Contents

1 Data Loaders Removal - Complete Refactoring

1.1 Summary

The `data_loaders.py` file has been completely removed. All data loading functionality has been moved into the extraction mode class hierarchy, and `classifier_v2.py` now uses extraction mode objects directly.

1.2 Changes Made

1.2.1 1. Removed File

- **Deleted**: `skol_classifier/data_loaders.py` (178 lines)

1.2.2 2. Modified Files

1.2.2.1 classifier_v2.py Initialization Changes (line 245-246):

```
# Before:
self.extraction_mode = extraction_mode  # String: 'line', 'paragraph', 'section'

# After:
from .extraction_modes import get_mode
self.extraction_mode = get_mode(extraction_mode)  # ExtractionMode object
```

Loading Methods Updated:

1. **_load_annotated_from_files()** (lines 878-884):

```
# Before:
from .data_loaders import AnnotatedTextLoader
loader = AnnotatedTextLoader(self.spark)
return loader.load_from_files(
    self.file_paths,
    line_level=self.line_level,
    collapse_labels=self.collapse_labels
)

# After:
return self.extraction_mode.load_annotated_from_files(
    spark=self.spark,
    file_paths=self.file_paths,
    collapse_labels=self.collapse_labels
)
```

2. **_load_raw_from_files()** (lines 772-780):

```
# Before:
if self.extraction_mode == 'section':
    return self._load_sections_from_files()
from .preprocessing import RawTextLoader
loader = RawTextLoader(self.spark)
df = loader.load_files(self.file_paths, line_level=self.line_level)
return self.load_raw_from_df(df)

# After:
df = self.extraction_mode.load_raw_from_files(
    spark=self.spark,
    file_paths=self.file_paths
)
return self.load_raw_from_df(df)
```

3. **_load_raw_from_couchdb()** (lines 782-799):

```
# Before:
if self.extraction_mode == 'section':
    return self._load_sections_from_couchdb()
conn = CouchDBConnection(...)
df = conn.load_distributed(self.spark, self.couchdb_pattern)
return self.load_raw_from_df(df)

# After:
if self.extraction_mode.name == 'section':
    return self._load_sections_from_couchdb()
df = self.extraction_mode.load_raw_from_couchdb(
    spark=self.spark,
    couchdb_url=self.couchdb_url,
    database=self.couchdb_database,
    username=self.couchdb_username,
    password=self.couchdb_password,
    pattern=self.couchdb_pattern
)
return self.load_raw_from_df(df)
```

String Conversion for Serialization (lines 926, 1313, 1471):

```
# When passing to functions expecting string:
extraction_mode=self.extraction_mode.name

# When saving to JSON:
'tokenizer': self.extraction_mode.name
```

1.3 Backwards Compatibility

1.3.1 String Comparisons Still Work

Thanks to __eq__ override in ExtractionMode:

```
# This still works:
if self.extraction_mode == 'section':
    ...

# Because ExtractionMode.__eq__ handles string comparison
```

1.3.2 External API Impact

No impact on external API. Code using TaxaClassifier continues to work:

```
# Still works - gets converted to ExtractionMode object internally
classifier = TaxaClassifier(
```

```
    spark=spark,
    extraction_mode='section',  # String is converted to object
    ...
)
```

1.4 Benefits

1.4.1 1. Cleaner Architecture

- Data loading logic is now in extraction mode classes where it belongs
- No intermediate wrapper classes needed
- Direct method calls instead of delegation through wrappers

1.4.2 2. Fewer Files to Maintain

- Removed 178 lines of wrapper code
- One less file in the module
- All related functionality in `extraction_modes/` directory

1.4.3 3. Better Type Safety

- `self.extraction_mode` is now strongly typed as `ExtractionMode`
- IDE autocomplete shows available methods
- Easier to find all uses of extraction mode features

1.4.4 4. Simplified Call Chain

```
Before:
classifier._load_raw_from_files()
  → creates RawTextLoader
    → RawTextLoader.load_from_files()
      → calls get_mode()
        → mode.load_raw_from_files()

After:
classifier._load_raw_from_files()
  → self.extraction_mode.load_raw_from_files()
```

1.5 Migration Impact

1.5.1 No Breaking Changes

All existing code continues to work because: 1. `TaxaClassifier.__init__()` still accepts string for `extraction_mode` 2. String comparisons with

`self.extraction_mode` still work 3. `.name` property provides string when needed

1.5.2 Files That Import data_loaders

None found! The grep search confirmed no files import `data_loaders`: - No `from .data_loaders import ...` - No `import data_loaders` - Safe to delete

1.6 Testing Verification

After removal, verify:

```
# No import errors
python -c "from skol_classifier.classifier_v2 import TaxaClassifier"

# Extraction modes work
python -c "from skol_classifier.extraction_modes import get_mode; print(get_mode

# Classifier still works
python -c "from skol_classifier import TaxaClassifier; c = TaxaClassifier(extrac
```

1.7 Code Statistics

1.7.1 Lines Removed

- `data_loaders.py`: 178 lines deleted
- `classifier_v2.py`: ~20 lines simplified (net reduction)
- **Total reduction**: ~200 lines

1.7.2 Lines Added

- `extraction_modes/line.py`: 241 lines (previously in data_loaders.py)
- `extraction_modes/paragraph.py`: 204 lines (previously in data_loaders.py)
- `extraction_modes/section.py`: 148 lines (new, replaces classifier logic)
- `classifier_v2.py`: ~5 lines for mode conversion
- **Total addition**: ~600 lines (better organized)

1.7.3 Net Impact

- Code is better organized in separate files by mode
- No duplication between modes
- Each mode is self-contained and testable

1.8 Future Improvements

With `data_loaders.py` removed, future work can focus on:

1. **Remove section mode special case**: Move `_load_sections_from_couchdb()` logic into `SectionExtractionMode`
2. **Simplify load_raw_from_df()**: This method still has `if self.line_level` checks that could be moved to modes
3. **Direct mode usage**: Update any remaining string-based `extraction_mode` checks to use object methods
4. **Deprecate string comparisons**: Eventually require explicit `.name` when comparing to strings

1.9 Documentation Updates

Updated documents: - DATA_LOADERS_REFACTORING.md - Original refactoring plan - EXTRACTION_MODE_HIERARCHY.md - Object hierarchy overview - This document - Removal completion notes