Kaggle Viva – Make Data Count (සිංහල සාරාංශ ගෛඩ්)

මෙය ඔබට විවාදීකරණය (Viva) හි ඉතා පැහැදිලිව පැහැදිලි කිරීමට උදවු කරන සරල, සම්පූර්ණ සාරාංශයක්. කෙටි වචන, ඉතා පැහැදිලි උදාහරණ, සහ ඉක්මන් පිළිතුරු ඇතුළත් වේ.

---

1) තරඟය සහ ඉලක්කය

- **තරඟය:** Kaggle – *Make Data Count: Finding Data References*
- **ඉලක්කය:** විද්‍යා ලේඛන (XML) තුළ ඇති **dataset හැඳින්වීම්** Primary / Secondary / Missing ලෙස **වර්ගීකරණය** කිරීම.
- **ප්‍රයෝජනය:** දත්ත සමුදා **නිර්මාණය/නිකුත් කිරීම** (Primary) සහ **නැවත භාවිතය/උපුටා දක්වීම** (Secondary) ස්වයංක්‍රීයව හඳුනාගෙන **ප්‍රභාවය** මැනීම.

---

2) Primary / Secondary / Missing කියන්නේ මොනවද?

- **Primary (ප්‍රාථමික):** ලේඛනය **දත්ත එකතු/නිර්මාණය/නිකුත්** කරයි.
  උදා: "අපි දත්ත එකතු කළා", "මෙම දත්ත සමුදාය නිකුත් කරමු".
- **Secondary (ද්විතීය):** වෙනත් අය සෑදූ දත්ත **භාවිතා/උපුටා** කරයි.
  උදා: "as reported by", "based on".
- **Missing:** හඳුනාගත් candidate එකක් නමුත් **අදාළ නොවූ/අස්පෂ්ට** හෝ **භාවිතයට නොගත හැකි**.

---

3) දත්ත කට්ටලය (Dataset)

- **train_labels.csv:** `article_id`, `dataset_id`, `type` (Primary/Secondary/Missing).
- **XML ගොනු:** train/test සඳහා සූර්ණ ලිපි පාඨ.
- **Split:** 80/20 (train/validation), `random_state=42` – නැවත නිපදවිය හැකි ප්‍රතිඵල.

---

4) Preprocessing (XML → Context)

1. **XML කියවීම:** `ElementTree` මඟින් ලිපියේ පාඨය ලබාගන්න.
2. **Regex mentions:** DOI/Accession හැඳින්වීම් (`10.xxxx/...`, `GSE12345`, `PRJ...`, `CHEMBL...`) සොයන්න.
3. **Context**: Label එකට ගැළපෙන mention එකට ආසන්න **වාක්‍යය** context එක ලෙස ගන්න.
4. **Filter**: වචන **5ට අඩු** contexts ඉවත් කර ශබ්ද දූෂණය අඩු කරන්න.

---

5) Features (TF-IDF)

- **Unigram + Bigram**, **English stop-words** ඉවත් කිරීම, **~3000 features**.
- **අර්ථය:** ලේඛනයේ සාමාන්‍ය වචනවල බර අඩුවෙයි; විශේෂ වචන/ප්‍රකාශන **වැදගත්කම වැඩිවෙයි**.

---

6) Model (Multiclass Logistic Regression)

- **ඉක්මන් + පැහැදිලි + අර්ථදායී** baseline.
- `class_weight='balanced'` – පංති අසමසමතාවය (Primary කලාපය කුඩා) හසුරවයි.

---

7) Training & Evaluation

- **Split:** 80/20.
- **Metrics: Macro Precision / Recall / F1** (සියලු පංති සමාන වටිනාකමකින්).
- **පෙරදේසි දෝෂ:** Short/අස්පෂ්ට context → Secondary → Missing ලෙස වැරදි; Primary අල්ලාගැනීම අභියෝගාත්මක.

---

8) Inference & Submission

- **Test XML** → Regex mentions → **TF-IDF** → **Predict**.
- **submission.csv**: `row_id`, `article_id`, `dataset_id`, `type`.
- Duplicate පේළි **ඉවත්** කර උපරිම නිරවද්‍යතාවය රැකගන්න.

---

9) Reproducibility

- Kaggle මත **එකම inputs** සමඟ නැවත ධාවනය කළ හැක.
- **Fixed seed**, සපිළිඹඳ පයිප්ලයින් පියවර.

---

10) සීමා & වර්ධක අදහස්

- **Context කෙටි** වීම; **Regex ආවරණය** සීමිත.
- **වර්ධනය:** විශාල context window (කොටස් අනුව), **Dataset-mention NER**, **SciBERT/BERT** fine-tune, **Ensembles**.

---

11) 60-Second Pitch (සිංහලෙන්)

"අපි Kaggle හි Make Data Count කටයුතු සඳහා සරල නමුත් ශක්තිමත් NLP පයිප්ලයිනක් සකස් කළා. XML ලිපිවලින් Regex මඟින් දත්ත හැඳින්වීම් සොයා ඒවාට ආසන්න වාක්‍ය Context එක ගත්තා. ඒ Context වලින් TF-IDF features සකස් කර class_weight='balanced' යොදා Multiclass Logistic Regression මූලාකෘතිය පුහුණු කළා. ඇගයීමට Macro Precision/Recall/F1 භාවිතා කළා. පසුකාලීනව SciBERT/BERT fine-tune සහ විශාල Context windows යොදා Primary vs Secondary වඩා පැහැදිලිව වෙන්කළ හැකි. Kaggle මත එය නැවත ධාවනය කර submission.csv ජනනය කළ හැක."

---

12) Viva ප්‍රශ්න — ඉක්මන් පිළිතුරු

- **තරඟය/ඉලක්කය?** Dataset mentions Primary/Secondary/Missing ලෙස වර්ගීකරණය.
- **දත්ත මූලාශ්‍ර?** train_labels.csv + article XML.
- **Preprocessing?** XML → Regex mentions → Context → Short filter.
- **Features + Model?** TF-IDF (1–2 grams) + Logistic Regression (balanced).
- **Metrics?** Macro Precision/Recall/F1.
- **Submission?** row_id, article_id, dataset_id, type.
- **සීමා/භාවිතා කළ විසඳුම්?** Class imbalance → balanced weights; Context කෙටි → window වැඩිකිරීම.