BAB 2

LANDASAN TEORI

Dalam menyusun dan melaksanakan penelitian ini, dilakukan studi literatur yang akan mendasari teori dari penelitian yang sedang dilakukan. Teori-teori yang mendasari penelitian ini dapat dijabarkan sebagai berikut:

2.1 Natural Language Processing

Natural Language Processing (NLP) adalah rangkaian dari teknik komputasi untuk menganalisa dan merepresentasikan teks yang terjadi secara alami pada satu atau lebih tingkat analisis linguistik dengan tujuan mencapai pemrosesan bahasa yang menyerupai manusia untuk berbagai tugas atau aplikasi (Liddy & D., 2001). NLP merupakan pengembangan dari Artificial Intelligence atau kecerdasan buatan yang diharapkan dapat mempelajari bahasa yang digunakan oleh manusia. Saat ini, NLP sudah banyak digunakan manusia seperti Google Translate, Google Assistant, Siri, Alexa, dan sebagainya. NLP memiliki banyak potensi untuk dikembangkan lebih lanjut, salah satunya adalah *multilingual text classification* yang akan dibahas lebih dalam pada penelitian ini.

2.2 Multilingual Text Classification

Text classification atau klasifikasi teks merupakan proses yang dilakukan oleh mesin atau komputer untuk mengkategorikan suatu teks ke dalam kelompok yang terorganisir. *Text classification* merupakan bagian dari NLP yang dapat menganalisa teks secara otomatis dan memberikan label atau kategori yang telah ditentukan sebelumnya berdasarkan konten dari teks tersebut. *Text classification* memungkinkan *user* untuk menjelajah dengan lebih mudah pada teks yang mereka

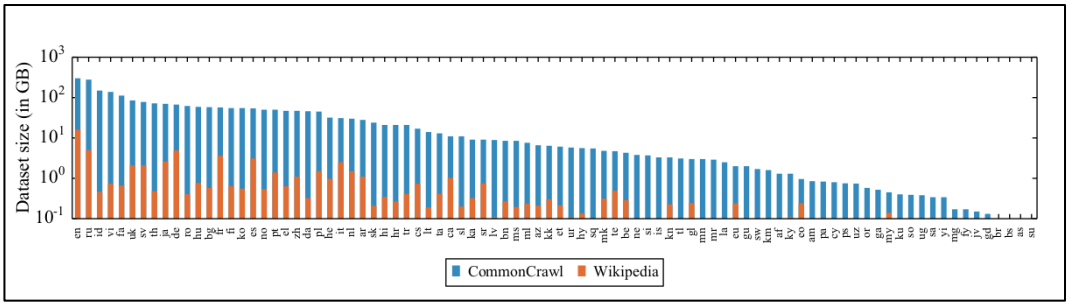
minati dan paradigma ini sangat efektif dalam penyaringan informasi seperti dalam pengembangan layanan secara *online* (Gonalves & Quaresma, 2010).

Walaupun kebanyakan dari penelitian mengenai *text classification* mengarah pada *monolingual text classification*, saat ini mulai banyak penelitian yang mengarah pada *multilingual text classification*. *Multilingual text classification* sendiri merupakan penelitian baru yang memungkinkan untuk melakukan *text classification* pada dokumen teks dengan berbagai bahasa. Dari kebanyakan penelitian mengenai *multilingual text classification*, model yang digunakan untuk melakukan klasifikasi mengandalkan *training* pada *monolingual documents* atau dokumen dengan satu bahasa, kemudian menggunakan *translation mechanism* atau mekanisme penerjemahan untuk melakukan klasifikasi dokumen yang tertulis dengan bahasa lain (Bel, 2003; Rigutini, 2005; Lee, 2009).

2.3 XLM-R

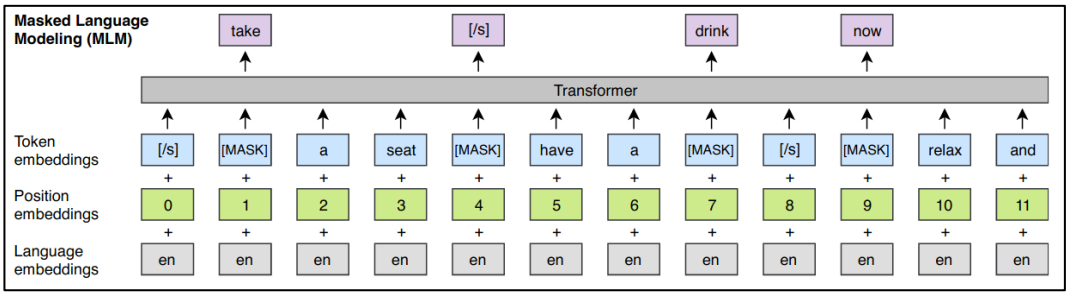
XLM-R atau *Cross Lingual Model – RoBERTa* adalah pengembangan dari XLM dan mBERT yang merupakan penelitian *multilingual Natural Language Processing* (NLP) model sebelumnya. XLM-R menggunakan *transformer-based multilingual mask language model* (MLM) yang sudah dilakukan *pre-trained* pada teks dengan 100 bahasa dan mampu menunjukkan performa yang sangat baik pada *cross-lingual classification, sequence labeling,* dan *question answering* (Conneau & Khandelwal, 2020). XLM-R merupakan salah satu algoritma *machine learning* di dalam *library* Transformer, sebuah *library* yang menyediakan ribuan *pretained-model* untuk melakukan pekerjaan seperti klasifikasi teks, *information extraction*, *question answering*, dan lain-lain. Pengembangan XLM-R memiliki tujuan untuk

meningkatkan kemampuan mesin atau komputer dalam melakukan *multilingual Natural Language Processing* (NLP) terutama pada *low-resource languages*.



Gambar 2.1 Jumlah data dalam GiB untuk 88 bahasa yang digunakan dalam penelitian XLM-R

Tidak seperti XLM, pengembangan XLM-R justru menghindari metode yang digunakan oleh XLM yaitu Translation Language Modeling (TLM). XLM-R menggunakan metode Mask Language Modeling (MLM) karena pengembangan XLM-R lebih fokus pada *unsupervised learning*. MLM merupakan bagian dari Transformer model yang memiliki tujuan untuk memprediksi token yang hilang dari sebuah input dan merekonstruksi ulang urutan input yang sesungguhnya (dapat dilihat pada Gambar 2.2). Namun, MLM tidak memiliki akses terhadap keseluruhan input tersebut, melainkan hanya memiliki akses terhadap *masked token*. Konsep MLM ini juga digunakan pada BERT, mBERT, RoBERTa, dan XLM (MLM pada XLM dikombinasikan dengan *language modeling* yang lain).



Gambar 2.2 Konsep Masked Language Modeling

Secara sederhana, XLM-R hanya melatih RoBERTa pada *multilingual dataset* dengan ukuran yang sangat besar. *Multilingual dataset* dengan teks yang belum dilabeli dalam 100 bahasa tersebut diekstrak dari *CommonCrawl datasets*, dengan ukuran sebesar 2,5 TB. Pada penelitian XLM-R, dipetimbangan untuk membuang 12 bahasa untuk bahasa lain, yang secara signifikan meningkatkan ukuran dataset, terutama untuk *low-resource languages* seperti Burma dan Swahili, sehingga hanya tersisa 88 bahasa (Conneau & Khandelwal, 2020). Gambar 2.1 merupakan jumlah data dalam GiB untuk masing-masing bahasa yang diambil baik dari CommonCrawl maupun Wikipedia. Berdasarkan perbandingan data tersebut, terlihat jelas bahwa penelitian XLM-R menggunakan ukuran data yang jauh lebih besar dibandingkan penelitian sebelumnya.

Walaupun menggunakan metode yang sama dengan RoBERTa, keduanya memiliki perbedaan yang sangat mendasar pada ukuran kosakata: di mana RoBERTa menggunakan 50.000 token sedangkan XLM-R menggunakan 250.002 token. Jumlah kosakata atau *vocabulary size* yang digunakan oleh XLM-R merupakan gabungan kosakata dari beberapa bahasa, sehingga *tokenizer* milik XLM-R memungkinkan untuk melakukan *tokenization* pada beberapa bahasa

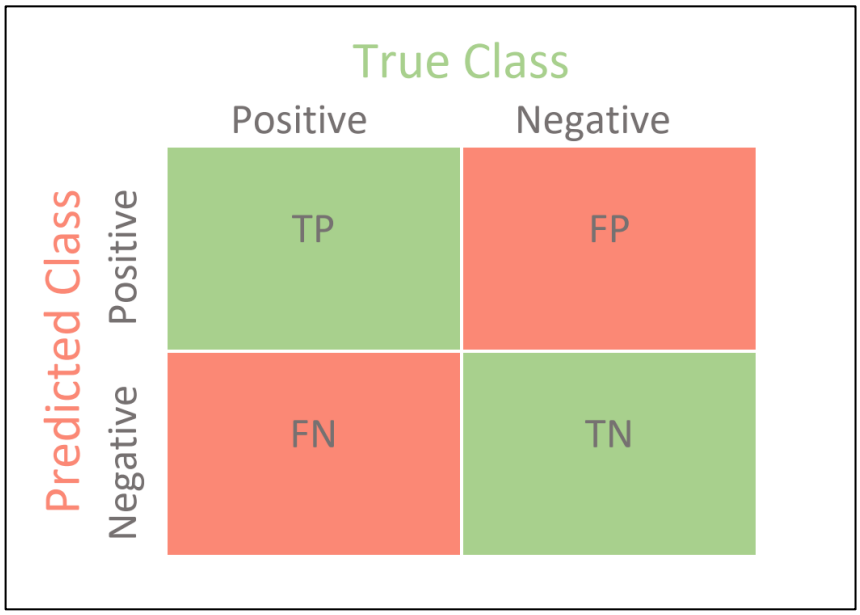
secara langsung. Berbeda dengan *tokenization tools* yang digunakan oleh BERT dan XLM, XLM-R melatih sebuah Sentence Piece Model (SPM) dan diaplikasikan secara langsung pada *raw text data* untuk semua bahasa. Sence Piece Model menggunakan *library* SentencePiece dalam proses pembuatan modelnya, di mana *libray* tersebut merupakan *subword tokenizer and detokenizer* independen yang dirancang untuk *neural-based text processing*, termasuk Neural Machine Translation. SentencePiece mampu melakukan *subword tokenization* dan langsung mengkonversi teks menjadi *id sequence* (Kudo & Richardson, 2018).

XLM-R mampu melakukan *zero-shot transfer*, yaitu melakukan *fine-tune* pada *multilingual model* menggunakan *training dataset* satu bahasa saja, kemudian melakukan *cross-lingual transfer* pada bahasa lain. Artinya, XLM-R hanya perlu memberikan label dan mempelajari pada satu bahasa saja (contoh bahasa Inggris) yang kemudian dapat melakukan *self-supervised learning* atau memberikan label secara otomatis pada *datasets* dengan bahasa yang berbeda (contoh bahasa Indonesia). Dengan melakukan *cross-lingual transfer* (pada *training set* berbahasa Inggris), XLM-R mendapatkan rata-rata akurasi sebesar 80,9% (Conneau & Khandelwal, 2020).

2.4 Confusion Matrix

Confusion Matrix merupakan salah satu metrik pengukuran performa untuk klasifikasi *machine learning* yang memiliki hasil keluaran dua kelas atau lebih. Penghitungan pada Confusion Matrix mempertimbangkan nilai antara hasil prediksi dengan nilai sesungguhnya. Terdapat 4 variabel yang digunakan Confusion

Matrix dalam melakukan perhitungan, yaitu True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN).



Gambar 2.3 Confusion Matrix

Gambar 2.4 merupakan matrix merupakan gambaran lebih jelas dari variabel yang telah disebutkan sebelumnya. Notasi pada Confusion Matrix dapat dijelaskan sebagai berikut:

- True Positive (TP): Ini mengacu pada jumlah prediksi di mana mesin pengklasifikasi benar memprediksi kelas positif sebagai negatif.
- True Negative (NP): mengacu pada jumlah prediksi di mana mesin pengklasifikasi benar memprediksi kelas negatif sebagai negatif
- False Positive (FP): mengacu pada jumlah prediksi di mana mesin pengklasifikasi salah memprediksi kelas negatif sebagai positif
- False Negative (NP): mengacu pada jumlah prediksi di mana mesin pengklasifikasi salah memprediksi kelas positif sebagai negatif

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} * 100\%$$

$$Presisi = \frac{TP}{FP+TP} * 100\%$$

$$Recall = \frac{TP}{FN+TP} * 100\%$$

Gambar 2.4 Rumus Accuracy, Precision, dan Recall pada Confusion Matrix

Penghitungan Confusion Matrix akan menghasilkan 4 nilai akhir, yaitu *accuracy, precision, recall,* dan *F1-score*. Gambar 2.5 merupakan rumus penghitungan untuk *accuracy, precision,* dan *recall*. Di antara seluruh kelas (positif dan negatif), nilai akurasi mengukur seberapa benar mesin pengklasifikasi berhasil melakukan prediksi. Nilai *precision* mengukur dari seluruh hasil prediksi positif, seberapa banyak data yang memiliki nilai aktual positif. Sedangkan untuk nilai *recall* mengukur dari seluruh hasil prediksi positif, seberapa banyak data yang berhasil diprediksi dengan benar.

$$F1 = \frac{2\ x\ Recall\ x\ Precision}{Recall + Precision}$$

Gambar 2.5 Rumus F1-score pada Confusion Matrix

Nilai *F1-score* digunakan untuk mempertimbangkan nilai antara *precision* dan *recall*. Gambar 2.6 merupakan rumus yang digunakan untuk melakukan penghitungan *F1-score*. Berdasarkan rumus tersebut, variabel yang digunakan untuk perhitungan adalah nilai dari *precision* dan *recall* sebelumnya

2.5 Matthew Correlation Coefficient

Ketidakseimbangan data merupakan hal yang sering terjadi dalam sebuah penelitian ketika ukuran sampel di kelas atau kategori yang digunakan tidak merata (Daskalaki, et al., 2006). Salah satu cara untuk mengatasi ketidakseimbangan data adalah dengan menggunakan algoritma perhitungan yang mempertimbangkan ukuran sampel yang digunakan. Matthew Correlation Coefficient atau MCC merupakan salah satu algoritma yang dapat digunakan untuk menghitung nilai dari data yang memiliki ukuran sampel yang tidak merata (Boughorbel, et al., 2017).

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

Gambar 2.6 Rumus Matthew Correlation Coefficient

Notasi pada gambar 2.3 terdiri dari True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN). Hasil dari penghitungan menggunakan algoritma MCC akan mengembalikan nilai antara -1 hingga +1. Hasil nilai mendekati atau sama dengan -1 memiliki arti ketidaksesuaian prediksi dengan data seharusnya. Sebaliknya, nilai MCC yang mendekati atau sama dengan +1 memiliki arti kesesuaian prediksi dengan data seharusnya. Penghitungan Matthew Correlation Coefficient mempertimbangkan korelasi antara nilai True Positive dengan nilai True Negative, sehingga nilai MCC akan memiliki hasil yang lebih baik pada data dengan ukuran sampel yang tidak merata untuk setiap kelasnya. Pada penelitian ini, digunakan nilai MCC sebagai salah satu nilai pengukuran yang digunakan untuk mengevaluasi *model* yang telah dibuat.