**OVERVIEW**

Teks diberikan kepada model, dan model mengklasifikasikan teks tersebut ke dalam salah satu dari beberapa kelas yang mungkin. Misalnya, mengklasifikasikan artikel berita ke dalam kategori seperti "olahraga", "politik", "teknologi", dll

**TAHAPAN**

1. Pengumpulan data
2. Prapemrosesan data

* Pembersihan teks
* Normalisasi
* Tokenisasi → memecah teks menjadi kata-kata/token
* Stop word removal → Menghapus kata-kata umum yang tidak berkontribusi banyak pada makna (misalnya, "dan", "yang")
* stemming/lemmatization → Mengurangi kata-kata ke bentuk dasarnya. Stemming adalah Memotong akhiran kata untuk mendapatkan akar kata (lebih cepat, tetapi kurang akurat) sedangkan Lemmatization adalah Mengurangi kata ke bentuk leksikalnya yang benar dengan mempertimbangkan konteks (lebih lambat, tetapi lebih akurat)

Beberapa algoritma untuk melakukan stemming/lemmatization

* + Porter stemmer
  + Snowball stemmer
  + Lancaster stemmer
  + WordNet lemmatizer
  + SpaCy lemmatizer

1. Representasi teks

* Bag of words (BoW): Mewakili teks sebagai vektor frekuensi kata.
* TF-IDF (Term Frequency-Inverse Document Frequency): Menghitung pentingnya kata dalam dokumen relatif terhadap seluruh korpus
* Word Embeddings: Menggunakan representasi kata berdimensi rendah seperti Word2Vec, GloVe, atau representasi berbasis transformer seperti BERT.

1. Pembagian data

* Pisahkan data menjadi data pelatihan, validasi, dan pengujian.

1. Pemilihan model

Beberapa model yang umum digunakan untuk text classification:

* Naive Bayes
* Logistic Regression
* Support Vector Machine (SVM)
* Decision trees/random forest
* Neural networks
* Transformer-based models (BERT, GPT)

1. Pelatihan model

* Gunakan cross-validation untuk menghindari overfitting.

1. Evaluasi model

* Gunakan metrik seperti akurasi, presisi, recall, F1-score, dan confusion matrix untuk mengevaluasi performa model pada data validasi dan pengujian.

1. Hyperparameter tuning

* Sesuaikan hyperparameter model untuk mendapatkan kinerja terbaik menggunakan teknik seperti Grid Search, Random Search, Bayesian Optimization, Automated Machine Learning, dll
* Tips melakukan hyperparameter tuning:

1. Mulai dengan default: Banyak model memiliki hyperparameter default yang cukup baik. Mulailah dengan default sebelum mencoba penyesuaian yang lebih kompleks.
2. Pahami model anda: Setiap model memiliki hyperparameter yang berbeda dan penting untuk memahami apa yang dilakukan setiap hyperparameter. beberapa contoh model pembelajaran mesin yang populer beserta hyperparameter mereka:
3. Decision tree

| from sklearn.tree import DecisionTreeClassifier  model = DecisionTreeClassifier(criterion='gini', max\_depth=10, min\_samples\_split=2, min\_samples\_leaf=1) |
| --- |

* criterion: Fungsi yang mengukur kualitas split
* max\_depth: Kedalaman maksimum dari tree
* min\_samples\_split: Jumlah minimum sampel yang diperlukan untuk membagi node internal
* min\_samples\_leaf: Jumlah minimum sampel yang diperlukan untuk berada di leaf node

1. Random forest

| from sklearn.ensemble import RandomForestClassifier  model = RandomForestClassifier(n\_estimators=100, max\_depth=10, min\_samples\_split=2, min\_samples\_leaf=1) |
| --- |

* n\_estimators: Jumlah pohon dalam forest
* criterion: Fungsi yang mengukur kualitas split
* max\_depth: Kedalaman maksimum dari setiap pohon
* min\_samples\_split: Jumlah minimum sampel yang diperlukan untuk membagi node internal
* min\_samples\_leaf: Jumlah minimum sampel yang diperlukan untuk berada di leaf node

1. Gunakan cross-validation: Selalu gunakan cross-validation untuk mengevaluasi kinerja model selama penyesuaian hyperparameter
2. Iteratif: Mulai dengan pencarian yang lebih kasar dan secara bertahap mempersempit rentang hyperparameter
3. Sumber daya komputasi: Pertimbangkan sumber daya komputasi yang tersedia karena beberapa metode penyesuaian hyperparameter bisa sangat mahal secara komputasi
4. Deployment

* Setelah model dilatih dan dievaluasi, deploy model untuk digunakan pada data nyata.

**GLOSARIUM**

**Korpus**

Teks-teks yang dikumpulkan dan disusun secara sistematis untuk berbagai tujuan penelitian dan aplikasi, seperti pelatihan model pembelajaran mesin, analisis linguistik, dan pengembangan teknologi bahasa.

Jenis-jenis korpus:

1. Korpus monolingual → Kumpulan teks dalam satu bahasa, cth: Korpus berita berbahasa Indonesia, korpus artikel ilmiah berbahasa Inggris
2. Korpus bilingual → Kumpulan teks yang terdiri dari pasangan teks dalam dua bahasa yang berbeda, cth: Teks artikel dalam bahasa Inggris dan terjemahannya dalam bahasa Prancis
3. Korpus multilingual → Kumpulan teks dalam beberapa bahasa
4. Korpus terspesialisasi → Kumpulan teks yang terfokus pada bidang atau topik tertentu
5. Korpus seimbang → Korpus yang dirancang untuk mencerminkan variasi bahasa dalam konteks yang berbeda secara proporsional, cth: Korpus British National Corpus (BNC) yang mencakup berbagai genre seperti fiksi, non-fiksi, ilmiah, dan percakapan sehari-hari

Contoh korpus yang terkenal

1. British National Corpus (BNC)
2. Corpus of Contemporary American English (COCA)
3. Wikipedia Corpus
4. Brown Corpus