News Text Classification using LSTM

Anggota Kelompok:

Andrea Octaviani - 2602075895

Cristian Agusta - 2602157705

Moh. Khoirul Umam Al Amin - 2602198472

1. Pendahuluan

Di era big data ini, kemajuan pesat dalam daya komputasi dan teknologi Internet telah mendorong kemunculan berbagai inovasi baru terkait bidang informasi dan teknologi, yang secara mendalam mempengaruhi kehidupan kita. Internet telah menjadi bagian tak terpisahkan dari kehidupan sehari-hari, di mana kita mencari atau menyampaikan informasi setiap hari. Akibatnya, volume data yang tersedia di Internet telah tumbuh secara eksponensial. Data tersebut hadir dan tersedia dalam berbagai bentuk, termasuk teks, audio, dan gambar. Informasi tekstual sendiri memiliki berbagai bentuk misalnya, melalui blog, unggahan di forum, berita, dan email.

Text classification merupakan salah satu task yang sangat sering dilakukan dalam konteks *Natural Language Processing* (NLP) karena juga dapat membantu mengolah volume informasi yang sangat besar. Text classification juga memiliki peranan yang penting khususnya dalam konteks news classification. News classification tidak hanya membantu pengguna untuk menemukan informasi yang sesuai dengan minat mereka. Selain itu, news classification juga dapat membantu meningkatkan efisiensi dalam pengelolaan informasi berita. Dengan adanya news classification ini dapat membantu efisiensi dan meningkatkan beberapa task, misalnya dalam melakukan proses otomatisasi pengelompokan berita ke dalam kategori (kelas) yang sesuai dapat meningkatkan personalisasi konten, mempercepat waktu pencarian informasi.

Sejumlah penelitian mengenai text classification telah dilakukan untuk meningkatkan metode classification dengan berbagai metode. Pada awalnya, metode berbasis ekstraksi fitur manual seperti Term Frequency-Inverse Document Frequency (TF-IDF) dan bag-of-words digunakan secara luas untuk merepresentasikan data teks. Pendekatan ini, meskipun sederhana, memberikan kuat untuk model klasifikasi tradisional seperti Naive Bayes, Support Vector Machines (SVM), dan Logistic Regression. Kemudian dengan berkembangnya teknologi, penelitian kemudian beralih ke metode neural network (deep learning) yang memanfaatkan representasi kata yang lebih kompleks seperti Word Embeddings dan menggunakan model seperti Convolutional Neural Networks (CNN) dan Recurrent Neural Networks (RNN), termasuk Long Short-Term Memory (LSTM).

2. Metodologi

2.1 Metode Penyelesaian Masalah

Proses penyelesaian masalah dalam project ini dapat digambarkan melalui diagram pada fig.2.1 dibawah.

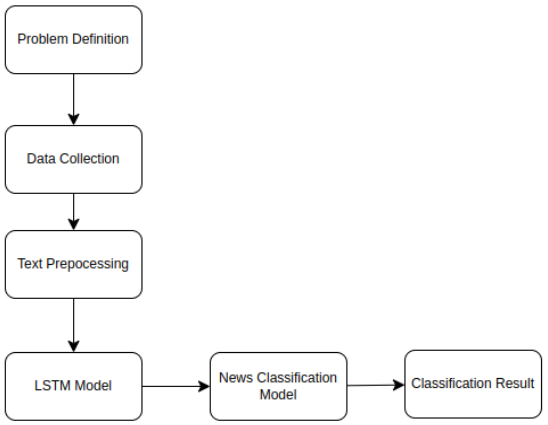


Fig.2.1. Problem Solving Methods

Langkah awal dalam proses ini adalah mendefinisikan masalah secara jelas. Pada tahap ini, tujuan utama dari penelitian dirumuskan, yaitu mengembangkan model klasifikasi berita yang mampu mengelompokkan artikel ke dalam kategori yang sesuai. Definisi masalah disini mencakup mulai dari problem klasifikasi apa yang akan diselesaikan, dataset yang akan digunakan, dan model evaluation apa yang akan digunakan nantinya. Setelah masalah didefinisikan, langkah berikutnya adalah pengumpulan data. Project ini menggunakan secondary dataset yang kami dapatkan melalui kaggle, yaitu **news category dataset.**

Tahapan selanjutnya adalah text preprocessing yang dimana tahapan ini bertujuan untuk mengolah data text menjadi data yang lebih siap untuk digunakan. Pada tahapan ini melibatkan beberapa langkah, seperti :

- Case folding
- Stopword removal
- Remove punctuation
- Lemmatization
- Tokenization dan Word embedding
- Encoding

Kemudian model berbasis Long Short-Term Memory (LSTM) diterapkan untuk menangani data teks. LSTM, yang merupakan jenis Recurrent Neural Network (RNN), dirancang untuk mempelajari data yang recurrent seperti text, sehingga cocok untuk menangkap hubungan antar kata dalam sebuah kalimat atau dokumen. Model ini dilatih menggunakan data yang telah diproses sebelumnya, dengan tujuan untuk menghasilkan representasi fitur yang lebih dalam dari teks. Setelah tahapan modeling akan dihasilkan model klasifikasi yang bertugas untuk mengelompokkan teks news ke dalam kategori tertentu berdasarkan pola yang telah dipelajari selama proses training.

Tahapan terakhir adalah evaluasi model dengan tujuan untuk memastikan bahwa model dapat memprediksi kategori dengan akurat dan andal. Salah satu langkah penting dalam evaluasi adalah pembagian dataset menjadi train, test, dan validation sets. Selanjutnya, evaluasi dapat dilakukan menggunakan beberapa matriks seperti accuracy, recall, precision, dan f1-score

2.2 Dataset dan Task

Dataset yang digunakan pada project ini terdiri dari enam variabel atau feature sebagai berikut :

- category: category in which the article was published.
- headline: the headline of the news article.
- authors: list of authors who contributed to the article.
- link: link to the original news article.
- short_description: Abstract of the news article.
- date: publication date of the article.

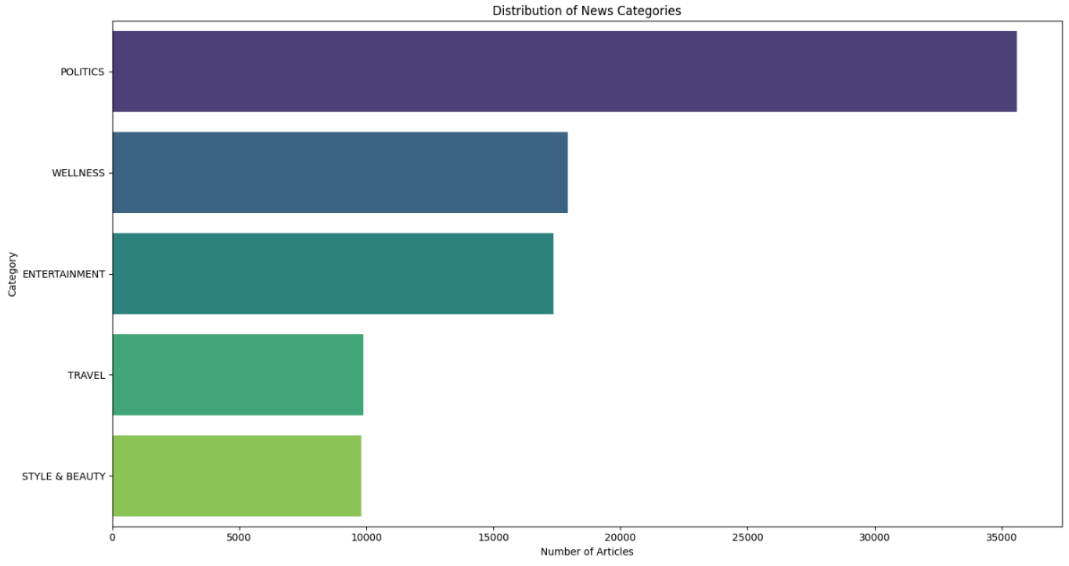
	link	headline	category	short_description	authors	date
135121	https://www.huffingtonpost.com/entry/supermode...	Supermodels Kate Moss And Cara Delevingne In R...	STYLE & BEAUTY	Altogether, Wybrun used more Twiglets, Cream C...	Carly Ledbetter	2014-04-02
90532	https://www.huffingtonpost.com/entry/every-tea...	Here's Every Teacher You Wish You Never Had	COMEDY	You'll want to give them all detention.	Ron Dicker	2015-08-26
9992	https://www.huffingtonpost.com/entry/scooters-...	Scooters, Resisted Elsewhere, Face Less Skepti...	POLITICS	Electric scooters have received pushback in so...	Stateline, Editorial Partner	2018-05-01
113056	https://www.huffingtonpost.com/entry/worlds-la...	These Moms Are Helping Knit The World's Larges...	IMPACT		Alena Hall	2014-12-11
144950	https://www.huffingtonpost.com/entry/our-girl-...	Our Girl in Havana: The Artists of Camagüey	TRAVEL	Some people assume the strictures of the socia...	Tyler Wetherall, Contributor\nInsight Cuba, Tr...	2013-12-17
179959	https://www.huffingtonpost.comhttp://www.paren...	19 Things I Will Miss Most About Pregnancy	PARENTING	36w, 6d. Tomorrow marks full-term for this bab...		2012-12-12
195896	https://www.huffingtonpost.com/entry/italian-c...	Drink Like An Italian	FOOD & DRINK	From pasta and pizza to tiramisu, Americans ha...	Liquor.com, Contributor\nLiquor.com	2012-06-22
48384	https://www.huffingtonpost.com/entry/trumps-in...	Trump's Inauguration, the Musicians and the Bo...	RELIGION		Rabbi Jason Miller, ContributorEntrepreneur, E...	2016-12-17
121134	https://www.huffingtonpost.com/entry/sons-guns...	'Sons Of Guns' Star Charged With Rape Of Anoth...	CRIME		Andres Jauregui	2014-09-09
160142	https://www.huffingtonpost.comhttp://www.refin...	All You Need To Know About Natural Makeup	STYLE & BEAUTY	Last week we showed you the down and dirty dee...		2013-07-09

Dalam proyek ini, dua jenis data teks yang berbeda digunakan sebagai input untuk proses preprocessing dan training model. Data yang dimaksud meliputi *headline* dan *category*, yang keduanya akan diproses secara terpisah dan kemudian dibandingkan kinerjanya dalam

konteks training model klasifikasi. Secara teknis, project ini hanya menggunakan tiga fitur atau variabel utama, yaitu *short_description*, *headline*, dan *category*, yang dipilih dengan pertimbangan relevansi masing-masing terhadap tujuan klasifikasi dengan *category* akan digunakan sebagai target variabel. Tujuan dari eksperimen ini adalah untuk membandingkan bagaimana performa model berubah ketika berbagai jenis data teks (headline vs. category) digunakan sebagai input dalam proses pelatihan.

	headline	category	short_description
20	Golden Globes Returning To NBC In January Afte...	ENTERTAINMENT	For the past 18 months, Hollywood has effectiv...
21	Biden Says U.S. Forces Would Defend Taiwan If ...	POLITICS	President issues vow as tensions with China rise.
24	'Beautiful And Sad At The Same Time': Ukrainia...	POLITICS	An annual celebration took on a different feel...
28	James Cameron Says He 'Clashed' With Studio Be...	ENTERTAINMENT	The "Avatar" director said aspects of his 2009...
30	Biden Says Queen's Death Left 'Giant Hole' For...	POLITICS	U.S. President Joe Biden, in London for the fu...

Tujuan dari proyek ini adalah untuk melakukan klasifikasi teks dengan membatasi target kelas pada lima kategori utama(multiclass classification), yaitu ‘ENTERTAINMENT' , 'POLITICS' , 'STYLE & BEAUTY' , 'TRAVEL' , dan 'WELLNESS' . Pemilihan kelima kelas ini didasarkan pada distribusi frekuensi yang paling tinggi dalam dataset, dengan fokus pada kategori yang paling banyak ditemui dalam data.



Dari distribusi setiap kelas dapat dilihat bahwasannya kondisi data imbalance, yang dimana kondisi ini dapat menyebabkan model bias terhadap suatu kelas karena perbedaan jumlah data dari setiap kelas. Untuk mengatasi masalah ini dilakukan downsampling, yaitu teknik dalam pemrosesan data dengan mengurangi jumlah data dari kelas yang lebih banyak (dominan) untuk menyeimbangkan distribusi kelas dalam dataset. Tujuannya adalah agar

model tidak terlatih untuk lebih memprioritaskan kelas dengan jumlah data yang lebih banyak, yang bisa menyebabkan bias dalam classification nantinya. Dalam konteks ini, downsampling dilakukan pada kelas dengan distribusi data paling banyak (kelas dominan) untuk mengurangi jumlahnya, sehingga kelas yang lebih sedikit (minoritas) memiliki proporsi yang lebih seimbang. Dalam case ini kelas yang paling rendah adalah 'STYLE & BEAUTY', maka angka downsampling akan disesuaikan berdasarkan kelas 'STYLE & BEAUTY'. Artinya akan dilakukan pengurangan jumlah sampel dari kelas lainnya (yang memiliki jumlah lebih banyak) agar jumlah sampelnya setara dengan kelas 'STYLE & BEAUTY'.

```
Will downsample all categories to 9814 samples

Original distribution:
category
POLITICS          35602
WELLNESS          17945
ENTERTAINMENT     17362
TRAVEL             9900
STYLE & BEAUTY     9814
Name: count, dtype: int64

Downsampled distribution:
category
ENTERTAINMENT     9814
POLITICS          9814
WELLNESS          9814
STYLE & BEAUTY    9814
TRAVEL            9814
Name: count, dtype: int64

Original dataset size: 90623
Downsampled dataset size: 49070
```

Setelah dilakukan downsample dapat dilihat bahwasannya setiap kelas memiliki jumlah data yang sama banyaknya. Kemudian jumlah total dataset yang sebelumnya berada di **90623** turun menjadi **49070**.

2.3 Text Preprocessing

Text preprocessing adalah tahap awal yang penting dalam pemrosesan data teks, di mana data mentah dibersihkan dan dipersiapkan agar sesuai untuk pelatihan model. Pada tahapan ini dilakukan beberapa tahapan teknik pemrosesan text menggunakan library spaCy. Pertama, dalam tahap ini dilakukan metode case folding.

Dalam metode case folding teks diubah menjadi huruf kecil (lowercase) untuk memastikan konsistensi dalam pengolahan kata. Selanjutnya dilakukan remove punctuation, yaitu semua tanda baca dihapus juga menghapus karakter-karakter yang tidak relevan, seperti titik, koma, tanda tanya, dan sebagainya. Kemudian juga dilakukan metode remove whitespace yang berfungsi untuk menghapus spasi ekstra dalam teks dengan cara memadatkan spasi lebih dari satu menjadi satu spasi. Proses selanjutnya adalah menghapus **stopwords** menggunakan spaCy, yang mengandalkan daftar kata-kata umum (seperti "the",

"and", "is") yang tidak membawa informasi penting dalam analisis teks. Metode ini bertujuan agar memfokuskan analisis pada kata-kata yang lebih bermakna. Kemudian pada tahapan ini juga dilakukan lemmatization, yaitu teks diubah ke bentuk dasar (lemma) dari setiap kata menggunakan model linguistik spaCy Proses lemmatization mengurangi variasi kata yang berasal dari infleksi (seperti "running" menjadi "run" atau "better" menjadi "good").

Data text yang sudah dilakukan preprocessing selanjutnya di tokenize, yaitu teks dipecah menjadi token(kata atau sub kata). Pada project ini dilakukan tokenization dan word embedding menggunakan BertTokenizer. Jadi sederhananya setelah dilakukan tokenization akan dilakukan word embedding atau sederhanya diberikan representasi numerik atau vektor oleh Bert. Bert disini dapat memberikan representasi semantic dari kata bukan hanya suatu nilai vektor yang tidak menberikan representasi kontekstual seperti TF-IDF. Saat melakukan tokenization BERT menggunakan pendekatan *WordPiece* untuk memecah kata menjadi sub-kata yang lebih kecil jika kata tersebut tidak ada dalam kamus model. Misalnya, kata "unhappiness" bisa dipecah menjadi "un", "happi", dan "ness", sehingga model bisa memahami bagian-bagian dari kata tersebut. Perbedaan dengan metode word embedding seperti Word2vec adalah jika pada yang menghasilkan representasi tetap untuk setiap kata (misalnya, kata "bank" selalu memiliki representasi yang sama), BERT menghasilkan representasi yang dinamis. Artinya, kata yang sama akan memiliki representasi yang berbeda tergantung pada kata-kata lain disekitarnya dalam kalimat. Ini dilakukan dengan menggunakan arsitektur Transformer yang didalamnya menerapkan self attention ang memungkinkan model untuk memperhatikan kata-kata atau token lain di dalam kalimat yang dapat mempengaruhi pemahaman kata tertentu. Dalam Project ini inisialisasi Bert Tokenizer dengan menggunakan pretrained *'bert-base-uncased'*. Kemudian dalam proses tokenizer dilakukan mekanisme padding agar semua teks memiliki panjang yang sama

Dalam library bert tokenizer dia akan mengubah jadi id tokens yang sesuai dengan vocab(model bert), ID token ini adalah representasi numerik dari kata-kata dalam kalimat yang digunakan oleh model untuk memproses informasi. Kemudian, fungsi mengubah ID token tersebut kembali menjadi bentuk yang lebih mudah dipahami oleh manusia, yaitu token berupa kata atau sub-kata. Selain itu, BERT menggunakan teknik *subword tokenization* untuk menangani kata-kata yang tidak ada dalam kosakata model, dengan memecahnya menjadi token yang lebih kecil.

```
Training shapes:
Input IDs: (39256, 51)
Attention Mask: (39256, 51)
Labels: (39256,)

Testing shapes:
Input IDs: (9814, 51)
Attention Mask: (9814, 51)
Labels: (9814,)
```

Apabila kita lihat di codenya kita bisa liat ada Input Id dan Attention Mask yang dihasilkan dari function *enode_plus()* pada *BertTokenizer* yang dimana *input_ids* berisi Id token dan *attention_mask* menandakan token mana yang diproses atau tidak (attention mechanism).

2.3 Model dan Model Training

Pada project ini digunakan model Long Short Term Memory (LSTM) yang merupakan salah satu jenis arsitektur yang sebenarnya dirancang untuk mengatasi kelemahan dari Recurrent Neural Network (RNN), yaitu masalah **vanishing gradient** dan **exploding gradient** yang terjadi karena sequence data yang terlalu panjang yang menyebabkan kehilangan informasi (dalam konteks text). LSTM mengatasi masalah ini dengan menggunakan struktur sel memori yang memungkinkan informasi untuk dipertahankan dalam jangka panjang. Dengan memanfaatkan **gates** (input gate, forget gate, dan output gate), LSTM dapat memutuskan apakah informasi tertentu harus dipertahankan atau dibuang.

Dalam konteks pengolahan teks, LSTM sangat berguna karena mampu memahami konteks panjang dalam kalimat atau dokumen. Model ini dapat mengingat informasi penting dari kata-kata atau frasa sebelumnya dan menggunakan informasi tersebut.

Model LSTM pada project ini digunakan untuk multiclass classification

Layer (type)	Output Shape	Param #	Connected to
input_layer (InputLayer)	(None, 159)	0	-
embedding (Embedding)	(None, 159, 128)	3,906,816	input_layer[0][0]
not_equal (NotEqual)	(None, 159)	0	input_layer[0][0]
lstm (LSTM)	(None, 128)	131,584	embedding[0][0], not_equal[0][0]
dropout (Dropout)	(None, 128)	0	lstm[0][0]
dense (Dense)	(None, 5)	645	dropout[0][0]

Total params: 4,039,045 (15.41 MB)
Trainable params: 4,039,045 (15.41 MB)
Non-trainable params: 0 (0.00 B)

Proses dimulai dengan mendefinisikan **input layer** yang menerima urutan token dari teks yang telah diproses menjadi representasi numerik melalui BertTokenizer. Kemudian selanjutnya lapisan **embedding** mengubah representasi integer ini menjadi vektor berdimensi

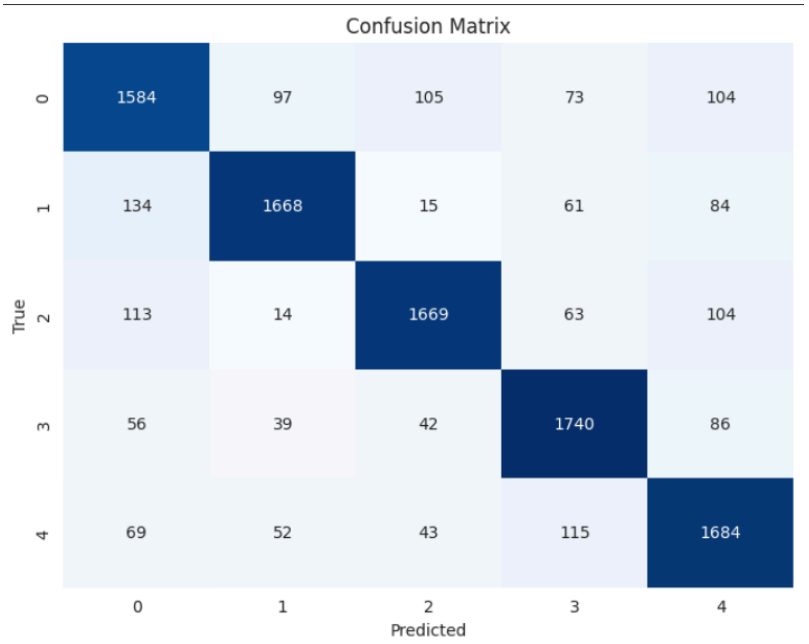
tetap, dalam hal ini vektor berdimensi 128. Lapisan embedding ini bertujuan untuk menangkap representasi semantik kata yang lebih padat sehingga kata-kata yang memiliki makna serupa dapat didekatkan dalam ruang vektor. Setelah itu, **lapisan LSTM** digunakan untuk memproses urutan kata tersebut. LSTM memiliki keunggulan dalam mempelajari ketergantungan jangka panjang dalam data urutan, seperti teks, yang mana informasi yang jauh dalam urutan tetap bisa dipertahankan dan digunakan untuk menghasilkan klasifikasi. Untuk mencegah model terlalu mengandalkan data pelatihan dan mengalami **overfitting**, lapisan **dropout** diterapkan dengan tingkat 20%. Ini membantu memastikan bahwa model belajar untuk menggeneralisasi dengan baik dan tidak terlalu "menghafal" data pelatihan. Kemudian, model diakhiri dengan **lapisan output** berupa lapisan Dense dengan fungsi aktivasi **softmax**, yang akan menghasilkan probabilitas untuk setiap kelas target. Model ini dioptimalkan menggunakan **sparse categorical cross entropy** sebagai fungsi loss, yang merupakan pilihan umum untuk masalah klasifikasi multi-kelas dimana label target adalah integer, serta **Adam optimizer** yang terkenal efektif dalam pengaturan parameter model.Untuk lebih lanjut memastikan pelatihan yang efisien, **EarlyStopping** callback diterapkan. Ini memungkinkan pelatihan dihentikan lebih awal jika model tidak menunjukkan peningkatan yang signifikan pada **validation loss** selama beberapa epoch berturut-turut, dengan tujuan untuk menghindari pelatihan yang tidak perlu dan overfitting.

3. Hasil dan Analisa

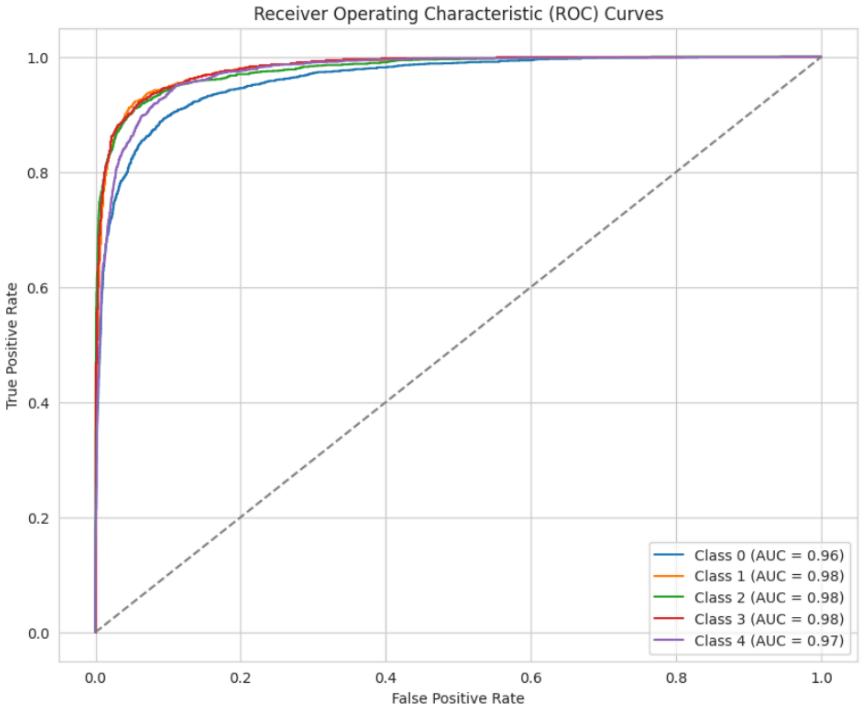
Dataset	Accuracy	Loss	Val_Accuracy	Val_Loss
Headline	0.97	0.09	0.84	0.6
short_description	0.91	0.22	0.72	0.96

Melalui data pada tabel, dapat terlihat bahwa ketika model dilatih dengan data “headline”, memperoleh akurasi sebesar 0.97, akurasi validasi sebesar 0.84, loss sebesar 0.09, dan loss validasi sebesar 0.6. Sementara ketika model dilatih dengan data “short_description”, memperoleh akurasi sebesar 0.91, akurasi validasi sebesar 0.72, loss sebesar 0.22, dan loss validasi sebesar 0.96. Dapat dilihat bahwa, nilai akurasi antara kedua model cukup mirip namun terdapat perbedaan yang cukup signifikan akan nilai loss. Hal ini dapat disebabkan karena sequence pada data short_description yang lebih panjang ditambah token yang

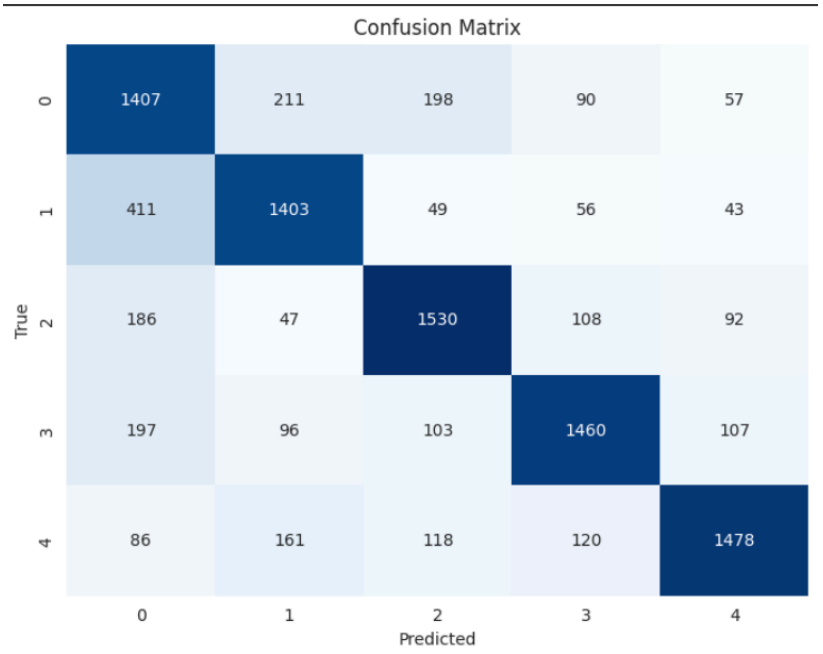
digunakan tidak secara langsung merujuk pada label yang bersangkutan mempengaruhi kemampuan model dalam mempelajari pola yang membedakan satu label dengan lainnya.



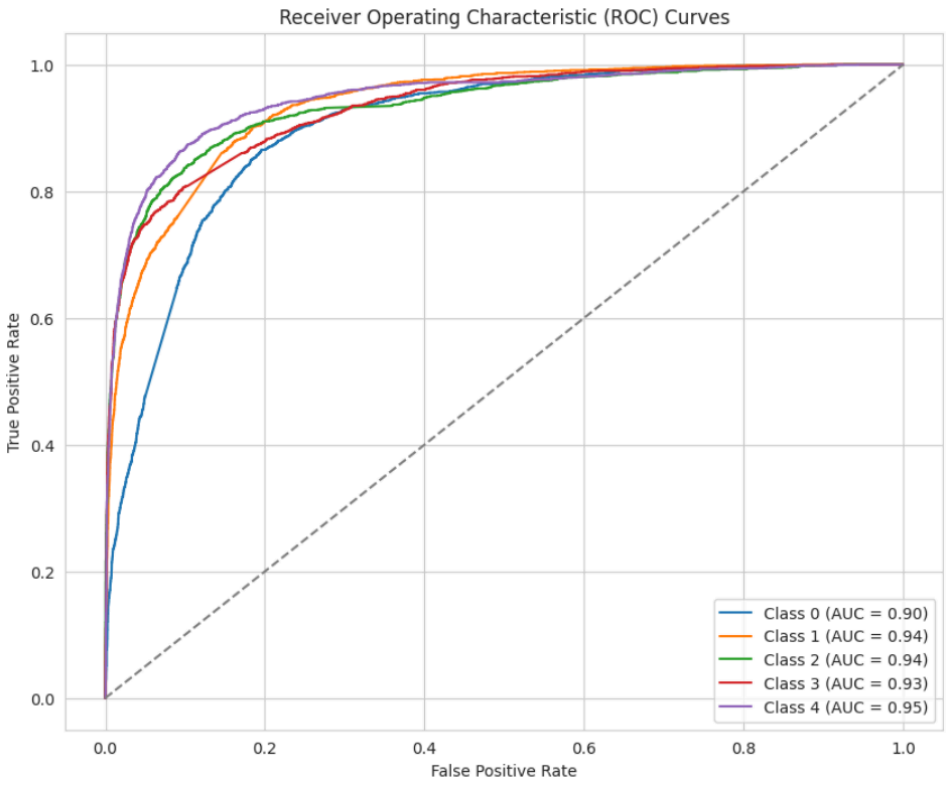
Gambar di atas merupakan confusion matrix ketika model dilatih dengan data “headline”. Dapat terlihat bahwa rate antara true answer dan predicted answer yang sama jauh lebih tinggi, sehingga persentase model dalam memprediksi sesuai dengan kelasnya lebih tinggi.



Gambar di atas merupakan kurva ROC AUC untuk model dengan dataset “headline”. Dapat terlihat bahwa nilai AUC untuk semua kelas berada di kisaran 0.97, angka yang sangat tinggi menggambarkan bahwa model telah bekerja dengan optimal.



Gambar di atas merupakan confusion matrix ketika model dilatih dengan data “short_description”. Dapat terlihat bahwa rate antara true answer dan predicted answer yang sama cukup tinggi, namun total keseluruhan dari jumlah true answer dan predicted answer yang sama, lebih rendah dibandingkan dengan model yang dilatih data “headline”.



Gambar di atas merupakan kurva ROC AUC untuk model dengan dataset "short_description". Dapat terlihat bahwa nilai AUC untuk semua kelas berada di kisaran 0.93, angka yang cukup tinggi menggambarkan bahwa model telah bekerja dengan cukup optimal, namun nilai keseluruhannya lebih rendah dibandingkan dengan data "headline" menandakan bahwa model dengan data "headline" bekerja dengan lebih optimal.

4. Kesimpulan

Kesimpulan yang didapatkan berdasarkan hasil percobaan yang dilakukan terhadap kolom yang telah ditentukan, yaitu *headline* dan *short_description*, ditemukan bahwa model menunjukkan akurasi yang lebih tinggi dalam melakukan prediksi pada data *headline* dibandingkan data *short_description*. Hasil ini mengindikasikan bahwa kolom *headline* lebih efektif untuk klasifikasi teks dalam dataset yang kami pilih. Perbedaan akurasi ini dapat dipengaruhi oleh beberapa faktor. Pertama panjang sequence pada kolom *headline* cenderung lebih pendek, yang memungkinkan model untuk lebih cepat dan efisien dalam memproses informasi. Kedua, kata-kata yang ada di kolom *headline* lebih fokus dan relevan terhadap label yang ingin diprediksi, sehingga model lebih mudah mengenali pola dan konteks yang berkaitan dengan kategori berita. Sedangkan, kolom *short_description* mengandung

informasi yang lebih panjang dan kompleks, yang dapat menyebabkan model lebih sulit untuk memproses data dan mengurangi akurasi prediksi.

5. Referensi

1. Misra, Rishabh. "News Category Dataset." arXiv preprint arXiv:2209.11429 (2022).
2. Liu, C. (2024). Long short-term memory (LSTM)-based news classification model. *Plos one*, *19*(5), e0301835. https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0301835
3. Zhang, Y. (2021, April). Research on text classification method based on LSTM neural network model. In *2021 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)* (pp. 1019-1022). IEEE. https://ieeexplore.ieee.org/abstract/document/9421225
4. Misra, Rishabh and Jigyasa Grover. "Sculpting Data for ML: The first act of Machine Learning." ISBN 9798585463570 (2021).
5. Word embedding : https://serokell.io/blog/word2vec
6. BertTokenizer : https://www.analyticsvidhya.com/blog/2021/09/an-explanatory-guide-to-bert-tokenizer/

Link Video Presentasi: Text Mining_AoL_Presentasi