksimal suatu sequence
* Num classes: jumlah class atau label yang dimiliki

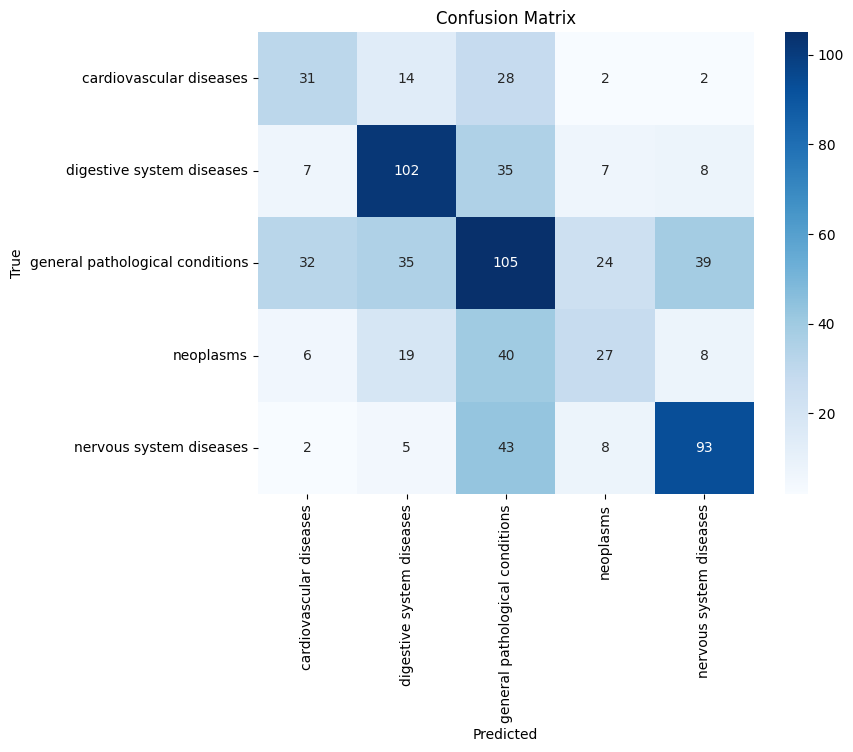
Penggunaan text representation word2vec

Kemudian semua hal tersebut diinput ke dalam pembuatan BiLSTM

Kemudian Model di evaluasi untuk melihat performancenya, hasil yang didapatkan adalah:

| Class | Precision | Recall | F1-score | Support |
| --- | --- | --- | --- | --- |
| Cardiovascular diseases | 0.15 | 0.03 | 0.04 | 77 |
| Digestive system diseases | 0.26 | 0.07 | 0.11 | 159 |
| General pathological conditions | 0.33 | 0.89 | 0.48 | 235 |
| Neoplasms | 0.50 | 0.01 | 0.02 | 100 |
| Nervous system diseases | 0.31 | 0.06 | 0.10 | 151 |

Dengan akurasi sebesar 32%



Dapat disimpulkan bahwa:

* penyakit general pathological conditions, terlalu general dikarenakan membenarkan semua input sampel dari setiap kelas
* Sedangkan kelas penyakit lain terbilang buruk performanya

Hal tersebut dapat dipicu oleh=

* Class imbalance
* Tingkat representatif word2vec yang rendah di bidang medis

## **Try some hyperparameter Tuning**

Melihat hal tersebut dilakukan tuning tanpa menggunakan pretrained text representation

Beberapa hyperparameter disesuaikan dengan pengaturan:

- LSTM units: 64

- Dropout: 0.4

- Dense units: 64

- Learning rate: 0.001

Kemudian hasil finetune BiLSTM, menunjukkan kenaikan performa sebagai berikut,

| Class | Precision | Recall | F1-score | Support |
| --- | --- | --- | --- | --- |
| Cardiovascular diseases | 0.46 | 0.36 | 0.41 | 77 |
| Digestive system diseases | 0.55 | 0.64 | 0.59 | 159 |
| General pathological conditions | 0.40 | 0.48 | 0.44 | 235 |
| Neoplasms | 0.44 | 0.36 | 0.40 | 100 |
| Nervous system diseases | 0.68 | 0.50 | 0.57 | 151 |

Dengan akurasi 49%, yang menandakan kenaikkan pada model dan tingkat performa yang seimbang pada beberapa kelas lain terutama digestive and nervous system diseases.

## LLM - BERT

Proses preparing data pada BERT persis seperti LSTM akan tetapi pada bagian tokenizer, proses ini menggunakan BERT tokenizer dimana proses tokenizer menggunakan desain bert dan tokenisasi suatu kata memiliki beberapa token khusus untuk merepresentasikan kata tersebut.

Hasil performance model adalah sebagai berikut,

| Class | Precision | Recall | F1-score | Support |
| --- | --- | --- | --- | --- |
| Cardiovascular diseases | 0.58 | 0.46 | 0.52 | 77 |
| Digestive system diseases | 0.69 | 0.88 | 0.77 | 159 |
| General pathological conditions | 0.47 | 0.58 | 0.52 | 235 |
| Neoplasms | 0.65 | 0.42 | 0.51 | 100 |
| Nervous system diseases | 0.67 | 0.39 | 0.49 | 151 |

Dengan tingkat akurasi sebesar 59%, hal ini membuktikan performa bert yang lebih unggul dibanding BiLSTM, beberapa hal yang dapat menjadi faktor pemicu, seperti:

* Pemahaman model pretrained BERT menghasilkan generalisasi yang lebih kuat
* BiLSTM mengalami kesulitan dengan adanya ketidakseimbangan kelas

Kemudian untuk memastikan hal tersebut, dilakukan juga hyperparameter tuning, dengan pengaturan

* Batch\_size : 32,
* epochs: 3,
* Learning rate: 3e-05

Dan menghasilkan performa sebagi berikut:

| Class | Precision | Recall | F1-score | Support |
| --- | --- | --- | --- | --- |
| Cardiovascular diseases | 0.56 | 0.62 | 0.59 | 77 |
| Digestive system diseases | 0.71 | 0.86 | 0.78 | 159 |
| General pathological conditions | 0.51 | 0.56 | 0.53 | 235 |
| Neoplasms | 0.61 | 0.49 | 0.54 | 100 |
| Nervous system diseases | 0.73 | 0.48 | 0.58 | 151 |

Dengan akurasi sebesar 62% yang mengungguli setiap model sebelumnya

Kesimpulan akhir:

* Kelas Digestive system diseases memiliki performa terbaik
* Nervous system diseases and general pathological conditions memiliki peningkatan dibanding performance model sebelumnya akan tetapi angka recall masih dapat dikatakan kecil
* Performa f1 score yang lebih seimbang di tiap kelas dibanding model sebelumnya

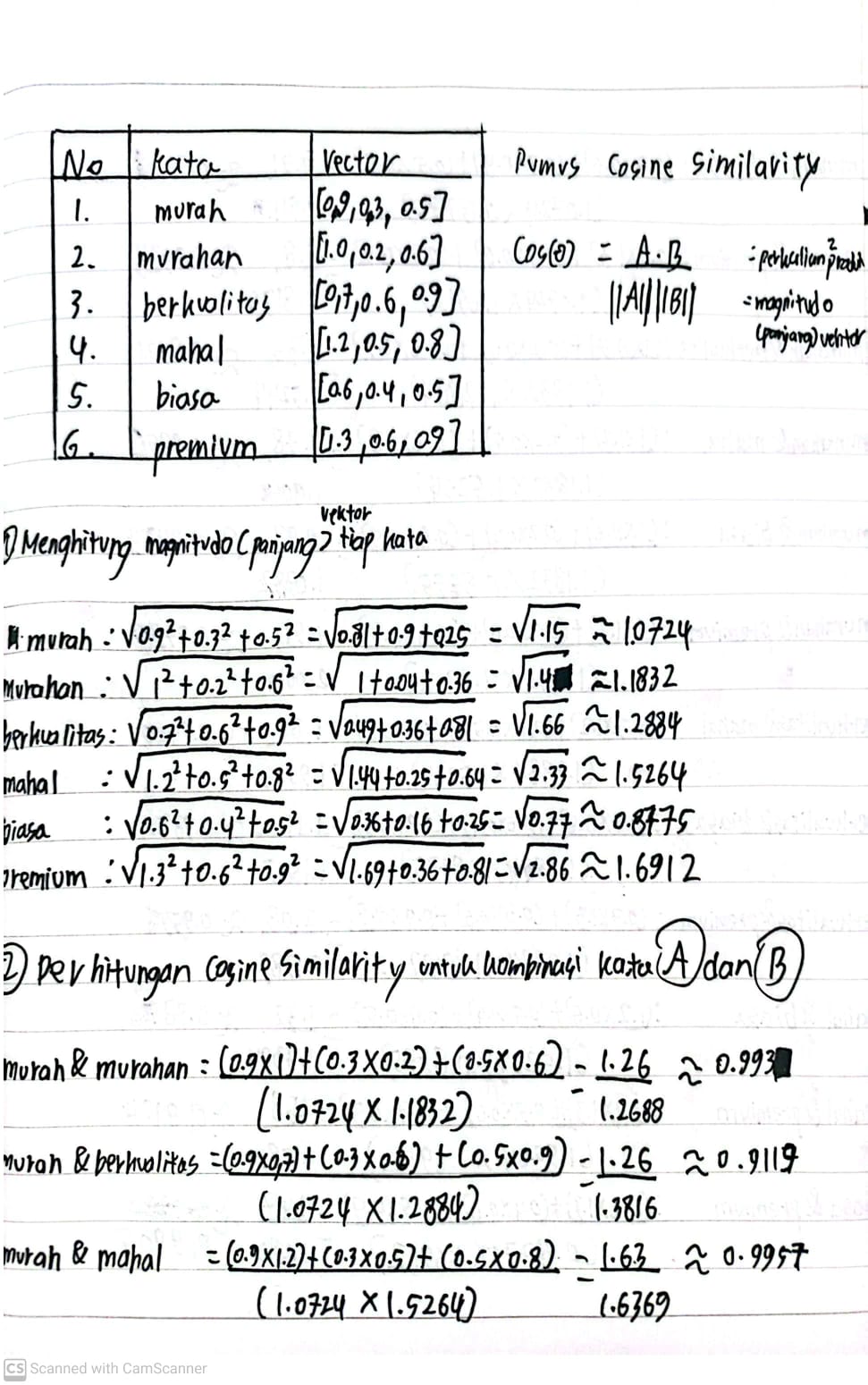
3.

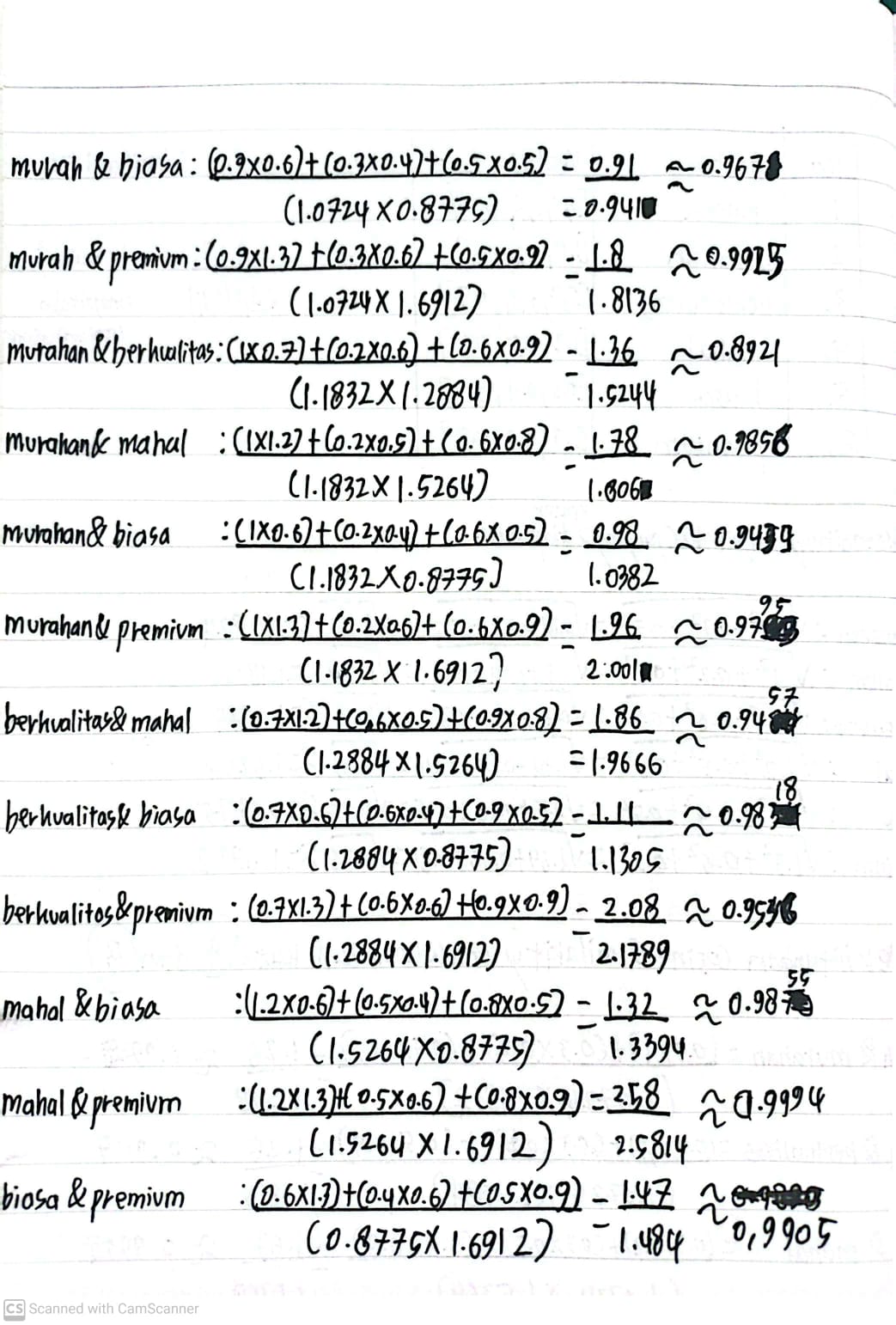
| 1 | murah | [0.9, 0.3, 0.5] |
| --- | --- | --- |
| 2 | murahan | [1.0, 0.2, 0.6] |
| 3 | berkualitas | [0.7, 0.6, 0.9] |
| 4 | mahal | [1.2, 0.5, 0.8] |
| 5 | biasa | [0.6, 0.4, 0.5] |
| 6 | premium | [1.3, 0.6, 0.9] |

# 3. Maka, lakukan analisis kata-kata mana saja yang memiliki kecenderungan makna yang berdekatan. Uraikan penjelasan Anda lengkap dengan keterhubungannya pada konsep-konsep dasar mengenai Word Embeddings!

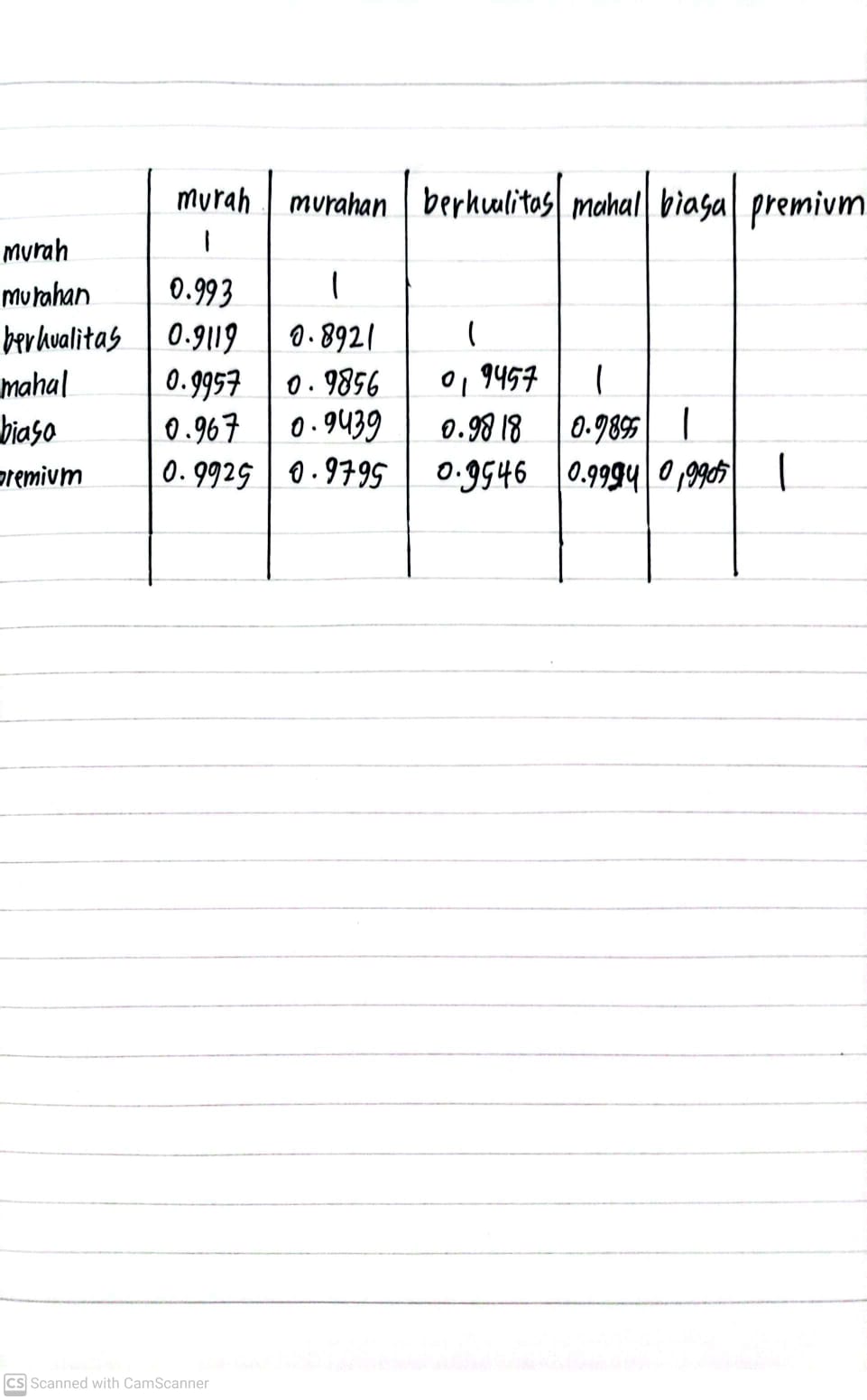
Untuk menilai suatu kesamaan atau jarak kesamaan antar kata secara objektif, kita perlu menggunakan metode umum yang digunakan untuk menghitung jarak kesamaan antar kata

Salah satunya adalah Formula cosine similarity, berikut adalah kalkulasi sistematis dari formula tersebut





Kemudian hasil kalkulasi tersebut dimasukan kedalam tabel matrik cosine similarity



Dari tabel diatas dapat disimpulkan bahwa

* Kata murah memiliki kedekatan dengan kata murahan,mahal,premium hal ini bisa saja terjadi karena konteks kata murah dan murahan yang memiliki 1 akar kata, dengan kata mahal yang menjadi antonim serta premium
* Kata murahan memiliki kedekatan dengan kata murah, sama seperti kesimpulan awal dimana kata tersebut merupakan sinonim dari kata murah
* Kata berkualitas memiliki kedekatan dengan kata biasa, hal ini mungkin terjadi dikarenakan faktor antonim sebelumnya
* Kata mahal memiliki kedekatan dengan kata murah dan premium, mahal dan premium sangatlah berkaitan jika melihat konteks kata dan kemunculan kedua kata tersebut dalam 1 sequence.
* Kata biasa memiliki kedekatan dengan kata premium dan berkualitas hal ini juga mungkin terjadi dikarenakan kata antonim
* Kata premium memiliki kedekatan dengan kata biasa dan mahal, biasa, dan murah hal tersebut juga mengacu pada penjelasan sebelumnya

Sebagai pembanding juga dilakukan pengukuran jarak kesamaan dengan metode euclidean distance (proses perhitungan distance dilakukan pada code no3 yang dilampirkan), dimana metode ini lebih mengutamakan selisih dari jarak antara 2 vektor, dan hasilnya adalah sebagai berikut

|  | murah | murahan | berkualitas | mahal | biasa | premium |
| --- | --- | --- | --- | --- | --- | --- |
| murah | 0 |  |  |  |  |  |
| murahan | 0.173 | 0 |  |  |  |  |
| berkualitas | 0.539 | 0.583 | 0 |  |  |  |
| mahal | 0.469 | 0.412 | 0.520 | 0 |  |  |
| biasa | 0.316 | 0.458 | 0.458 | 0.678 | 0 |  |
| premium | 0.640 | 0.583 | 0.600 | 0.173 | 0.831 | 0 |

Terdapat kesamaan intepretasi antara euclidean distance,yakni:

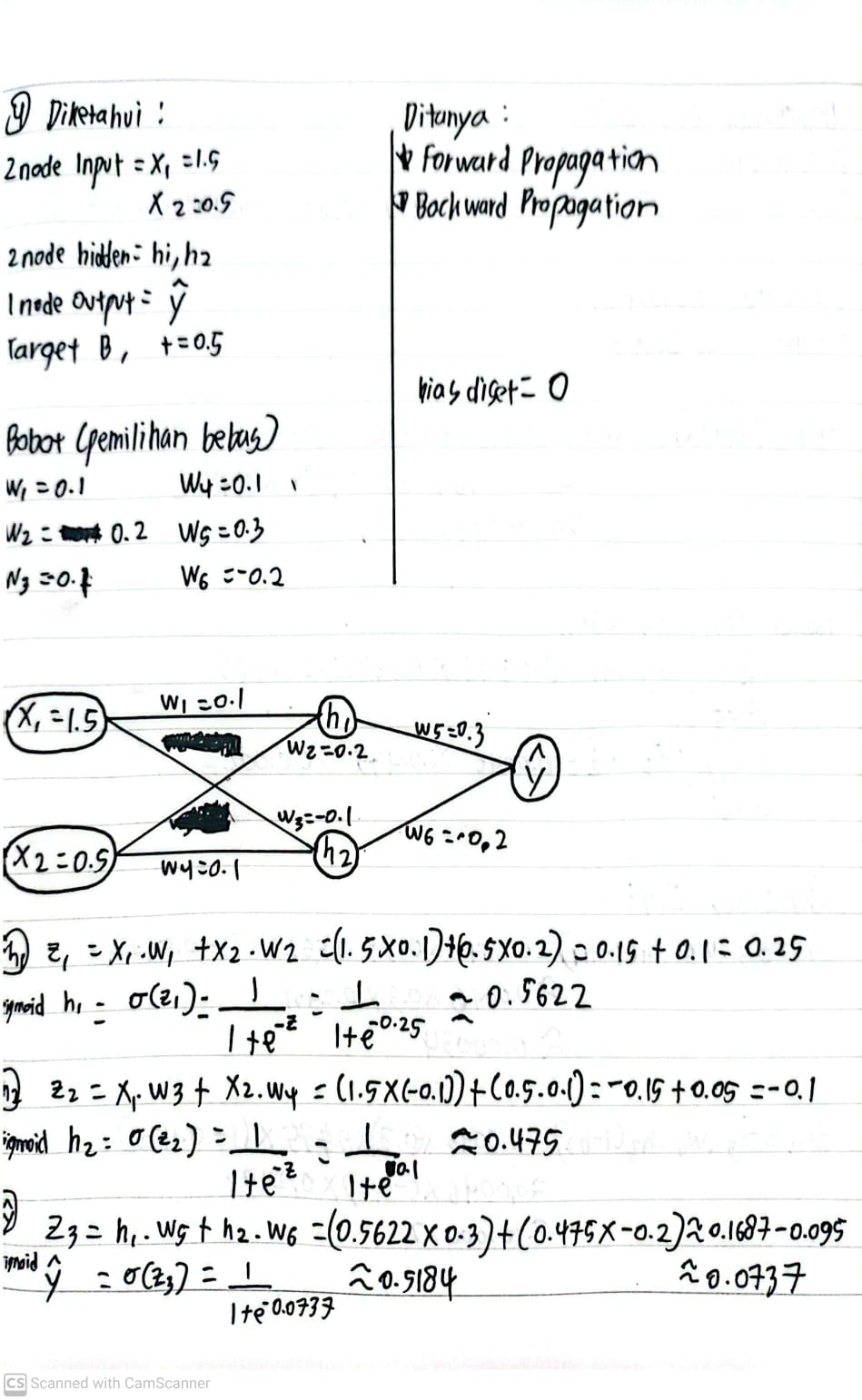
* Kata murah dengan kata murahan
* Kata mahal dengan kata premium

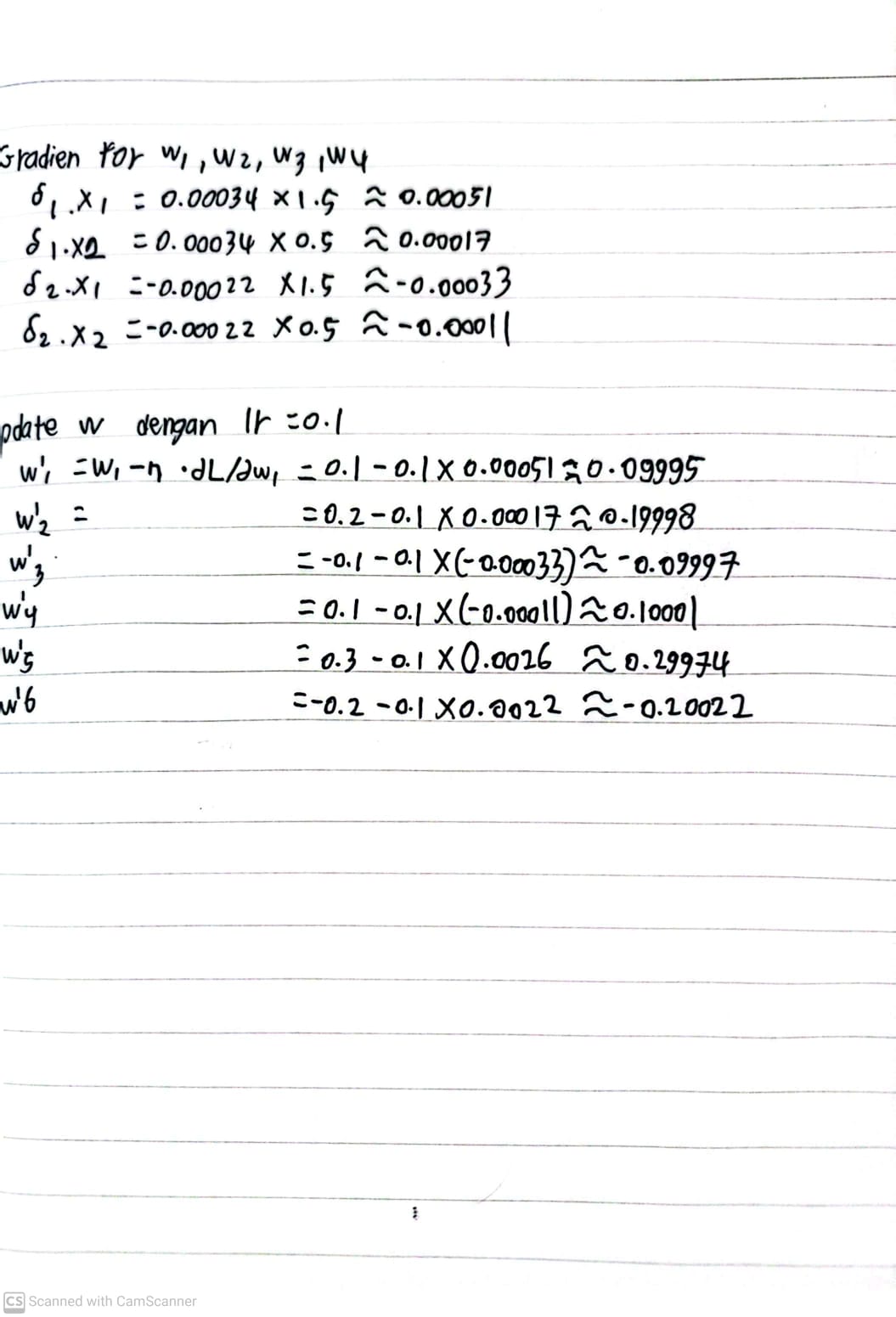
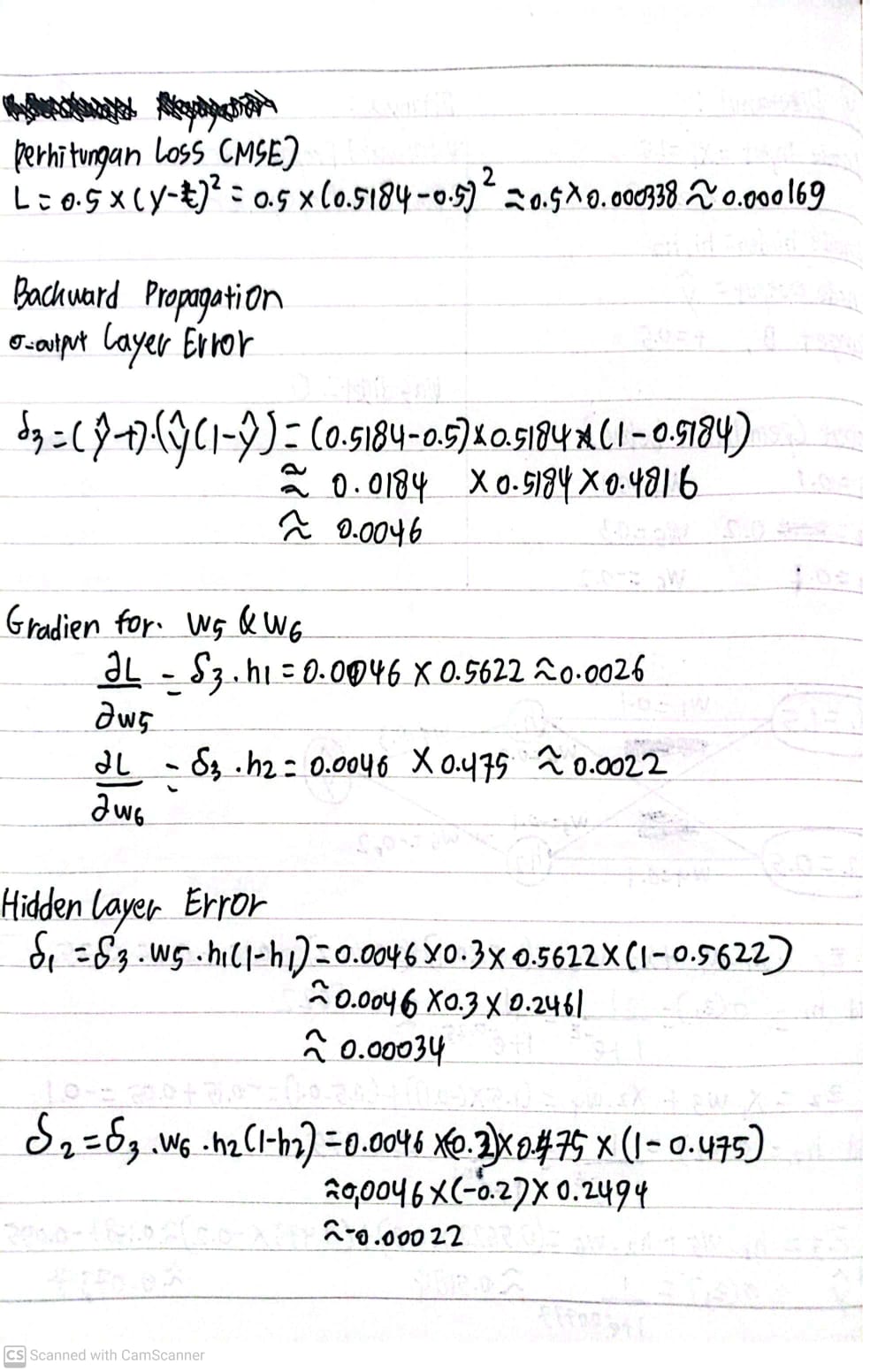
Sedangkan kata biasa dan berkualitas tidak memiliki jarak kedekatan yang signifikan

Kesimpulan akhir, kedua metode memiliki kelebihan dan ciri khasnya masing masing, metode euclidean distance lebih bergantung pada bobot value dari vektornya, sedangan metode cosine similarity, lebih dapat mengenaralisir suatu kata dengan konteks kata lainnya, seperti antonim, sinonim, dan subkata.

4.

# 4-a. Lakukan 1 kali forward propagation dan 1 kali backward propagation, dan bagaimana perubahan bobotnya. Apakah ANN sudah berhasil belajar, dan seberapa berhasil?





Updated weight

w1 = 0.09995

w2 = 0.19998

w3 = - 0.09997

w4 = 0.10001

w5 = 0.29974

w6 = - 0.20022

Hasil awal output dengan weight bebas menghasilkan besaran = 0.5184 (berdekatan dengan output target = 0.5)

Dalam perhitungan MSE diketahui besaran loss = 0.000169

Kedua hal tersebut merupakan poin positif yang menandakan hasil belajar bagus

# 4-b. Jika setelah proses di atas, output masih jauh dari target, apakah yang akan dilakukan?

• Mengubah learning rate?

• Mengubah fungsi aktivasi?

• Menambah jumlah node?

• Atau melakukan hal lain?

Jelaskan pilihan Anda berikut dengan penjelasannya!

Mengubah learning rate?

* Menaikan learning rate dapat menjadi salah satu opsi yang bagus, hal ini dipertimbangkan dengan melihat perubahan bobot yang tidak terlalu signifikan dari bobot awal

Mengubah fungsi aktivasi?

* Melihat hasil output yang sudah mendekati target, pergantian fungsi aktivasi belum perlu untuk dilakukan.

Menambah jumlah node?

* Sama dengan penjelasan sebelumnya dimana performa model sudah cukup baik sehingga perubahan tidak perlu dilakukan

5

# 5-a. Apakah representasi budaya dapat dilatih oleh model generatif tanpa kehilangan konteks makna? Jelaskan!

AI generatif merupakan teknologi yang tercipta dari adanya pembelajaran data historis sebelumnya, proses belajarnya adalah mengenal dan mengidentifikasi pola secara statistik, oleh karena itu proses generatif suatu gambar dalam model tersebut sangat dipengaruhi oleh dataset yang dipelajari juga, model sangat bergantung pada pola yang didapatkan dan dipelajari dari dataset tersebut, output yang dikeluarkan merupakan hasil representatif dari pola gambar yang dipelajari, hal ini menandakan konteks makna dalam suatu gambar belum tentu dapat dipelajari secara semantic oleh karena itu perlu adanya pengawasan dari manusia sendiri dalam menjamin proses pembuatan model generatif tersebut.

Beberapa hal yang perlu diawasi dalam pengembangan tersebut adalah: **Proses pengembangan Dataset yang terawasi,** hal ini mengacu pada proses persiapan dataset (Dataset-specific tuning, yang terdapat pada paper) yang akan dipakai sebagai bahan pembelajaran dari model yang ingin dibuat, salah satu hal yang bisa dilakukan adalah melakukan kurasi pada dataset tersebut seperti pada contoh “Gambar tokoh pewayangan yang telah dikurasi”, Perlu ada kerjasama antara si perancang model dan juga ahli budaya yang bersangkutan dengan tujuan pembuatan model tersebut; **Melakukan penyuluhan terkait model generatif tersebut,** hal ini mengacu pada proses edukasi teknologi generatif tersebut kepada para pengguna teknologi ini nantinya, perlu adanya pemahaman dimana setiap teknologi masih belajar berdasarkan identifikasi pola dan belum sepenuhnya memiliki pemikiran dan representasi seperti manusia merepresentasi suatu hal, Hal ini diperlukan agar setiap pengguna juga bijak dalam penggunaan teknologi tersebut; **Keterlibatan ahli budaya,** Untuk memastikan bahwa model melakukan tugas sesuai dengan tujuan yang ditetapkan, perlu adanya suatu ahli untuk mengkonfirmasi hasil tersebut, bukan hanya pada output terakhir akan tetapi juga pada prosesnya.

# 5-b. *Apakah GAN berisiko menghasilkan “distorsi budaya”?*

Berdasarkan acuan model dimana digunakan “StyleGAN2 pretrained”, menandakan proses pembuatan model nantinya menggunakan model yang telah dilatih sebelumnya pada dataset yang berbeda dengan tujuan akhir model, hal ini menandakan distorsi bisa saja terjadi, jika dataset awal yang digunakan pada proses pelatihan pretrained model tersebut berbeda dengan dataset pewayangan yang nantinya digunakan.

Kemudian berdasarkan catatan tantangan dari proses pengembangan model tersebut “wajah dan elemen visual kadang bercampur, hasil tidak konsisten”, menandakan bahwa distorsi dapat saja terjadi sewaktu waktu jika proses pengemabangan ini tidak dipahami dan diawasi secara seksama, beberapa solusi yang dijelaskan pada pertanyaan sebelumnya, dapat menjadi salah satu solusi untuk mengatasi resiko tersebut.

# Bagaimana Anda menyikapi isu ini secara etis?

Budaya dapat terbentuk dari proses akulturasi, dan asimilasi, oleh karena itu hal ini sangat bergantung pada bagaimana cara masyarakat mengantisipasi atau merespon pada hal ini. Sedangkan di lain sisi, kemajuan dan perkembangan digital tentu tidak dapat dihindari melihat perkembangan teknologi yang terus berevolusi secara terus menerus, hal ini berlaku tanpa terkecuali bagi setiap generasi baru yang terlahir di era teknologi ini. Perlu ada keterbukaan antara generasi sebelumnya dan generasi penerus ini, sehingga dapat terjadi kerja sama sehingga generasi sebelumnya dapat mulai memahami dan menggunakan teknologi terkini dengan mempertimbangakn beberapa aspek resiko yang sudah dapat mereka seleksi berdasarkan pengalamannya dan pemahaman budayanya, setelah pemahaman tersebut didapatkan maka hal tersebut dapat dikembangkan untuk diterapkan dan diadaptasikan terhadap teknologi masa kini sebagai salah satu jalan memperkenalkan budaya kepada generasi penerus yang sudah terpaut pada teknologi ini, sehingga proses akulturasi dan asimilasi budaya ini dapat dicapai sesuai dengan kesepakatan dan norma yang diyakini.