

✓ Experiments with models trained on subword TF-IDF features

Start coding or [generate](#) with AI.

```
!pip install -qq datasets wandb scikit-learn xgboost scipy
```



Show hidden output

```
import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import pickle
import xgboost as xgb
from google.colab import drive
from sklearn.dummy import DummyClassifier, DummyRegressor
from sklearn.linear_model import LogisticRegression, Ridge
from sklearn.metrics import accuracy_score, f1_score, mean_squared_error, mean_
from scipy.sparse import vstack
from datasets import load_dataset
import wandb
import logging
import time
import gc
import json
import yaml
from typing import List, Dict, Tuple, Optional, Any

pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', 20)
pd.set_option('display.width', 1000)

SEED = 42
np.random.seed(SEED)
```


Start coding or [generate](#) with AI.

```
drive.mount('/content/drive')

DATA_PATH = "/content/drive/MyDrive/ColabNotebooks/qtype-eval/data/"
CACHE_DIR = os.path.join(DATA_PATH, "cache")
VECTORS_DIR = os.path.join(DATA_PATH, "tfidf_vectors")
OUTPUT_DIR = os.path.join(DATA_PATH, "results")
```

```
os.makedirs(CACHE_DIR, exist_ok=True)
os.makedirs(VECTORS_DIR, exist_ok=True)
os.makedirs(OUTPUT_DIR, exist_ok=True)

LANGUAGES = ["ar", "en", "fi", "id", "ja", "ko", "ru"]
TASKS = ["question_type", "complexity"]
SUBMETRICS = ["avg_links_len", "avg_max_depth", "avg_subordinate_chain_len", "avg_
```

 Mounted at /content/drive

```
class ExperimentConfig:
    def __init__(
        self,
        task: str,
        model_type: str,
        languages: List[str] = None,
        submetric: Optional[str] = None,
        control_index: Optional[int] = None,
        model_params: Dict = None,
        experiment_name: Optional[str] = None):

        self.task = task
        self.model_type = model_type
        self.languages = languages or ["all"]
        self.submetric = submetric
        self.control_index = control_index
        self.model_params = model_params or {}
        self.task_type = "classification" if task == "question_type" else "regr

        if not experiment_name:
            control_str = f"_control{control_index}" if control_index else ""
            submetric_str = f"_{submetric}" if submetric else ""
            self.experiment_name = f"{model_type}_{task}{submetric_str}{control
        else:
            self.experiment_name = experiment_name

        self.output_dir = os.path.join(OUTPUT_DIR, self.experiment_name)
        os.makedirs(self.output_dir, exist_ok=True)

    def to_dict(self):
        return {
            "task": self.task,
            "model_type": self.model_type,
            "languages": self.languages,
            "submetric": self.submetric,
            "control_index": self.control_index,
            "task_type": self.task_type,
            "experiment_name": self.experiment_name,
            "model_params": self.model_params}

    def get_dataset_name(task, control_index=None, submetric=None):
        if control_index is None:
            return "base"
```

```

        return base

    if task == "question_type":
        return f"control_question_type_seed{control_index}"
    elif task == "complexity":
        if submetric:
            return f"control_{submetric}_seed{control_index}"
        else:
            return f"control_complexity_seed{control_index}"

    return "base"

def load_dataset_from_huggingface(config_name, split="train"):
    repo_id = "rokokot/question-type-and-complexity"

    if split == "train":
        file_path = f"tydi_train_{config_name}.csv"
    elif split == "validation":
        file_path = "dev_base.csv"
    elif split == "test":
        file_path = "ud_test_base.csv"
    else:
        raise ValueError(f"Unknown split: {split}")

    try:
        dataset = load_dataset(repo_id, data_files={split: file_path}, split=split)
        return dataset
    except Exception as e:
        print(f"Error loading {split} dataset with config {config_name}: {e}")
        raise

def load_tfidf_features(split: str, config_name=None, vectors_dir: str = VECTORS_DIR):
    file_path = os.path.join(vectors_dir, f"X_{split}.npy")

    print(f"Loading features from {file_path}")
    if not os.path.exists(file_path):
        raise FileNotFoundError(f"TF-IDF features not found at {file_path}")

    try:
        vectors = np.load(file_path, allow_pickle=True)
        sparse_matrices = [vectors[i, 0] for i in range(vectors.shape[0])]
        stacked = vstack(sparse_matrices)

        print(f"Loaded vectors shape: {vectors.shape}")
        return stacked
    except Exception as e:
        print(f"Error loading TF-IDF features: {e}")
        raise

```

Start coding or [generate](#) with AI.

Start coding or [generate](#) with AI.

```

def prepare_datasets(config: ExperimentConfig):
    config_name = get_dataset_name(config.task, config.control_index, config.submetric)
    print(f"Using dataset config: {config_name}")

    try:
        train_dataset = load_dataset_from_huggingface(config_name, "train")
        val_dataset = load_dataset_from_huggingface("base", "validation")
        test_dataset = load_dataset_from_huggingface("base", "test")

        train_df = train_dataset.to_pandas()
        val_df = val_dataset.to_pandas()
        test_df = test_dataset.to_pandas()

        print(f"Loaded dataset splits - Train: {len(train_df)}, Validation: {len(val_df)}, Test: {len(test_df)}")
    except Exception as e:
        print(f"Error loading datasets: {e}")
        raise

    if config.task == "question_type":
        label_col = "question_type"
    elif config.task == "complexity":
        if config.submetric:
            label_col = config.submetric
        else:
            label_col = "lang_norm_complexity_score"
    else:
        raise ValueError(f"Unknown task: {config.task}")

    y_train = train_df[label_col].values
    y_val = val_df[label_col].values
    y_test = test_df[label_col].values

    X_train = load_tfidf_features("train")
    X_val = load_tfidf_features("dev")
    X_test = load_tfidf_features("test")

    print(f"Train features: {X_train.shape}, Train samples: {len(y_train)}")
    print(f"Dev features: {X_val.shape}, Dev samples: {len(y_val)}")
    print(f"Test features: {X_test.shape}, Test samples: {len(y_test)}")

    if config.task_type == "classification":
        y_train = y_train.astype(np.int64)
        y_val = y_val.astype(np.int64)
        y_test = y_test.astype(np.int64)
    else:
        y_train = y_train.astype(np.float32)
        y_val = y_val.astype(np.float32)
        y_test = y_test.astype(np.float32)

    return (X_train, y_train), (X_val, y_val), (X_test, y_test)

def print_dataset_info(dataset_dict):
    """Print dataset information"""
    for dataset, (X, y) in dataset_dict.items():
        print(f"{dataset} features: {X.shape}, samples: {len(y)}")

```

```

for name, df in dataset_dict.items():
    print(f"\n{'-' * 40}\n{name} Dataset Information:")
    print(f"Shape: {df.shape}")
    print(f"Columns: {' , '.join(df.columns)}")

    print("\nFirst 5 rows:")
    print(df.head(5))

    if "question_type" in df.columns:
        print("\nQuestion Type Distribution:")
        print(df["question_type"].value_counts())

    if "lang_norm_complexity_score" in df.columns:
        print("\nComplexity Score Statistics:")
        print(df["lang_norm_complexity_score"].describe())

    for submetric in SUBMETRICS:
        if submetric in df.columns:
            print(f"\n{submetric} Statistics:")
            print(df[submetric].describe())

print("Dataset Loading Verification")
print("=" * 50)

configs_to_check = [
    {"task": "question_type", "control_index": None, "submetric": None},
    {"task": "question_type", "control_index": 1, "submetric": None},
    {"task": "complexity", "control_index": 1, "submetric": None},
    {"task": "complexity", "control_index": 1, "submetric": SUBMETRICS[0]}
]

for config in configs_to_check:
    task = config["task"]
    control_index = config["control_index"]
    submetric = config["submetric"]

    print("\n\n" + "=" * 50)
    print(f"Verifying configuration: task={task}, control={control_index}, subn

    config_name = get_dataset_name(task, control_index, submetric)
    print(f"Dataset name: {config_name}")
    datasets = {}
    try:
        train_dataset = load_dataset_from_huggingface(config_name, "train")
        val_dataset = load_dataset_from_huggingface("base", "validation")
        test_dataset = load_dataset_from_huggingface("base", "test")

        datasets["Train"] = train_dataset.to_pandas()
        datasets["Validation"] = val_dataset.to_pandas()
        datasets["Test"] = test_dataset.to_pandas()

        print_dataset_info(datasets)

        print("\nChecking TF-IDF feature paths:")

```

```

load_train_features("train")
load_tfidf_features("dev")
load_tfidf_features("test")

except Exception as e:
    print(f"Error verifying configuration: {e}")

print("=" * 50)

print("\nVerification complete.")

```

Show hidden output

```

def create_sklearn_model(model_type: str, task_type: str, **kwargs):
    print(f"Creating {model_type} model for {task_type} task")

    if model_type == "dummy":
        if task_type == "classification":
            return DummyClassifier(
                strategy=kwargs.get("strategy", "most_frequent"),
                random_state=kwargs.get("random_state", SEED)
            )
        else:
            return DummyRegressor(strategy=kwargs.get("strategy", "mean"))

    elif model_type == "logistic":
        if task_type != "classification":
            print("Logistic regression is for classification only; continuing a

        return LogisticRegression(
            C=kwargs.get("C", 1.0),
            max_iter=kwargs.get("max_iter", 1000),
            random_state=kwargs.get("random_state", SEED),
            n_jobs=kwargs.get("n_jobs", -1),
            solver=kwargs.get("solver", "liblinear"),
            penalty=kwargs.get("penalty", "l2")
        )

    elif model_type == "ridge":
        if task_type != "regression":
            print("Ridge regression is for regression only; continuing anyway")

        return Ridge(
            alpha=kwargs.get("alpha", 1.0),
            random_state=kwargs.get("random_state", SEED),
            solver=kwargs.get("solver", "auto")
        )

    elif model_type == "xgboost":
        if task_type == "classification":
            return xgb.XGBClassifier(
                n_estimators=kwargs.get("n_estimators", 100),
                max_depth=kwargs.get("max_depth", 6),
                learning_rate=kwargs.get("learning_rate", 0.1).

```

```

        reg_alpha=kwards.get("reg_alpha", 0),
        reg_lambda=kwards.get("reg_lambda", 1),
        subsample=kwards.get("subsample", 0.8),
        colsample_bytree=kwards.get("colsample_bytree", 0.8),
        random_state=kwards.get("random_state", SEED),
        use_label_encoder=False,
        eval_metric=kwards.get("eval_metric", "logloss"),
    )
else:
    return xgb.XGBRegressor(
        n_estimators=kwards.get("n_estimators", 100),
        max_depth=kwards.get("max_depth", 6),
        learning_rate=kwards.get("learning_rate", 0.1),
        reg_alpha=kwards.get("reg_alpha", 0),
        reg_lambda=kwards.get("reg_lambda", 1),
        subsample=kwards.get("subsample", 0.8),
        colsample_bytree=kwards.get("colsample_bytree", 0.8),
        random_state=kwards.get("random_state", SEED),
        eval_metric=kwards.get("eval_metric", ["rmse", "mae"]),
    )

else:
    raise ValueError(f"Unknown model type: {model_type}")

def calculate_metrics(y_true, y_pred, task_type):
    if task_type == "classification":
        return {
            "accuracy": float(accuracy_score(y_true, y_pred)),
            "f1": float(f1_score(y_true, y_pred, average="binary")),
        }
    else:
        return {
            "mse": float(mean_squared_error(y_true, y_pred)),
            "rmse": float(np.sqrt(mean_squared_error(y_true, y_pred))),
            "mae": float(mean_absolute_error(y_true, y_pred)),
            "r2": float(r2_score(y_true, y_pred)),
        }

def train_and_evaluate(
    model,
    X_train, y_train,
    X_val, y_val,
    X_test, y_test,
    task_type,
    output_dir=None,
    wandb_run=None
):
    print(f"Training {model.__class__.__name__} on {X_train.shape[0]} examples,

    start_time = time.time()

    if hasattr(model, 'eval_metric') and X_val is not None and y_val is not None:
        model.fit(

```

```
        X_train, y_train,
        eval_set=[(X_train, y_train), (X_val, y_val)],
        verbose=100
    )
else:
    model.fit(X_train, y_train)

training_time = time.time() - start_time
print(f"Training completed in {training_time:.2f} seconds")

y_pred_val = model.predict(X_val) if X_val is not None else None
y_pred_test = model.predict(X_test) if X_test is not None else None

val_metrics = calculate_metrics(y_val, y_pred_val, task_type) if y_pred_val
test_metrics = calculate_metrics(y_test, y_pred_test, task_type) if y_pred_

if val_metrics:
    print(f"Validation metrics: {val_metrics}")
if test_metrics:
    print(f"Test metrics: {test_metrics}")

results = {
    "model_type": model.__class__.__name__,
    "training_time": training_time,

    "val_metrics": val_metrics,
    "test_metrics": test_metrics,
}

if wandb_run:
    wandb_metrics = {
        "training_time": training_time,
    }

    if val_metrics:
        wandb_metrics.update({f"val_{k}": v for k, v in val_metrics.items()})

    if test_metrics:
        wandb_metrics.update({f"test_{k}": v for k, v in test_metrics.items()})

    wandb_run.log(wandb_metrics)

if hasattr(model, 'feature_importances_'):
    try:
        importance = model.feature_importances_
        if len(importance) > 50:
            top_indices = np.argsort(importance)[-50:]
            top_importances = importance[top_indices]
            feature_data = [[int(idx), float(imp)] for idx, imp in zip(
        else:
            feature_data = [[int(i), float(imp)] for i, imp in enumerat
```



```
wandb_run.log({
    "feature_importance_table": wandb.Table(
        data=feature_data,
        columns=["feature_index", "importance"]
    )
})

if len(feature_data) > 0:
    plt.figure(figsize=(10, 6))
    indices = [row[0] for row in feature_data[:20]]
    importances = [row[1] for row in feature_data[:20]]
    plt.bar(range(min(20, len(indices))), importances)
    plt.title('Top Feature Importances')
    plt.xlabel('Feature Index')
    plt.ylabel('Importance')
    plt.tight_layout()
    wandb_run.log({"feature_importance_plot": wandb.Image(plt)})
    plt.close()
except Exception as e:
    print(f"Could not log feature importances: {e}")

if task_type == "regression" and wandb_run:
    if y_pred_val is not None:
        plt.figure(figsize=(8, 8))
        plt.scatter(y_val, y_pred_val, alpha=0.5)
        plt.plot([min(y_val), max(y_val)], [min(y_val), max(y_val)], 'r--')
        plt.xlabel('Actual Complexity')
        plt.ylabel('Predicted Complexity')
        plt.title('Predicted vs Actual Complexity (Val Set)')
        plt.tight_layout()
        wandb_run.log({"val_predictions_scatter": wandb.Image(plt)})
        plt.close()

    if y_pred_test is not None:
        plt.figure(figsize=(8, 8))
        plt.scatter(y_test, y_pred_test, alpha=0.5)
        plt.plot([min(y_test), max(y_test)], [min(y_test), max(y_test)], 'r--')
        plt.xlabel('Actual Complexity')
        plt.ylabel('Predicted Complexity')
        plt.title('Predicted vs Actual Complexity (Test Set)')
        plt.tight_layout()
        wandb_run.log({"test_predictions_scatter": wandb.Image(plt)})
        plt.close()

if output_dir:
    with open(os.path.join(output_dir, "results.json"), "w") as f:
        json.dump(results, f, indent=2)

    try:
        if hasattr(model, 'save_model'):
            model.save_model(os.path.join(output_dir, "model.json"))
        else:
            with open(os.path.join(output_dir, "model.pkl"), "wb") as f:
```

```

        pickle.dump(model, f)
    except Exception as e:
        print(f"Could not save model: {e}")

    return results, y_pred_val, y_pred_test

def run_experiment(config: ExperimentConfig):
    print(f"Starting experiment: {config.experiment_name}")
    print(f"Configuration: {config.to_dict()}")

    wandb_run = wandb.init(
        project="multilingual-question-probing",
        name=config.experiment_name,
        config=config.to_dict(),
        tags=[config.model_type, config.task, config.task_type] +
            ([f"control_{config.control_index}"] if config.control_index else
             [config.submetric] if config.submetric else []),
        job_type="sklearn_baseline",
        reinit=True
    )

    try:
        (X_train, y_train), (X_val, y_val), (X_test, y_test) = prepare_datasets

        print(f"Train set: {X_train.shape[0]} examples, {X_train.shape[1]} feat
        print(f"Validation set: {X_val.shape[0]} examples, {X_val.shape[1]} feat
        print(f"Test set: {X_test.shape[0]} examples, {X_test.shape[1]} feature

        model = create_sklearn_model(config.model_type, config.task_type, **cor

        results, _, _ = train_and_evaluate(
            model,
            X_train, y_train,
            X_val, y_val,
            X_test, y_test,
            config.task_type,
            output_dir=config.output_dir,
            wandb_run=wandb_run
        )

        results.update({
            "task": config.task,
            "languages": config.languages,
            "control_index": config.control_index,
            "submetric": config.submetric
        })

        with open(os.path.join(config.output_dir, "results_with_metadata.json")
                  json.dump(results, f, indent=2)

        print(f"Experiment {config.experiment_name} completed successfully")
        return results

    except Exception as e:

```

```
except Exception as e:
    print(f"Error in experiment {config.experiment_name}: {e}")
    raise

finally:
    if wandb_run:
        wandb_run.finish()

    gc.collect()

def get_default_model_params(model_type, task_type="regression"):
    if model_type == "dummy":
        if task_type == "classification":
            return {"strategy": "most_frequent"}
        if task_type == "regression":
            return {"strategy": "mean"}
        else:
            return {"strategy": "most_frequent"}

    elif model_type == "logistic":
        return {
            "C": 1.0,
            "max_iter": 1000,
            "solver": "liblinear",
            "penalty": "l2"
        }

    elif model_type == "ridge":
        return {
            "alpha": 1.0,
            "solver": "auto"
        }

    elif model_type == "xgboost":
        return {
            "n_estimators": 200,
            "max_depth": 6,
            "learning_rate": 0.1,
            "reg_alpha": 0,
            "reg_lambda": 1,
            "subsample": 0.8,
            "colsample_bytree": 0.8,
            "early_stopping_rounds": 20
        }

    else:
        return {}

def generate_question_type_experiments():
    experiments = []

    for model_type in ["dummy", "logistic", "xgboost"]:
        experiments.append(
            ExperimentConfig(
```

```
        task="question_type",
        model_type=model_type,
        languages=["all"],
        model_params=get_default_model_params(model_type, "classificati
    )
)

for control_idx in [1, 2, 3]:
    experiments.append(
        ExperimentConfig(
            task="question_type",
            model_type=model_type,
            languages=["all"],
            control_index=control_idx,
            model_params=get_default_model_params(model_type, "classifi
        )
    )

return experiments

def generate_complexity_experiments():
    experiments = []

    for model_type in ["dummy", "ridge", "xgboost"]:
        experiments.append(
            ExperimentConfig(
                task="complexity",
                model_type=model_type,
                languages=["all"],
                model_params=get_default_model_params(model_type, "regression")
            )
        )

    for control_idx in [1, 2, 3]:
        experiments.append(
            ExperimentConfig(
                task="complexity",
                model_type=model_type,
                languages=["all"],
                control_index=control_idx,
                model_params=get_default_model_params(model_type, "regressi
            )
        )

    return experiments

def generate_submetric_experiments():
    experiments = []

    for submetric in SUBMETRICS:
        for model_type in ["ridge", "xgboost"]:
            experiments.append(
                ExperimentConfig(
                    task="complexity",
```

```

        model_type=model_type,
        languages=["all"],
        submetric=submetric,
        model_params=get_default_model_params(model_type, "regressi
    )
)

return experiments

def run_all_experiments(experiment_type="all"):
    if experiment_type == "question_type":
        experiments = generate_question_type_experiments()
    elif experiment_type == "complexity":
        experiments = generate_complexity_experiments()
    elif experiment_type == "submetrics":
        experiments = generate_submetric_experiments()
    elif experiment_type == "all":
        experiments = (
            generate_question_type_experiments() +
            generate_complexity_experiments() +
            generate_submetric_experiments()
        )
    else:
        raise ValueError(f"Unknown experiment type: {experiment_type}")

    print(f"Generated {len(experiments)} {experiment_type} experiments")

    results = {}
    for i, config in enumerate(experiments):
        print(f"\nRunning experiment {i+1}/{len(experiments)}: {config.experiment_name}")

        try:
            result = run_experiment(config)
            results[config.experiment_name] = result
            print(f"Experiment {config.experiment_name} completed successfully")
        except Exception as e:
            print(f"Error in experiment {config.experiment_name}: {e}")

    with open(os.path.join(OUTPUT_DIR, f"{experiment_type}_all_results.json"),
              "w") as f:
        json.dump(results, f, indent=2)

    print(f"\nAll {experiment_type} experiments completed!")
    return results

# Debug cell - Examine TF-IDF vector structure
split = "train"
file_path = os.path.join(VECTORS_DIR, f"X_{split}.npz")
vectors = np.load(file_path, allow_pickle=True)

print(f"Vector type: {type(vectors)}")
print(f"Vector shape: {vectors.shape}")

```

```

# Inspect first element
print(f"First element type: {type(vectors[0, 0])}")
if hasattr(vectors[0, 0], 'shape'):
    print(f"First element shape: {vectors[0, 0].shape}")

# If vectors are nested, check further
if hasattr(vectors[0, 0], 'toarray'):
    # It might be a sparse matrix
    dense = vectors[0, 0].toarray()
    print(f"Sparse matrix converted to dense shape: {dense.shape}")
elif isinstance(vectors[0, 0], np.ndarray):
    print(f"Nested array structure - first element length: {len(vectors[0, 0])}")

# Print a sample of the vector values for inspection
print("\nSample vector values:")
print(vectors[0, 0][:10] if isinstance(vectors[0, 0], np.ndarray) else vectors[

# Test the fixed loading function
X_train = load_tfidf_features("train")
print(f"Final shape: {X_train.shape}")

# Test with a simple model
from sklearn.dummy import DummyClassifier
dummy = DummyClassifier(strategy="most_frequent")
y_dummy = np.zeros(X_train.shape[0]) # Just dummy labels
dummy.fit(X_train, y_dummy)
print("Model fitting successful!")

```

```

Vector type: <class 'numpy.ndarray'>
Vector shape: (7460, 1)
First element type: <class 'scipy.sparse._csr.csr_matrix'>
First element shape: (1, 128104)
Sparse matrix converted to dense shape: (1, 128104)

```

Sample vector values:

```

<Compressed Sparse Row sparse matrix of dtype 'float64'
  with 15 stored elements and shape (1, 128104)>

```

Coords	Values
(0, 24)	0.21821789023599242
(0, 56)	0.21821789023599242
(0, 59)	0.21821789023599242
(0, 245)	0.21821789023599242
(0, 266)	0.21821789023599242
(0, 272)	0.21821789023599242
(0, 336)	0.21821789023599242
(0, 743)	0.43643578047198484
(0, 3073)	0.21821789023599242
(0, 4267)	0.43643578047198484
(0, 5686)	0.21821789023599242
(0, 24302)	0.21821789023599242
(0, 25703)	0.21821789023599242
(0, 36454)	0.21821789023599242
(0, 86932)	0.21821789023599242

```

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (7460, 1)
Final shape: (7460, 128104)
Model fitting successful!

```

```
wandb.login()

# Example 1: single experiment on all languages (no controls)
config_question_type = ExperimentConfig(
    task="question_type",
    model_type="logistic",
    languages=["all"],
    model_params=get_default_model_params("logistic")
)
#run_experiment(config_question_type) # Uncomment to run

# single experiment on all languages with control
config_question_type_control = ExperimentConfig(
    task="question_type",
    model_type="logistic",
    languages=["all"],
    control_index=1,
    model_params=get_default_model_params("logistic")
)
#run_experiment(config_question_type_control) # Uncomment to run

# Example 3: single complexity experiment
config_complexity = ExperimentConfig(
    task="complexity",
    model_type="xgboost",
    languages=["all"],
    model_params={
        "n_estimators": 200,
        "max_depth": 6,
        "learning_rate": 0.1,
        "subsample": 0.8,
        "colsample_bytree": 0.8,
        "early_stopping_rounds": 20,
        "eval_metric": ["rmse", "mae"]
    }
)
#run_experiment(config_complexity)

#Run a single submetric experiment
config_submetric = ExperimentConfig(
    task="complexity",
    model_type="ridge",
    languages=["all"], # Use all languages
    submetric="n_tokens", # Focus on token count submetric
    model_params=get_default_model_params("ridge")
)
#run_experiment(config_submetric) # Uncomment to run

# run_all_experiments("question_type") # question type experiments (base + cont
#run_all_experiments("complexity") # complexity experiments (base + controls
#run_all_experiments("submetrics") # submetric experiments
run_all_experiments("all") # experiments combined
```

wandb: Using wandb-core as the SDK backend. Please refer to <https://wandb.ai>.
wandb: Logging into wandb.ai. (Learn how to deploy a W&B server locally: <https://wandb.ai/aut>)
wandb: You can find your API key in your browser here: <https://wandb.ai/aut>
wandb: Paste an API key from your profile and hit enter:
wandb: **WARNING** If you're specifying your api key in code, ensure this code
wandb: **WARNING** Consider setting the WANDB_API_KEY environment variable, or
wandb: No netrc file found, creating one.
wandb: Appending key for api.wandb.ai to your netrc file: /root/.netrc
wandb: Currently logged in as: **rokii** (**rokii-ku-leuven**) to <https://api.wandb.ai>
 Generated 36 all experiments

Running experiment 1/36: dummy_question_type_all
 Starting experiment: dummy_question_type_all
 Configuration: {'task': 'question_type', 'model_type': 'dummy', 'languages': 'all'}
wandb: **WARNING** Using a boolean value for 'reinit' is deprecated. Use 'return' instead.
 Tracking run with wandb version 0.19.9
 Run data is saved locally in /content/wandb/run-20250410_115704-ox6rnnvj9
 Syncing run **dummy_question_type_all** to [Weights & Biases \(docs\)](https://wandb.ai/rokii-ku-leuven/multilingual-question-probing)
 View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>
 View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/ox6rnnvj9>
 Using dataset config: base
 Loaded dataset splits - Train: 7460, Validation: 441, Test: 719
 Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
 Loaded vectors shape: (7460, 1)
 Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
 Loaded vectors shape: (441, 1)
 Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
 Loaded vectors shape: (719, 1)
 Train features: (7460, 128104), Train samples: 7460
 Dev features: (441, 128104), Dev samples: 441
 Test features: (719, 128104), Test samples: 719
 Train set: 7460 examples, 128104 features
 Validation set: 441 examples, 128104 features
 Test set: 719 examples, 128104 features
 Creating dummy model for classification task
 Training DummyClassifier on 7460 examples, 128104 features
 Training completed in 0.00 seconds
 Validation metrics: {'accuracy': 0.5056689342403629, 'f1': 0.0}
 Test metrics: {'accuracy': 0.5104311543810849, 'f1': 0.0}
 Experiment dummy_question_type_all completed successfully

Run history:

test_accuracy	—
test_f1	—
training_time	—
val_accuracy	—
val_f1	—

Run summary:

test_accuracy: 0.51043


```
test_accuracy 0.51043
test_f1       0
training_time 0.00112
val_accuracy  0.50567
val_f1        0
```

View run **dummy_question_type_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/ox6rnvj9>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 0 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_115704-ox6rnvj9/logs

Experiment dummy_question_type_all completed successfully

Running experiment 2/36: dummy_question_type_control1_all

Starting experiment: dummy_question_type_control1_all

Configuration: {'task': 'question_type', 'model_type': 'dummy', 'languages'

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_115710-ne8aldj2

Syncing run **dummy_question_type_control1_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/ne8aldj2>

Using dataset config: control_question_type_seed1

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating dummy model for classification task

Training DummyClassifier on 7460 examples, 128104 features

Training completed in 0.00 seconds

Validation metrics: {'accuracy': 0.5056689342403629, 'f1': 0.0}

Test metrics: {'accuracy': 0.5104311543810849, 'f1': 0.0}

Experiment dummy_question_type_control1_all completed successfully

Run history:

```
test_accuracy _
test_f1       _
training_time _
val_accuracy  _
val_f1        _
```

Run summary:

```
test accuracy 0.51043
```

```

test_f1      0
training_time 0.00084
val_accuracy 0.50567
val_f1       0

```

View run **dummy_question_type_control1_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/ne8aldj2>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 0 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_115710-ne8aldj2/logs

Experiment dummy_question_type_control1_all completed successfully

Running experiment 3/36: dummy_question_type_control2_all

Starting experiment: dummy_question_type_control2_all

Configuration: {'task': 'question_type', 'model_type': 'dummy', 'languages'

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_115717-uy0ep5aw

Syncing run **dummy_question_type_control2_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/uy0ep5aw>

Using dataset config: control_question_type_seed2

```

(...)di_train_control_question_type_seed2.csv: 100%          1.03M/1.03M [00:00<00:00, 54.7MB/
s]

```

Generating train split: 7460/0 [00:00<00:00, 81651.71 examples/s]

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating dummy model for classification task

Training DummyClassifier on 7460 examples, 128104 features

Training completed in 0.00 seconds

Validation metrics: {'accuracy': 0.5056689342403629, 'f1': 0.0}

Test metrics: {'accuracy': 0.5104311543810849, 'f1': 0.0}

Experiment dummy_question_type_control2_all completed successfully

Run history:

```

test_accuracy _
test_f1       _
training_time _
val_accuracy  _
val_f1       _

```

Run summary:

```
test_accuracy 0.51043
test_f1       0
training_time 0.00095
val_accuracy  0.50567
val_f1        0
```

View run **dummy_question_type_control2_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/uy0ep5aw>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 0 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_115717-uy0ep5aw/logs

Experiment dummy_question_type_control2_all completed successfully

Running experiment 4/36: dummy_question_type_control3_all

Starting experiment: dummy_question_type_control3_all

Configuration: {'task': 'question_type', 'model_type': 'dummy', 'languages'

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_115723-xqmtg5so

Syncing run **dummy_question_type_control3_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/xqmtg5so>

Using dataset config: control_question_type_seed3

```
(...)di_train_control_question_type_seed3.csv: 100%          1.03M/1.03M [00:00<00:00, 16.6MB/s]
```

Generating train split: 7460/0 [00:00<00:00, 69842.18 examples/s]

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating dummy model for classification task

Training DummyClassifier on 7460 examples, 128104 features

Training completed in 0.00 seconds

Validation metrics: {'accuracy': 0.5056689342403629, 'f1': 0.0}

Test metrics: {'accuracy': 0.5104311543810849, 'f1': 0.0}

Experiment dummy_question_type_control3_all completed successfully

Run history:

```
test_accuracy _
. . .
```

```
test_f1      _
training_time _
val_accuracy _
val_f1       _
```

Run summary:

```
test_accuracy 0.51043
test_f1        0
training_time  0.00093
val_accuracy   0.50567
val_f1         0
```

View run **dummy_question_type_control3_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/xqmtg5so>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 0 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_115723-xqmtg5so/logs

Experiment dummy_question_type_control3_all completed successfully

Running experiment 5/36: logistic_question_type_all

Starting experiment: logistic_question_type_all

Configuration: {'task': 'question_type', 'model_type': 'logistic', 'language': 'all'}

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_115728-yj2jmnzw

Syncing run **logistic_question_type_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/yj2jmnzw>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating logistic model for classification task

Training LogisticRegression on 7460 examples, 128104 features

```
/usr/local/lib/python3.11/dist-packages/sklearn/linear_model/_logistic.py:1
warnings.warn(
```

Training completed in 0.10 seconds

Validation metrics: {'accuracy': 0.8662131519274376, 'f1': 0.86310904872389}

Test metrics: {'accuracy': 0.7510431154381085, 'f1': 0.7672301690507152}

Experiment logistic_question_type_all completed successfully

Run history:

```
test_accuracy _
test_f1 _
training_time _
val_accuracy _
val_f1 _
```

Run summary:

```
test_accuracy 0.75104
test_f1 0.76723
training_time 0.09629
val_accuracy 0.86621
val_f1 0.86311
```

View run **logistic_question_type_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/yj2jmnzw>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 0 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_115728-yj2jmnzw/logs

Experiment logistic_question_type_all completed successfully

Running experiment 6/36: logistic_question_type_control1_all

Starting experiment: logistic_question_type_control1_all

Configuration: {'task': 'question_type', 'model_type': 'logistic', 'language': 'all'}

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_115732-zbcao07w

Syncing run **logistic_question_type_control1_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/zbcao07w>

Using dataset config: control_question_type_seed1

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating logistic model for classification task

Training LogisticRegression on 7460 examples, 128104 features

```
/usr/local/lib/python3.11/dist-packages/sklearn/linear_model/_logistic.py:1
warnings.warn(
```

Training completed in 0.63 seconds

Validation metrics: {'accuracy': 0.5147392290249433, 'f1': 0.46231155778894}

Test metrics: {'accuracy': 0.48400556328233657, 'f1': 0.4370257966616085}

Experiment logistic_question_type_control1_all completed successfully

Run history:

```
run_metrics:
```

```
test_accuracy _
test_f1 _
training_time _
val_accuracy _
val_f1 _
```

Run summary:

```
test_accuracy 0.48401
test_f1 0.43703
training_time 0.63019
val_accuracy 0.51474
val_f1 0.46231
```

View run **logistic_question_type_control1_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/zbcas07w>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 0 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_115732-zbcas07w/logs

Experiment logistic_question_type_control1_all completed successfully

Running experiment 7/36: logistic_question_type_control2_all

Starting experiment: logistic_question_type_control2_all

Configuration: {'task': 'question_type', 'model_type': 'logistic', 'language': 'multilingual'}

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_115739-u42hj5ie

Syncing run **logistic_question_type_control2_all** to [Weights & Biases](https://wandb.ai/rokii-ku-leuven/multilingual-question-probing) ([docs](#))

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/u42hj5ie>

Using dataset config: control_question_type_seed2

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating logistic model for classification task

Training LogisticRegression on 7460 examples, 128104 features

Training completed in 0.11 seconds

Validation metrics: {'accuracy': 0.49206349206349204, 'f1': 0.4641148325358}

Test metrics: {'accuracy': 0.46870653685674546, 'f1': 0.4415204678362573}

/usr/local/lib/python3.11/dist-packages/sklearn/linear_model/_logistic.py:1

warnings.warn(

Experiment logistic_question_type_control2_all completed successfully

Run history:

```
test_accuracy _
test_f1       _
training_time _
val_accuracy  _
val_f1        _
```

Run summary:

```
test_accuracy 0.46871
test_f1        0.44152
training_time  0.10854
val_accuracy   0.49206
val_f1         0.46411
```

View run **logistic_question_type_control2_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/u42hj5ie>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 0 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_115739-u42hj5ie/logs

Experiment logistic_question_type_control2_all completed successfully

Running experiment 8/36: logistic_question_type_control3_all

Starting experiment: logistic_question_type_control3_all

Configuration: {'task': 'question_type', 'model_type': 'logistic', 'language': 'dutch'}

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_115743-6szipioc

Syncing run **logistic_question_type_control3_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/6szipioc>

Using dataset config: control_question_type_seed3

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating logistic model for classification task

Training LogisticRegression on 7460 examples, 128104 features

```
/usr/local/lib/python3.11/dist-packages/sklearn/linear_model/_logistic.py:1
warnings.warn(
```

Training completed in 0.12 seconds

Validation metrics: {'accuracy': 0.5306122448979592, 'f1': 0.50596658711217}

Test metrics: {'accuracy': 0.5187760778859527, 'f1': 0.5167597765363129}

Experiment logistic_question_type_control3_all completed successfully

Run history:

```
test_accuracy _
test_f1       _
training_time _
val_accuracy  _
val_f1        _
```

Run summary:

```
test_accuracy 0.51878
test_f1        0.51676
training_time  0.1243
val_accuracy   0.53061
val_f1         0.50597
```

View run **logistic_question_type_control3_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/6szipioc>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 0 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_115743-6szipioc/logs

Experiment logistic_question_type_control3_all completed successfully

Running experiment 9/36: xgboost_question_type_all

Starting experiment: xgboost_question_type_all

Configuration: {'task': 'question_type', 'model_type': 'xgboost', 'language

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_115748-ig427sj6

Syncing run **xgboost_question_type_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/ig427sj6>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating xgboost model for classification task

Training XGBClassifier on 7460 examples, 128104 features

/usr/local/lib/python3.11/dist-packages/xgboost/core.py:158: UserWarning: [

Parameters: { "use_label_encoder" } are not used.

```
warnings.warn(msg, UserWarning)
```

```
[0] validation 0-ldloss:0.66821 validation 1-ldloss:0.67287
```



```
[100] validation_0-logloss:0.19327 validation_1-logloss:0.35781
[199] validation_0-logloss:0.11859 validation_1-logloss:0.30881
Training completed in 151.01 seconds
Validation metrics: {'accuracy': 0.8616780045351474, 'f1': 0.85647058823529
Test metrics: {'accuracy': 0.7649513212795549, 'f1': 0.7623066104078763}
Experiment xgboost_question_type_all completed successfully
```

Run history:

```
test_accuracy _
test_f1 _
training_time _
val_accuracy _
val_f1 _
```

Run summary:

```
test_accuracy 0.76495
test_f1 0.76231
training_time 151.00834
val_accuracy 0.86168
val_f1 0.85647
```

View run **xgboost_question_type_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/ig427sj6>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_115748-ig427sj6/logs

Experiment xgboost_question_type_all completed successfully

Running experiment 10/36: xgboost_question_type_control1_all

Starting experiment: xgboost_question_type_control1_all

Configuration: {'task': 'question_type', 'model_type': 'xgboost', 'language

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_120026-419mj14b

Syncing run **xgboost_question_type_control1_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/419mj14b>

Using dataset config: control_question_type_seed1

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

```

Creating xgboost model for classification task
Training XGBClassifier on 7460 examples, 128104 features
/usr/local/lib/python3.11/dist-packages/xgboost/core.py:158: UserWarning: [
Parameters: { "use_label_encoder" } are not used.

```

```

warnings.warn(msg, UserWarning)
[0]    validation_0-logloss:0.69192    validation_1-logloss:0.69231
[100]  validation_0-logloss:0.63765    validation_1-logloss:0.68640
[199]  validation_0-logloss:0.60339    validation_1-logloss:0.69212
Training completed in 151.28 seconds
Validation metrics: {'accuracy': 0.5215419501133787, 'f1': 0.44908616187989}
Test metrics: {'accuracy': 0.502086230876217, 'f1': 0.39730639730639733}
Experiment xgboost_question_type_control1_all completed successfully

```

Run history:

```

test_accuracy _
test_f1       _
training_time _
val_accuracy  _
val_f1        _

```

Run summary:

```

test_accuracy 0.50209
test_f1       0.39731
training_time 151.28243
val_accuracy  0.52154
val_f1        0.44909

```

View run **xgboost_question_type_control1_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/419mj14b>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_120026-419mj14b/logs

Experiment xgboost_question_type_control1_all completed successfully

Running experiment 11/36: xgboost_question_type_control2_all

Starting experiment: xgboost_question_type_control2_all

Configuration: {'task': 'question_type', 'model_type': 'xgboost', 'language': 'en'}

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_120304-3lad9g5r

Syncing run **xgboost_question_type_control2_all** to [Weights & Biases](https://wandb.ai/rokii-ku-leuven/multilingual-question-probing) (docs)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/3lad9g5r>

Using dataset config: control_question_type_seed2

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

```

Loaded vectors shape: (719, 1)
Train features: (7460, 128104), Train samples: 7460
Dev features: (441, 128104), Dev samples: 441
Test features: (719, 128104), Test samples: 719
Train set: 7460 examples, 128104 features
Validation set: 441 examples, 128104 features
Test set: 719 examples, 128104 features
Creating xgboost model for classification task
Training XGBClassifier on 7460 examples, 128104 features
/usr/local/lib/python3.11/dist-packages/xgboost/core.py:158: UserWarning: [
Parameters: { "use_label_encoder" } are not used.

    warnings.warn(msg, UserWarning)
[0]    validation_0-logloss:0.69243  validation_1-logloss:0.69338
[100]  validation_0-logloss:0.64103  validation_1-logloss:0.69920
[199]  validation_0-logloss:0.60811  validation_1-logloss:0.70559
Training completed in 150.18 seconds
Validation metrics: {'accuracy': 0.5419501133786848, 'f1': 0.52803738317757}
Test metrics: {'accuracy': 0.5006954102920723, 'f1': 0.5102319236016372}
Experiment xgboost_question_type_control2_all completed successfully

```

Run history:

```

test_accuracy _
test_f1       _
training_time _
val_accuracy  _
val_f1        _

```

Run summary:

```

test_accuracy 0.5007
test_f1       0.51023
training_time 150.18313
val_accuracy  0.54195
val_f1        0.52804

```

View run **xgboost_question_type_control2_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/3lad9g5r>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_120304-3lad9g5r/logs

Experiment xgboost_question_type_control2_all completed successfully

Running experiment 12/36: xgboost_question_type_control3_all

Starting experiment: xgboost_question_type_control3_all

Configuration: {'task': 'question_type', 'model_type': 'xgboost', 'language

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_120541-zlkfiwvb

Syncing run **xgboost_question_type_control3_all** to [Weights & Biases](#) (docs)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/zlkfiwvb>

Using dataset config: control question type seed3

```

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (7460, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (441, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (719, 1)
Train features: (7460, 128104), Train samples: 7460
Dev features: (441, 128104), Dev samples: 441
Test features: (719, 128104), Test samples: 719
Train set: 7460 examples, 128104 features
Validation set: 441 examples, 128104 features
Test set: 719 examples, 128104 features
Creating xgboost model for classification task
Training XGBClassifier on 7460 examples, 128104 features
/usr/local/lib/python3.11/dist-packages/xgboost/core.py:158: UserWarning: [
Parameters: { "use_label_encoder" } are not used.

    warnings.warn(smsg, UserWarning)
[0]    validation_0-logloss:0.69219  validation_1-logloss:0.69091
[100] validation_0-logloss:0.64147  validation_1-logloss:0.69112
[199] validation_0-logloss:0.60854  validation_1-logloss:0.68978
Training completed in 150.21 seconds
Validation metrics: {'accuracy': 0.5578231292517006, 'f1': 0.53681710213776}
Test metrics: {'accuracy': 0.5354659248956884, 'f1': 0.49393939393939396}
Experiment xgboost_question_type_control3_all completed successfully

```

Run history:

```

test_accuracy _
test_f1       _
training_time _
val_accuracy  _
val_f1        _

```

Run summary:

```

test_accuracy 0.53547
test_f1       0.49394
training_time 150.21125
val_accuracy  0.55782
val_f1        0.53682

```

View run **xgboost_question_type_control3_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/zlkfiwvb>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_120541-zlkfiwvb/logs

Experiment xgboost_question_type_control3_all completed successfully

Running experiment 13/36: dummy_complexity_all

Starting experiment: dummy_complexity_all

```
Configuration: {'task': 'complexity', 'model_type': 'dummy', 'languages': [
Tracking run with wandb version 0.19.9
Run data is saved locally in /content/wandb/run-20250410_120817-ij3vgwhg
Syncing run dummy\_complexity\_all to Weights & Biases (docs)
View project at https://wandb.ai/rokii-ku-leuven/multilingual-question-probing
View run at https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/ij3vgwhg
Using dataset config: base
Loaded dataset splits - Train: 7460, Validation: 441, Test: 719
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (7460, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (441, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (719, 1)
Train features: (7460, 128104), Train samples: 7460
Dev features: (441, 128104), Dev samples: 441
Test features: (719, 128104), Test samples: 719
Train set: 7460 examples, 128104 features
Validation set: 441 examples, 128104 features
Test set: 719 examples, 128104 features
Creating dummy model for regression task
Training DummyRegressor on 7460 examples, 128104 features
Training completed in 0.00 seconds
Validation metrics: {'mse': 0.05940214544534683, 'rmse': 0.2437255535337787
Test metrics: {'mse': 0.04718531668186188, 'rmse': 0.21722181447051278, 'ma
Experiment dummy_complexity_all completed successfully
```

Run history:

test_mae	—
test_mse	—
test_r2	—
test_rmse	—
training_time	—
val_mae	—
val_mse	—
val_r2	—
val_rmse	—

Run summary:

test_mae	0.17158
test_mse	0.04719
test_r2	-0.04262
test_rmse	0.21722
training_time	0.00085
val_mae	0.19819
val_mse	0.0594
val_r2	-0.06715
val_rmse	0.24373

View run **dummy_complexity_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/ij3vgwhg>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_120817-ij3vgwhg/logs

Experiment dummy_complexity_all completed successfully

Running experiment 14/36: dummy_complexity_control1_all

Starting experiment: dummy_complexity_control1_all

Configuration: {'task': 'complexity', 'model_type': 'dummy', 'languages': [

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_120823-287885sf

Syncing run **dummy_complexity_control1_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/287885sf>

Using dataset config: control_complexity_seed1

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating dummy model for regression task

Training DummyRegressor on 7460 examples, 128104 features

Training completed in 0.00 seconds

Validation metrics: {'mse': 0.05940214544534683, 'rmse': 0.2437255535337787

Test metrics: {'mse': 0.04718531668186188, 'rmse': 0.21722181447051278, 'ma

Experiment dummy_complexity_control1_all completed successfully

Run history:

test_mae	—
test_mse	—
test_r2	—
test_rmse	—
training_time	—
val_mae	—
val_mse	—
val_r2	—
val_rmse	—

Run summary:

test_mae	0.17158
test_mse	0.04719

```

test_r2      -0.04262
test_rmse    0.21722
training_time 0.0007
val_mae      0.19819
val_mse      0.0594
val_r2       -0.06715
val_rmse     0.24373

```

View run **dummy_complexity_control1_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/287885sf>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_120823-287885sf/logs

Experiment dummy_complexity_control1_all completed successfully

Running experiment 15/36: dummy_complexity_control2_all

Starting experiment: dummy_complexity_control2_all

Configuration: {'task': 'complexity', 'model_type': 'dummy', 'languages': [

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_120829-b5fonwhy

Syncing run **dummy_complexity_control2_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/b5fonwhy>

Using dataset config: control_complexity_seed2

```

tydi_train_control_complexity_seed2.csv: 100%          1.03M/1.03M [00:00<00:00, 28.3MB/
s]

```

Generating train split: 7460/0 [00:00<00:00, 69279.14 examples/s]

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating dummy model for regression task

Training DummyRegressor on 7460 examples, 128104 features

Training completed in 0.00 seconds

Validation metrics: {'mse': 0.05940214544534683, 'rmse': 0.2437255535337787

Test metrics: {'mse': 0.04718531668186188, 'rmse': 0.21722181447051278, 'ma

Experiment dummy_complexity_control2_all completed successfully

Run history:

```

test_mae      _
test_mse      _
test_r2

```



```
test_rmse    _
training_time _
val_mae      _
val_rmse     _
val_r2       _
val_rmse     _
```

Run summary:

```
test_mae      0.17158
test_rmse     0.04719
test_r2       -0.04262
test_rmse     0.21722
training_time 0.00071
val_mae       0.19819
val_rmse      0.0594
val_r2        -0.06715
val_rmse      0.24373
```

View run **dummy_complexity_control2_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/b5fonwhy>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_120829-b5fonwhy/logs

Experiment dummy_complexity_control2_all completed successfully

Running experiment 16/36: dummy_complexity_control3_all

Starting experiment: dummy_complexity_control3_all

Configuration: {'task': 'complexity', 'model_type': 'dummy', 'languages': [

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_120834-e2wo04ve

Syncing run **dummy_complexity_control3_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/e2wo04ve>

Using dataset config: control_complexity_seed3

```
tydi_train_control_complexity_seed3.csv: 100%          1.03M/1.03M [00:00<00:00, 5.36MB/
s]
```

Generating train split: 7460/0 [00:00<00:00, 82989.44 examples/s]

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features


```

Creating dummy model for regression task
Training DummyRegressor on 7460 examples, 128104 features
Training completed in 0.00 seconds
Validation metrics: {'mse': 0.05940214544534683, 'rmse': 0.2437255535337787
Test metrics: {'mse': 0.04718531668186188, 'rmse': 0.21722181447051278, 'ma
Experiment dummy_complexity_control3_all completed successfully

```

Run history:

```

test_mae    _
test_mse    _
test_r2     _
test_rmse   _
training_time _
val_mae     _
val_mse     _
val_r2      _
val_rmse    _

```

Run summary:

```

test_mae    0.17158
test_mse    0.04719
test_r2     -0.04262
test_rmse   0.21722
training_time 0.00085
val_mae     0.19819
val_mse     0.0594
val_r2      -0.06715
val_rmse    0.24373

```

View run **dummy_complexity_control3_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/e2wo04ve>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_120834-e2wo04ve/logs

Experiment dummy_complexity_control3_all completed successfully

Running experiment 17/36: ridge_complexity_all

Starting experiment: ridge_complexity_all

Configuration: {'task': 'complexity', 'model_type': 'ridge', 'languages': [

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_120841-czhpjix0

Syncing run **ridge_complexity_all** to [Weights & Biases](#) ([docs](#))

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/czhpjix0>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/otvne-eval/data

```

Loading vectors from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (441, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (719, 1)
Train features: (7460, 128104), Train samples: 7460
Dev features: (441, 128104), Dev samples: 441
Test features: (719, 128104), Test samples: 719
Train set: 7460 examples, 128104 features
Validation set: 441 examples, 128104 features
Test set: 719 examples, 128104 features
Creating ridge model for regression task
Training Ridge on 7460 examples, 128104 features
Training completed in 0.11 seconds
Validation metrics: {'mse': 0.02834229564475883, 'rmse': 0.1683517022330301}
Test metrics: {'mse': 0.02617858559723656, 'rmse': 0.16179797772913157, 'ma
Experiment ridge_complexity_all completed successfully

```

Run history:

```

test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_mse       _
val_r2        _
val_rmse      _

```

Run summary:

```

test_mae      0.12908
test_mse      0.02618
test_r2       0.42155
test_rmse     0.1618
training_time 0.1122
val_mae       0.13134
val_mse       0.02834
val_r2        0.49084
val_rmse      0.16835

```

View run **ridge_complexity_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/czhpjixo>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_120841-czhpjixo/logs

Experiment ridge_complexity_all completed successfully

Running experiment 18/36: ridge_complexity_control1_all

Starting experiment: ridge_complexity_control1_all

Configuration: {'task': 'complexity', 'model_type': 'ridge', 'languages': [

Training run with wandb version 0.10.0

```

Tracking run with wandb version 0.19.9
Run data is saved locally in /content/wandb/run-20250410_120847-h3rwtqv8
Syncing run ridge_complexity_control1_all to Weights & Biases \(docs\)
View project at https://wandb.ai/rokii-ku-leuven/multilingual-question-probing
View run at https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/h3rwtqv8
Using dataset config: control_complexity_seed1
Loaded dataset splits - Train: 7460, Validation: 441, Test: 719
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (7460, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (441, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (719, 1)
Train features: (7460, 128104), Train samples: 7460
Dev features: (441, 128104), Dev samples: 441
Test features: (719, 128104), Test samples: 719
Train set: 7460 examples, 128104 features
Validation set: 441 examples, 128104 features
Test set: 719 examples, 128104 features
Creating ridge model for regression task
Training Ridge on 7460 examples, 128104 features
Training completed in 0.16 seconds
Validation metrics: {'mse': 0.06315823313570658, 'rmse': 0.2513130182376284}
Test metrics: {'mse': 0.04897569807893945, 'rmse': 0.22130453695968244, 'ma
Experiment ridge_complexity_control1_all completed successfully

```

Run history:

```

test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_mse       _
val_r2        _
val_rmse      _

```

Run summary:

```

test_mae      0.17379
test_mse      0.04898
test_r2       -0.08218
test_rmse     0.2213
training_time 0.15995
val_mae       0.20248
val_mse       0.06316
val_r2        -0.13463
val_rmse      0.25131

```

View run **ridge_complexity_control1_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question->

[probing/runs/h3rwtqv8](#)

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: `./wandb/run-20250410_120847-h3rwtqv8/logs`

Experiment `ridge_complexity_control1_all` completed successfully

Running experiment 19/36: `ridge_complexity_control2_all`

Starting experiment: `ridge_complexity_control2_all`

Configuration: `{'task': 'complexity', 'model_type': 'ridge', 'languages': [`

Tracking run with wandb version 0.19.9

Run data is saved locally in `/content/wandb/run-20250410_120852-mmsy4o7o`

Syncing run [ridge_complexity_control2_all](#) to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/mmsy4o7o>

Using dataset config: `control_complexity_seed2`

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from `/content/drive/MyDrive/ColabNotebooks/qtype-eval/data`

Loaded vectors shape: (7460, 1)

Loading features from `/content/drive/MyDrive/ColabNotebooks/qtype-eval/data`

Loaded vectors shape: (441, 1)

Loading features from `/content/drive/MyDrive/ColabNotebooks/qtype-eval/data`

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating ridge model for regression task

Training Ridge on 7460 examples, 128104 features

Training completed in 0.07 seconds

Validation metrics: `{'mse': 0.06119368838813863, 'rmse': 0.2473735806187448`

Test metrics: `{'mse': 0.04839423261099806, 'rmse': 0.2199868919072181, 'mae`

Experiment `ridge_complexity_control2_all` completed successfully

Run history:

test_mae	—
test_mse	—
test_r2	—
test_rmse	—
training_time	—
val_mae	—
val_mse	—
val_r2	—
val_rmse	—

Run summary:

test_mae	0.17601
test_mse	0.04839
test_r2	-0.06933

```

test_rmse    0.21999
training_time 0.06926
val_mae      0.20175
val_rmse     0.06119
val_r2       -0.09933
val_rmse     0.24737

```

View run **ridge_complexity_control2_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/mmsy4o7o>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_120852-mmsy4o7o/logs

Experiment ridge_complexity_control2_all completed successfully

Running experiment 20/36: ridge_complexity_control3_all

Starting experiment: ridge_complexity_control3_all

Configuration: {'task': 'complexity', 'model_type': 'ridge', 'languages': [

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_120858-h6q35qh7

Syncing run **ridge_complexity_control3_all** to [Weights & Biases](#) ([docs](#))

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/h6q35qh7>

Using dataset config: control_complexity_seed3

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating ridge model for regression task

Training Ridge on 7460 examples, 128104 features

Training completed in 0.09 seconds

Validation metrics: {'mse': 0.06666476686615211, 'rmse': 0.258195210773074,

Test metrics: {'mse': 0.05271930345372578, 'rmse': 0.229606845398228, 'mae'

Experiment ridge_complexity_control3_all completed successfully

Run history:

```

test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_rmse      _
val_r2        _

```

val_rmse —

Run summary:

```
test_mae    0.18119
test_mse    0.05272
test_r2     -0.1649
test_rmse   0.22961
training_time 0.08811
val_mae     0.20974
val_mse     0.06666
val_r2      -0.19762
val_rmse    0.2582
```

View run **ridge_complexity_control3_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/h6q35qh7>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_120858-h6q35qh7/logs

Experiment ridge_complexity_control3_all completed successfully

Running experiment 21/36: xgboost_complexity_all

Starting experiment: xgboost_complexity_all

Configuration: {'task': 'complexity', 'model_type': 'xgboost', 'languages':

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_120904-lm8arqhj

Syncing run **xgboost_complexity_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/lm8arqhj>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating xgboost model for regression task

Training XGBRegressor on 7460 examples, 128104 features

[0] validation_0-rmse:0.16577 validation_0-mae:0.13306 validation_1-rmse

[100] validation_0-rmse:0.09543 validation_0-mae:0.07481 validation_1-rmse

[199] validation_0-rmse:0.08222 validation_0-mae:0.06437 validation_1-rmse

Training completed in 176.20 seconds

Validation metrics: {'mse': 0.025757428258657455, 'rmse': 0.160491209287790

Test metrics: {'mse': 0.022604431957006454, 'rmse': 0.15034770353087024, 'm

Experiment xgboost_complexity_all completed successfully

Run history:

```

test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_mse       _
val_r2        _
val_rmse      _

```

Run summary:

```

test_mae      0.11821
test_mse      0.0226
test_r2       0.50053
test_rmse     0.15035
training_time 176.19617
val_mae       0.12436
val_mse       0.02576
val_r2        0.53727
val_rmse      0.16049

```

View run **xgboost_complexity_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/lm8arqhj>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 4 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_120904-lm8arqhj/logs

Experiment xgboost_complexity_all completed successfully

Running experiment 22/36: xgboost_complexity_control1_all

Starting experiment: xgboost_complexity_control1_all

Configuration: {'task': 'complexity', 'model_type': 'xgboost', 'languages':

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_121209-py9luwtf

Syncing run **xgboost_complexity_control1_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/py9luwtf>

Using dataset config: control_complexity_seed1

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features


```

test set: 719 examples, 128104 features
Creating xgboost model for regression task
Training XGBRegressor on 7460 examples, 128104 features
[0]   validation_0-rmse:0.16877 validation_0-mae:0.13539 validation_1-rmse
[100] validation_0-rmse:0.15196 validation_0-mae:0.12343 validation_1-rmse
[199] validation_0-rmse:0.14258 validation_0-mae:0.11638 validation_1-rmse
Training completed in 167.22 seconds
Validation metrics: {'mse': 0.05997728556394577, 'rmse': 0.2449026042408405}
Test metrics: {'mse': 0.04638424143195152, 'rmse': 0.21537001052131544, 'mae': 0.17064}
Experiment xgboost_complexity_control1_all completed successfully

```

Run history:

```

test_mae      _
test_rmse     _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_rmse      _
val_r2        _
val_rmse      _

```

Run summary:

```

test_mae      0.17064
test_rmse     0.04638
test_r2       -0.02492
test_rmse     0.21537
training_time 167.2183
val_mae       0.1997
val_rmse      0.05998
val_r2        -0.07748
val_rmse      0.2449

```

View run **xgboost_complexity_control1_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/py9luwtf>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 4 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_121209-py9luwtf/logs

Experiment xgboost_complexity_control1_all completed successfully

Running experiment 23/36: xgboost_complexity_control2_all

Starting experiment: xgboost_complexity_control2_all

Configuration: {'task': 'complexity', 'model_type': 'xgboost', 'languages': 'all'}

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_121503-1ezdyk3p

Syncing run **xgboost_complexity_control2_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/1ezdyk3p>

Using dataset config: control_complexity_seed2


```

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (7460, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (441, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (719, 1)
Train features: (7460, 128104), Train samples: 7460
Dev features: (441, 128104), Dev samples: 441
Test features: (719, 128104), Test samples: 719
Train set: 7460 examples, 128104 features
Validation set: 441 examples, 128104 features
Test set: 719 examples, 128104 features
Creating xgboost model for regression task
Training XGBRegressor on 7460 examples, 128104 features
[0]    validation_0-rmse:0.16885 validation_0-mae:0.13544 validation_1-rmse
[100]  validation_0-rmse:0.15303 validation_0-mae:0.12381 validation_1-rmse
[199]  validation_0-rmse:0.14413 validation_0-mae:0.11703 validation_1-rmse
Training completed in 163.41 seconds
Validation metrics: {'mse': 0.06018916890025139, 'rmse': 0.2453348098013231}
Test metrics: {'mse': 0.04748661443591118, 'rmse': 0.21791423642321117, 'ma
Experiment xgboost_complexity_control2_all completed successfully

```

Run history:

```

test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_mse       _
val_r2        _
val_rmse      _

```

Run summary:

```

test_mae      0.17359
test_mse      0.04749
test_r2       -0.04928
test_rmse     0.21791
training_time 163.41402
val_mae       0.19948
val_mse       0.06019
val_r2        -0.08129
val_rmse      0.24533

```

View run **xgboost_complexity_control2_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/1ezdyk3p>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synched 5 W&R file(s) 4 media file(s) 2 artifact file(s) and 0 other file(s)

```

Synchro 0 was me(0), 1 media me(0), 2 extract me(0), and 3 other me(0)
Find logs at: ./wandb/run-20250410_121503-lezdyk3p/logs
Experiment xgboost_complexity_control2_all completed successfully

```

```

Running experiment 24/36: xgboost_complexity_control3_all
Starting experiment: xgboost_complexity_control3_all
Configuration: {'task': 'complexity', 'model_type': 'xgboost', 'languages':
Tracking run with wandb version 0.19.9
Run data is saved locally in /content/wandb/run-20250410_121755-a5hexyum
Syncing run xgboost\_complexity\_control3\_all to Weights & Biases \(docs\)
View project at https://wandb.ai/rokii-ku-leuven/multilingual-question-probing
View run at https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/a5hexyum
Using dataset config: control_complexity_seed3
Loaded dataset splits - Train: 7460, Validation: 441, Test: 719
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (7460, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (441, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (719, 1)
Train features: (7460, 128104), Train samples: 7460
Dev features: (441, 128104), Dev samples: 441
Test features: (719, 128104), Test samples: 719
Train set: 7460 examples, 128104 features
Validation set: 441 examples, 128104 features
Test set: 719 examples, 128104 features
Creating xgboost model for regression task
Training XGBRegressor on 7460 examples, 128104 features
[0] validation_0-rmse:0.16875 validation_0-mae:0.13535 validation_1-rms
[100] validation_0-rmse:0.15313 validation_0-mae:0.12395 validation_1-rms
[199] validation_0-rmse:0.14422 validation_0-mae:0.11714 validation_1-rms
Training completed in 165.73 seconds
Validation metrics: {'mse': 0.06355418264865875, 'rmse': 0.2520995490846002
Test metrics: {'mse': 0.04827020689845085, 'rmse': 0.2197048176496156, 'mae'
Experiment xgboost_complexity_control3_all completed successfully

```

Run history:

```

test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_mse       _
val_r2        _
val_rmse      _

```

Run summary:

```

test_mae      0.17474
test_mse      0.04827
test_r2       -0.06659

```

```

test_rmse    0.2197
training_time 165.73015
val_mae      0.20341
val_mse      0.06355
val_r2       -0.14174
val_rmse     0.2521

```

View run **xgboost_complexity_control3_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/a5hexyum>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 4 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_121755-a5hexyum/logs

Experiment xgboost_complexity_control3_all completed successfully

Running experiment 25/36: ridge_complexity_avg_links_len_all

Starting experiment: ridge_complexity_avg_links_len_all

Configuration: {'task': 'complexity', 'model_type': 'ridge', 'languages': [

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_122048-yhfx2kcp

Syncing run **ridge_complexity_avg_links_len_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/yhfx2kcp>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating ridge model for regression task

Training Ridge on 7460 examples, 128104 features

Training completed in 0.07 seconds

Validation metrics: {'mse': 0.016922442148271905, 'rmse': 0.130086287318348

Test metrics: {'mse': 0.0276182713042068, 'rmse': 0.16618745832404683, 'mae

Experiment ridge_complexity_avg_links_len_all completed successfully

Run history:

```

test_mae      —
test_mse      —
test_r2       —
test_rmse     —
training_time —
val_mae       —
val_mse       —
val_r2        —

```

```

val_r2      _
val_rmse    _

```

Run summary:

```

test_mae    0.12081
test_mse    0.02762
test_r2     0.00017
test_rmse   0.16619
training_time 0.07398
val_mae     0.08993
val_mse     0.01692
val_r2      0.35314
val_rmse    0.13009

```

View run **ridge_complexity_avg_links_len_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/yhfx2kcp>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_122048-yhfx2kcp/logs

Experiment ridge_complexity_avg_links_len_all completed successfully

Running experiment 26/36: xgboost_complexity_avg_links_len_all

Starting experiment: xgboost_complexity_avg_links_len_all

Configuration: {'task': 'complexity', 'model_type': 'xgboost', 'languages':

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_122055-5ehrs4cn

Syncing run **xgboost_complexity_avg_links_len_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/5ehrs4cn>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating xgboost model for regression task

Training XGBRegressor on 7460 examples, 128104 features

[0] validation_0-rmse:0.13260 validation_0-mae:0.09966 validation_1-rms

[100] validation_0-rmse:0.07936 validation_0-mae:0.06128 validation_1-rms

[199] validation_0-rmse:0.06909 validation_0-mae:0.05371 validation_1-rms

Training completed in 178.20 seconds

Validation metrics: {'mse': 0.014561771415174007, 'rmse': 0.120672165038893

Test metrics: {'mse': 0.03211323544383049, 'rmse': 0.17920166138691485, 'ma

Experiment xgboost_complexity_avg_links_len_all completed successfully

Run history:

```

test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_mse       _
val_r2        _
val_rmse      _

```

Run summary:

```

test_mae      0.12981
test_mse      0.03211
test_r2       -0.16255
test_rmse     0.1792
training_time 178.20335
val_mae       0.0855
val_mse       0.01456
val_r2        0.44338
val_rmse      0.12067

```

View run **xgboost_complexity_avg_links_len_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/5ehrs4cn>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 4 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_122055-5ehrs4cn/logs

Experiment xgboost_complexity_avg_links_len_all completed successfully

Running experiment 27/36: ridge_complexity_avg_max_depth_all

Starting experiment: ridge_complexity_avg_max_depth_all

Configuration: {'task': 'complexity', 'model_type': 'ridge', 'languages': [

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_122400-soyl9959

Syncing run **ridge_complexity_avg_max_depth_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/soyl9959>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

```

Test set: 719 examples, 128104 features
Creating ridge model for regression task
Training Ridge on 7460 examples, 128104 features
Training completed in 0.36 seconds
Validation metrics: {'mse': 0.0208270859508049, 'rmse': 0.14431592410681815}
Test metrics: {'mse': 0.024637873506684126, 'rmse': 0.15696456130822692, 'm
Experiment ridge_complexity_avg_max_depth_all completed successfully

```

Run history:

```

test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_mse       _
val_r2        _
val_rmse      _

```

Run summary:

```

test_mae      0.12048
test_mse      0.02464
test_r2       0.4012
test_rmse     0.15696
training_time 0.36129
val_mae       0.10535
val_mse       0.02083
val_r2        0.45032
val_rmse      0.14432

```

View run **ridge_complexity_avg_max_depth_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/soyl9959>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_122400-soyl9959/logs

Experiment ridge_complexity_avg_max_depth_all completed successfully

```

Running experiment 28/36: xgboost_complexity_avg_max_depth_all
Starting experiment: xgboost_complexity_avg_max_depth_all
Configuration: {'task': 'complexity', 'model_type': 'xgboost', 'languages':
Tracking run with wandb version 0.19.9
Run data is saved locally in /content/wandb/run-20250410_122406-ivjh10na
Syncing run xgboost_complexity_avg_max_depth_all to Weights & Biases \(docs\)
View project at https://wandb.ai/rokii-ku-leuven/multilingual-question-probing
View run at https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/ivjh10na
Using dataset config: base
Loaded dataset splits - Train: 7460, Validation: 441, Test: 719
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (7460, 1)

```

```

Loaded vectors shape: (7460, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (441, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (719, 1)
Train features: (7460, 128104), Train samples: 7460
Dev features: (441, 128104), Dev samples: 441
Test features: (719, 128104), Test samples: 719
Train set: 7460 examples, 128104 features
Validation set: 441 examples, 128104 features
Test set: 719 examples, 128104 features
Creating xgboost model for regression task
Training XGBRegressor on 7460 examples, 128104 features
[0]   validation_0-rmse:0.15736 validation_0-mae:0.11552 validation_1-rmse
[100] validation_0-rmse:0.10215 validation_0-mae:0.07899 validation_1-rmse
[199] validation_0-rmse:0.09007 validation_0-mae:0.07011 validation_1-rmse
Training completed in 172.60 seconds
Validation metrics: {'mse': 0.018239542841911316, 'rmse': 0.135053851636713}
Test metrics: {'mse': 0.024972688406705856, 'rmse': 0.15802749256602744, 'm
Experiment xgboost_complexity_avg_max_depth_all completed successfully

```

Run history:

```

test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_mse       _
val_r2        _
val_rmse      _

```

Run summary:

```

test_mae      0.11626
test_mse      0.02497
test_r2       0.39306
test_rmse     0.15803
training_time 172.59872
val_mae       0.10044
val_mse       0.01824
val_r2        0.51861
val_rmse      0.13505

```

View run **xgboost_complexity_avg_max_depth_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/ivjh10na>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 4 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_122406-ivjh10na/logs

Experiment xgboost_complexity_avg_max_depth_all completed successfully


```
Running experiment 29/36: ridge_complexity_avg_subordinate_chain_len_all
Starting experiment: ridge_complexity_avg_subordinate_chain_len_all
Configuration: {'task': 'complexity', 'model_type': 'ridge', 'languages': []}
Tracking run with wandb version 0.19.9
Run data is saved locally in /content/wandb/run-20250410_122705-3jmh0yfa
Syncing run ridge\_complexity\_avg\_subordinate\_chain\_len\_all to Weights & Biases (docs)
View project at https://wandb.ai/rokii-ku-leuven/multilingual-question-probing
View run at https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/3jmh0yfa
Using dataset config: base
Loaded dataset splits - Train: 7460, Validation: 441, Test: 719
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (7460, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (441, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (719, 1)
Train features: (7460, 128104), Train samples: 7460
Dev features: (441, 128104), Dev samples: 441
Test features: (719, 128104), Test samples: 719
Train set: 7460 examples, 128104 features
Validation set: 441 examples, 128104 features
Test set: 719 examples, 128104 features
Creating ridge model for regression task
Training Ridge on 7460 examples, 128104 features
Training completed in 0.07 seconds
Validation metrics: {'mse': 0.03464916128849565, 'rmse': 0.1861428518329287}
Test metrics: {'mse': 0.03939764253899441, 'rmse': 0.19848839396547702, 'mae': 0.14603}
Experiment ridge_complexity_avg_subordinate_chain_len_all completed success
```

Run history:

test_mae	—
test_mse	—
test_r2	—
test_rmse	—
training_time	—
val_mae	—
val_mse	—
val_r2	—
val_rmse	—

Run summary:

test_mae	0.14603
test_mse	0.0394
test_r2	0.24465
test_rmse	0.19849
training_time	0.06869
val_mae	0.12158
val_mse	0.03465


```
val_r2      0.39023
val_rmse    0.18614
```

View run **ridge_complexity_avg_subordinate_chain_len_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/3jmh0yfa>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_122705-3jmh0yfa/logs

Experiment ridge_complexity_avg_subordinate_chain_len_all completed success

Running experiment 30/36: xgboost_complexity_avg_subordinate_chain_len_all

Starting experiment: xgboost_complexity_avg_subordinate_chain_len_all

Configuration: {'task': 'complexity', 'model_type': 'xgboost', 'languages':

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_122710-2iscgvfz

Syncing run **xgboost_complexity_avg_subordinate_chain_len_all** to [Weights & Biases](#) ([docs](#))

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/2iscgvfz>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating xgboost model for regression task

Training XGBRegressor on 7460 examples, 128104 features

[0] validation_0-rmse:0.18204 validation_0-mae:0.12738 validation_1-rmse

[100] validation_0-rmse:0.10281 validation_0-mae:0.05747 validation_1-rmse

[199] validation_0-rmse:0.08558 validation_0-mae:0.04487 validation_1-rmse

Training completed in 160.85 seconds

Validation metrics: {'mse': 0.03562215715646744, 'rmse': 0.1887383298550335

Test metrics: {'mse': 0.04516085982322693, 'rmse': 0.21251084636607828, 'ma

Experiment xgboost_complexity_avg_subordinate_chain_len_all completed succe

Run history:

```
test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_mse       _
val_r2        _
val_rmse      _
```

Run summary:

```

test_mae    0.14282
test_mse    0.04516
test_r2     0.13415
test_rmse   0.21251
training_time 160.85308
val_mae     0.10981
val_mse     0.03562
val_r2      0.37311
val_rmse    0.18874

```

View run **xgboost_complexity_avg_subordinate_chain_len_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/2iscgvfz>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 4 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_122710-2iscgvfz/logs

Experiment xgboost_complexity_avg_subordinate_chain_len_all completed succe

Running experiment 31/36: ridge_complexity_avg_verb_edges_all

Starting experiment: ridge_complexity_avg_verb_edges_all

Configuration: {'task': 'complexity', 'model_type': 'ridge', 'languages': [

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_122956-ps3beifp

Syncing run **ridge_complexity_avg_verb_edges_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/ps3beifp>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating ridge model for regression task

Training Ridge on 7460 examples, 128104 features

Training completed in 0.49 seconds

Validation metrics: {'mse': 0.032790483424747746, 'rmse': 0.181081427608542

Test metrics: {'mse': 0.0616369866818339, 'rmse': 0.24826797353229815, 'mae

Experiment ridge_complexity_avg_verb_edges_all completed successfully

Run history:

```

test_mae    _
test_mse    _

```

```

test_r2      _
test_rmse    _
training_time _
val_mae      _
val_mse      _
val_r2       _
val_rmse     _

```

Run summary:

```

test_mae      0.19534
test_rmse     0.06164
test_r2       0.02481
test_rmse     0.24827
training_time 0.48552
val_mae       0.14061
val_mse       0.03279
val_r2        0.40289
val_rmse      0.18108

```

View run **ridge_complexity_avg_verb_edges_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/ps3beifp>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_122956-ps3beifp/logs

Experiment ridge_complexity_avg_verb_edges_all completed successfully

Running experiment 32/36: xgboost_complexity_avg_verb_edges_all

Starting experiment: xgboost_complexity_avg_verb_edges_all

Configuration: {'task': 'complexity', 'model_type': 'xgboost', 'languages':

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_123002-yt3l2930

Syncing run **xgboost_complexity_avg_verb_edges_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/yt3l2930>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating xgboost model for regression task

Training XGBRegressor on 7460 examples, 128104 features

[0] validation_0-rmse:0.24259 validation_0-mae:0.20465 validation_1-rms

[100] validation_0-rmse:0.15104 validation_0-mae:0.12046 validation_1-rms

[1001] validation_0-rmse:0.13000 validation_0-mae:0.10245 validation_1-rms

```
[199] validation_0-rmse:0.13090 validation_0-mae:0.10245 validation_1-rmse
Training completed in 176.79 seconds
Validation metrics: {'mse': 0.036020196974277496, 'rmse': 0.189789875847679
Test metrics: {'mse': 0.06620804220438004, 'rmse': 0.2573092345882286, 'mae'
Experiment xgboost_complexity_avg_verb_edges_all completed successfully
```

Run history:

```
test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_mse       _
val_r2        _
val_rmse      _
```

Run summary:

```
test_mae      0.2066
test_mse      0.06621
test_r2       -0.04751
test_rmse     0.25731
training_time 176.79071
val_mae       0.14475
val_mse       0.03602
val_r2        0.34407
val_rmse      0.18979
```

View run **xgboost_complexity_avg_verb_edges_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/yt3l2930>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 4 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_123002-yt3l2930/logs

Experiment xgboost_complexity_avg_verb_edges_all completed successfully

Running experiment 33/36: ridge_complexity_lexical_density_all

Starting experiment: ridge_complexity_lexical_density_all

Configuration: {'task': 'complexity', 'model_type': 'ridge', 'languages': [

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_123305-4ym2h8wg

Syncing run **ridge_complexity_lexical_density_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/4ym2h8wg>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

```

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (719, 1)
Train features: (7460, 128104), Train samples: 7460
Dev features: (441, 128104), Dev samples: 441
Test features: (719, 128104), Test samples: 719
Train set: 7460 examples, 128104 features
Validation set: 441 examples, 128104 features
Test set: 719 examples, 128104 features
Creating ridge model for regression task
Training Ridge on 7460 examples, 128104 features
Training completed in 0.08 seconds
Validation metrics: {'mse': 0.024298308096099074, 'rmse': 0.155879145802442}
Test metrics: {'mse': 0.05117392084392208, 'rmse': 0.22621653530173713, 'mae': 0.17861}
Experiment ridge_complexity_lexical_density_all completed successfully

```

Run history:

```

test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_mse       _
val_r2        _
val_rmse      _

```

Run summary:

```

test_mae      0.17861
test_mse      0.05117
test_r2       -0.16016
test_rmse     0.22622
training_time 0.084
val_mae       0.11629
val_mse       0.0243
val_r2        0.51889
val_rmse      0.15588

```

View run **ridge_complexity_lexical_density_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/4ym2h8wg>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_123305-4ym2h8wg/logs

Experiment ridge_complexity_lexical_density_all completed successfully

Running experiment 34/36: xgboost_complexity_lexical_density_all

Starting experiment: xgboost_complexity_lexical_density_all

Configuration: {'task': 'complexity', 'model_type': 'xgboost', 'languages': 'all'}

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_123310-13030k7u

```

Syncing run xgboost\_complexity\_lexical\_density\_all to Weights & Biases \(docs\)
View project at https://wandb.ai/rokii-ku-leuven/multilingual-question-probing
View run at https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/13030k7u
Using dataset config: base
Loaded dataset splits - Train: 7460, Validation: 441, Test: 719
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (7460, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (441, 1)
Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data
Loaded vectors shape: (719, 1)
Train features: (7460, 128104), Train samples: 7460
Dev features: (441, 128104), Dev samples: 441
Test features: (719, 128104), Test samples: 719
Train set: 7460 examples, 128104 features
Validation set: 441 examples, 128104 features
Test set: 719 examples, 128104 features
Creating xgboost model for regression task
Training XGBRegressor on 7460 examples, 128104 features
[0]   validation_0-rmse:0.21781 validation_0-mae:0.17644 validation_1-rmse
[100] validation_0-rmse:0.12204 validation_0-mae:0.09491 validation_1-rmse
[199] validation_0-rmse:0.10340 validation_0-mae:0.08027 validation_1-rmse
Training completed in 178.85 seconds
Validation metrics: {'mse': 0.0270399022847414, 'rmse': 0.16443814121043027}
Test metrics: {'mse': 0.05391478165984154, 'rmse': 0.232195567700681, 'mae': 0.18157}
Experiment xgboost_complexity_lexical_density_all completed successfully

```

Run history:

```

test_mae      _
test_mse      _
test_r2       _
test_rmse     _
training_time _
val_mae       _
val_mse       _
val_r2        _
val_rmse      _

```

Run summary:

```

test_mae      0.18157
test_mse      0.05391
test_r2       -0.2223
test_rmse     0.2322
training_time 178.84868
val_mae       0.12171
val_mse       0.02704
val_r2        0.46461
val_rmse      0.16444

```

View run **xgboost_complexity_lexical_density_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/13030k7u>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 4 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_123310-13030k7u/logs

Experiment **xgboost_complexity_lexical_density_all** completed successfully

Running experiment 35/36: **ridge_complexity_n_tokens_all**

Starting experiment: **ridge_complexity_n_tokens_all**

Configuration: {'task': 'complexity', 'model_type': 'ridge', 'languages': ['

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_123614-rk4i7a19

Syncing run **ridge_complexity_n_tokens_all** to [Weights & Biases](#) ([docs](#))

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/rk4i7a19>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating ridge model for regression task

Training Ridge on 7460 examples, 128104 features

Training completed in 0.50 seconds

Validation metrics: {'mse': 0.017267449851224238, 'rmse': 0.131405669022398

Test metrics: {'mse': 0.027103662020791692, 'rmse': 0.16463189855186539, 'm

Experiment **ridge_complexity_n_tokens_all** completed successfully

Run history:

test_mae	—
test_mse	—
test_r2	—
test_rmse	—
training_time	—
val_mae	—
val_mse	—
val_r2	—
val_rmse	—

Run summary:

test_mae	0.12934
test mse	0.0271


```

test_r2      0.30697
test_rmse    0.16463
training_time 0.49677
val_mae      0.09301
val_mse      0.01727
val_r2       0.48545
val_rmse     0.13141

```

View run **ridge_complexity_n_tokens_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/rk4i7a19>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 2 media file(s), 0 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_123614-rk4i7a19/logs

Experiment ridge_complexity_n_tokens_all completed successfully

Running experiment 36/36: xgboost_complexity_n_tokens_all

Starting experiment: xgboost_complexity_n_tokens_all

Configuration: {'task': 'complexity', 'model_type': 'xgboost', 'languages':

Tracking run with wandb version 0.19.9

Run data is saved locally in /content/wandb/run-20250410_123620-zkxc7jw6

Syncing run **xgboost_complexity_n_tokens_all** to [Weights & Biases \(docs\)](#)

View project at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

View run at <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/zkxc7jw6>

Using dataset config: base

Loaded dataset splits - Train: 7460, Validation: 441, Test: 719

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (7460, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (441, 1)

Loading features from /content/drive/MyDrive/ColabNotebooks/qtype-eval/data

Loaded vectors shape: (719, 1)

Train features: (7460, 128104), Train samples: 7460

Dev features: (441, 128104), Dev samples: 441

Test features: (719, 128104), Test samples: 719

Train set: 7460 examples, 128104 features

Validation set: 441 examples, 128104 features

Test set: 719 examples, 128104 features

Creating xgboost model for regression task

Training XGBRegressor on 7460 examples, 128104 features

[0] validation_0-rmse:0.15040 validation_0-mae:0.11855 validation_1-rms

[100] validation_0-rmse:0.06672 validation_0-mae:0.05138 validation_1-rms

[199] validation_0-rmse:0.05470 validation_0-mae:0.04233 validation_1-rms

Training completed in 179.92 seconds

Validation metrics: {'mse': 0.015652107074856758, 'rmse': 0.125108381313390

Test metrics: {'mse': 0.02401013672351837, 'rmse': 0.15495204652897737, 'ma

Experiment xgboost_complexity_n_tokens_all completed successfully

Run history:

```

test_mae      —
test_rmse     —
test_r2       —
test_rmse     —

```



```

training_time _
val_mae _
val_mse _
val_r2 _
val_rmse _

```

Run summary:

```

test_mae    0.11202
test_mse    0.02401
test_r2     0.38607
test_rmse   0.15495
training_time 179.91634
val_mae     0.08619
val_mse     0.01565
val_r2      0.53359
val_rmse    0.12511

```

View run **xgboost_complexity_n_tokens_all** at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing/runs/zkxc7jw6>

View project at: <https://wandb.ai/rokii-ku-leuven/multilingual-question-probing>

Synced 5 W&B file(s), 4 media file(s), 2 artifact file(s) and 0 other file(s)

Find logs at: ./wandb/run-20250410_123620-zkxc7jw6/logs

Experiment xgboost_complexity_n_tokens_all completed successfully

All all experiments completed!

```

{'dummy_question_type_all': {'model_type': 'DummyClassifier',
  'training_time': 0.0011227130889892578,
  'val_metrics': {'accuracy': 0.5056689342403629, 'f1': 0.0},
  'test_metrics': {'accuracy': 0.5104311543810849, 'f1': 0.0},
  'task': 'question_type',
  'languages': ['all'],
  'control_index': None,
  'submetric': None},
'dummy_question_type_control1_all': {'model_type': 'DummyClassifier',
  'training_time': 0.0008437633514404297,
  'val_metrics': {'accuracy': 0.5056689342403629, 'f1': 0.0},
  'test_metrics': {'accuracy': 0.5104311543810849, 'f1': 0.0},
  'task': 'question_type',
  'languages': ['all'],
  'control_index': 1,
  'submetric': None},
'dummy_question_type_control2_all': {'model_type': 'DummyClassifier',
  'training_time': 0.0009539127349853516,
  'val_metrics': {'accuracy': 0.5056689342403629, 'f1': 0.0},
  'test_metrics': {'accuracy': 0.5104311543810849, 'f1': 0.0},
  'task': 'question_type',
  'languages': ['all'],
  'control_index': 2,
  'submetric': None},
'dummy_question_type_control3_all': {'model_type': 'DummyClassifier',
  'training_time': 0.0009281635284423828,
  'val_metrics': {'accuracy': 0.5056689342403629, 'f1': 0.0},
  'test_metrics': {'accuracy': 0.5104311543810849, 'f1': 0.0},

```

```
'task': 'question_type',
'languages': ['all'],
'control_index': 3,
'submetric': None},
'logistic_question_type_all': {'model_type': 'LogisticRegression',
'training_time': 0.09628891944885254,
'val_metrics': {'accuracy': 0.8662131519274376, 'f1':
0.8631090487238979},
'test_metrics': {'accuracy': 0.7510431154381085, 'f1':
0.7672301690507152},
'task': 'question_type',
'languages': ['all'],
'control_index': None,
'submetric': None},
'logistic_question_type_control1_all': {'model_type':
'LogisticRegression',
'training_time': 0.6301925182342529,
'val_metrics': {'accuracy': 0.5147392290249433, 'f1':
0.4623115577889447},
'test_metrics': {'accuracy': 0.48400556328233657, 'f1':
0.4370257966616085},
'task': 'question_type',
'languages': ['all'],
'control_index': 1,
'submetric': None},
'logistic_question_type_control2_all': {'model_type':
'LogisticRegression',
'training_time': 0.10853815078735352,
'val_metrics': {'accuracy': 0.49206349206349204, 'f1':
0.46411483253588515},
'test_metrics': {'accuracy': 0.46870653685674546, 'f1':
0.4415204678362573},
'task': 'question_type',
'languages': ['all'],
'control_index': 2,
'submetric': None},
'logistic_question_type_control3_all': {'model_type':
'LogisticRegression',
'training_time': 0.12429952621459961,
'val_metrics': {'accuracy': 0.5306122448979592, 'f1':
0.5059665871121718},
'test_metrics': {'accuracy': 0.5187760778859527, 'f1':
0.5167597765363129},
'task': 'question_type',
'languages': ['all'],
'control_index': 3,
'submetric': None},
'xgboost_question_type_all': {'model_type': 'XGBClassifier',
'training_time': 151.00834012031555,
'val_metrics': {'accuracy': 0.8616780045351474, 'f1':
0.8564705882352941},
'test_metrics': {'accuracy': 0.7649513212795549, 'f1':
0.7623066104078763},
'task': 'question_type',
'languages': ['all'],
'control_index': None,
'submetric': None},
'xgboost_question_type_control1_all': {'model_type': 'XGBClassifier',
```

```
'training_time': 151.28243374824524,
'val_metrics': {'accuracy': 0.5215419501133787, 'f1':
0.4490861618798956},
'test_metrics': {'accuracy': 0.502086230876217, 'f1':
0.39730639730639733},
'task': 'question_type',
'languages': ['all'],
'control_index': 1,
'submetric': None},
'xgboost_question_type_control2_all': {'model_type': 'XGBClassifier',
'training_time': 150.1831295490265,
'val_metrics': {'accuracy': 0.5419501133786848, 'f1':
0.5280373831775701},
'test_metrics': {'accuracy': 0.5006954102920723, 'f1':
0.5102319236016372},
'task': 'question_type',
'languages': ['all'],
'control_index': 2,
'submetric': None},
'xgboost_question_type_control3_all': {'model_type': 'XGBClassifier',
'training_time': 150.2112536430359,
'val_metrics': {'accuracy': 0.5578231292517006, 'f1':
0.5368171021377672},
'test_metrics': {'accuracy': 0.5354659248956884, 'f1':
0.49393939393939396},
'task': 'question_type',
'languages': ['all'],
'control_index': 3,
'submetric': None},
'dummy_complexity_all': {'model_type': 'DummyRegressor',
'training_time': 0.0008461475372314453,
'val_metrics': {'mse': 0.05940214544534683,
'rmse': 0.2437255535337787,
'mae': 0.19818523526191711,
'r2': -0.06714892387390137},
'test_metrics': {'mse': 0.04718531668186188,
'rmse': 0.21722181447051278,
'mae': 0.17157766222953796,
'r2': -0.042618393898010254},
'task': 'complexity',
'languages': ['all'],
'control_index': None,
'submetric': None},
'dummy_complexity_control1_all': {'model_type': 'DummyRegressor',
'training_time': 0.0007047653198242188,
'val_metrics': {'mse': 0.05940214544534683,
'rmse': 0.2437255535337787,
'mae': 0.19818523526191711,
'r2': -0.06714892387390137},
'test_metrics': {'mse': 0.04718531668186188,
'rmse': 0.21722181447051278,
'mae': 0.17157766222953796,
'r2': -0.042618393898010254},
'task': 'complexity',
'languages': ['all'],
'control_index': 1,
'submetric': None},
'dummy_complexity_control2_all': {'model_type': 'DummyRegressor',
```

```
'training_time': 0.0007073879241943359,
'val_metrics': {'mse': 0.05940214544534683,
  'rmse': 0.2437255535337787,
  'mae': 0.19818523526191711,
  'r2': -0.06714892387390137},
'test_metrics': {'mse': 0.04718531668186188,
  'rmse': 0.21722181447051278,
  'mae': 0.17157766222953796,
  'r2': -0.042618393898010254},
'task': 'complexity',
'languages': ['all'],
'control_index': 2,
'submetric': None,
'dummy_complexity_control3_all': {'model_type': 'DummyRegressor',
  'training_time': 0.0008459091186523438,
  'val_metrics': {'mse': 0.05940214544534683,
    'rmse': 0.2437255535337787,
    'mae': 0.19818523526191711,
    'r2': -0.06714892387390137},
  'test_metrics': {'mse': 0.04718531668186188,
    'rmse': 0.21722181447051278,
    'mae': 0.17157766222953796,
    'r2': -0.042618393898010254},
  'task': 'complexity',
  'languages': ['all'],
  'control_index': 3,
  'submetric': None},
'ridge_complexity_all': {'model_type': 'Ridge',
  'training_time': 0.11220335960388184,
  'val_metrics': {'mse': 0.02834229564475883,
    'rmse': 0.1683517022330301,
    'mae': 0.13133771533684654,
    'r2': 0.49083574300101995},
  'test_metrics': {'mse': 0.02617858559723656,
    'rmse': 0.16179797772913157,
    'mae': 0.12907747825580945,
    'r2': 0.4215514664697695},
  'task': 'complexity',
  'languages': ['all'],
  'control_index': None,
  'submetric': None},
'ridge_complexity_control1_all': {'model_type': 'Ridge',
  'training_time': 0.15994715690612793,
  'val_metrics': {'mse': 0.06315823313570658,
    'rmse': 0.25131301823762847,
    'mae': 0.20248373167927824,
    'r2': -0.13462632847304934},
  'test_metrics': {'mse': 0.04897569807893945,
    'rmse': 0.22130453695968244,
    'mae': 0.17379239627707305,
    'r2': -0.08217919670084828},
  'task': 'complexity',
  'languages': ['all'],
  'control_index': 1,
  'submetric': None},
'ridge_complexity_control2_all': {'model_type': 'Ridge',
  'training_time': 0.06925630569458008,
  'val_metrics': {'mse': 0.06119368838813863,
```

```
'rmse': 0.2473735806187448,  
'mae': 0.20174630291450335,  
'r2': -0.09933363449814658},  
'test_metrics': {'mse': 0.04839423261099806,  
'rmse': 0.2199868919072181,  
'mae': 0.1760088210857055,  
'r2': -0.06933099120938446},  
'task': 'complexity',  
'languages': ['all'],  
'control_index': 2,  
'submetric': None},  
'ridge_complexity_control3_all': {'model_type': 'Ridge',  
'training_time': 0.08811163902282715,  
'val_metrics': {'mse': 0.06666476686615211,  
'rmse': 0.258195210773074,  
'mae': 0.2097365699093491,  
'r2': -0.1976205779114968},  
'test_metrics': {'mse': 0.05271930345372578,  
'rmse': 0.229606845398228,  
'mae': 0.18119310161090776,  
'r2': -0.1648988314617741},  
'task': 'complexity',  
'languages': ['all'],  
'control_index': 3,  
'submetric': None},  
'xgboost_complexity_all': {'model_type': 'XGBRegressor',  
'training_time': 176.19616532325745,  
'val_metrics': {'mse': 0.025757428258657455,  
'rmse': 0.16049120928779076,  
'mae': 0.12436210364103317,  
'r2': 0.5372724533081055},  
'test_metrics': {'mse': 0.022604431957006454,  
'rmse': 0.15034770353087024,  
'mae': 0.11820846796035767,  
'r2': 0.5005269050598145},  
'task': 'complexity',  
'languages': ['all'],  
'control_index': None,  
'submetric': None},  
'xgboost_complexity_control1_all': {'model_type': 'XGBRegressor',  
'training_time': 167.21830010414124,  
'val_metrics': {'mse': 0.05997728556394577,  
'rmse': 0.24490260424084054,  
'mae': 0.19969910383224487,  
'r2': -0.07748115062713623},  
'test_metrics': {'mse': 0.04638424143195152,  
'rmse': 0.21537001052131544,  
'mae': 0.1706438511610031,  
'r2': -0.0249176025390625},  
'task': 'complexity',  
'languages': ['all'],  
'control_index': 1,  
'submetric': None},  
'xgboost_complexity_control2_all': {'model_type': 'XGBRegressor',  
'training_time': 163.41401720046997,  
'val_metrics': {'mse': 0.06018916890025139,  
'rmse': 0.2453348098013231,  
'mae': 0.1994825154542923,
```

```
'r2': -0.08128750324249268},
'test_metrics': {'mse': 0.04748661443591118,
'rmse': 0.21791423642321117,
'mae': 0.1735936552286148,
'r2': -0.049275994300842285},
'task': 'complexity',
'languages': ['all'],
'control_index': 2,
'submetric': None},
'xgboost_complexity_control3_all': {'model_type': 'XGBRegressor',
'training_time': 165.7301480770111,
'val_metrics': {'mse': 0.06355418264865875,
'rmse': 0.2520995490846002,
'mae': 0.20340751111507416,
'r2': -0.1417393684387207},
'test_metrics': {'mse': 0.04827020689845085,
'rmse': 0.2197048176496156,
'mae': 0.17473910748958588,
'r2': -0.06659042835235596},
'task': 'complexity',
'languages': ['all'],
'control_index': 3,
'submetric': None},
'ridge_complexity_avg_links_len_all': {'model_type': 'Ridge',
'training_time': 0.07397699356079102,
'val_metrics': {'mse': 0.016922442148271905,
'rmse': 0.13008628731834845,
'mae': 0.08992663945783835,
'r2': 0.35314230788828116},
'test_metrics': {'mse': 0.0276182713042068,
'rmse': 0.16618745832404683,
'mae': 0.12081335252288831,
'r2': 0.00017095303973324594},
'task': 'complexity',
'languages': ['all'],
'control_index': None,
'submetric': 'avg_links_len'},
'xgboost_complexity_avg_links_len_all': {'model_type': 'XGBRegressor',
'training_time': 178.20334911346436,
'val_metrics': {'mse': 0.014561771415174007,
'rmse': 0.12067216503889373,
'mae': 0.08550466597080231,
'r2': 0.4433785676956177},
'test_metrics': {'mse': 0.03211323544383049,
'rmse': 0.17920166138691485,
'mae': 0.1298091858625412,
'r2': -0.16255462169647217},
'task': 'complexity',
'languages': ['all'],
'control_index': None,
'submetric': 'avg_links_len'},
'ridge_complexity_avg_max_depth_all': {'model_type': 'Ridge',
'training_time': 0.36128902435302734,
'val_metrics': {'mse': 0.0208270859508049,
'rmse': 0.14431592410681815,
'mae': 0.10534839194017084,
'r2': 0.4503211045410648},
'test_metrics': {'mse': 0.024637873506684126,
```

```
'rmse': 0.15696456130822692,
'mae': 0.12048048104062958,
'r2': 0.4011965413934825},
'task': 'complexity',
'languages': ['all'],
'control_index': None,
'submetric': 'avg_max_depth'},
'xgboost_complexity_avg_max_depth_all': {'model_type': 'XGBRegressor',
'training_time': 172.59871649742126,
'val_metrics': {'mse': 0.018239542841911316,
'rmse': 0.1350538516367131,
'mae': 0.10044150054454803,
'r2': 0.5186128616333008},
'test_metrics': {'mse': 0.024972688406705856,
'rmse': 0.15802749256602744,
'mae': 0.1162642389535904,
'r2': 0.3930591940879822},
'task': 'complexity',
'languages': ['all'],
'control_index': None,
'submetric': 'avg_max_depth'},
'ridge_complexity_avg_subordinate_chain_len_all': {'model_type': 'Ridge',
'training_time': 0.06868767738342285,
'val_metrics': {'mse': 0.03464916128849565,
'rmse': 0.1861428518329287,
'mae': 0.12158427536186517,
'r2': 0.39023020721194224},
'test_metrics': {'mse': 0.03939764253899441,
'rmse': 0.19848839396547702,
'mae': 0.14602712267078857,
'r2': 0.24464822964504707},
'task': 'complexity',
'languages': ['all'],
'control_index': None,
'submetric': 'avg_subordinate_chain_len'},
'xgboost_complexity_avg_subordinate_chain_len_all': {'model_type':
'XGBRegressor',
'training_time': 160.85308480262756,
'val_metrics': {'mse': 0.03562215715646744,
'rmse': 0.18873832985503353,
'mae': 0.10980933904647827,
'r2': 0.37310701608657837},
'test_metrics': {'mse': 0.04516085982322693,
'rmse': 0.21251084636607828,
'mae': 0.1428225040435791,
'r2': 0.13415294885635376},
'task': 'complexity',
```

Start coding or [generate](#) with AI.

