Product Classification Challenge Report

Theofanopoulos Michail - *p3352401*
Kitsakis Georgios - *p3352406*

1. Problem Overview

This project tackles a multi-modal product classification challenge using Amazon sports product data. The goal is to classify 276,453 products into 16 sport categories using three complementary data modalities: text data (product descriptions), graph data (co-viewing relationships with 1.8M edges), and price data. The evaluation metric is multi-class logarithmic loss, which heavily penalizes overconfident incorrect predictions.

Key Dataset Characteristics:

- Highly imbalanced classes (ratio: 38.32, ranging from 1,129 to 43,260 samples)
- Sparse graph connectivity (density: 0.000047)
- Missing price data (28% of products)
- Variable text length (1-4,304 words per description)

2. Methodology

2.1 Feature Engineering

Text Features: We implemented multiple approaches with systematic feature selection:

(1) **Enhanced TF-IDF**: Multi-vectorizer strategy combining word-level (15K), character-level (8K), and brand-level (3K) features totaling 26K dimensions. Applied **SelectKBest with chi-squared test (k=15,000)** to identify most discriminative features for sparse TF-IDF data.

(2) **Word2Vec embeddings**: 100-dimensional vectors with skip-gram architecture and custom tokenization including lemmatization, stemming, and stopword removal.

(3) **Combined text features**: Merged TF-IDF (15K) + Word2Vec (100) = 15.1K features, then applied **SelectKBest with mutual information (k=20,000)** to handle mixed sparse and dense feature types.

(4) **Neural network preprocessing**: Used **TruncatedSVD (n_components=500)** to reduce high-dimensional TF-IDF for efficient MLP training.

Graph Features:

(1) **Node2Vec embeddings** (64-dimensional) learned through random walks simulating user browsing patterns (walk_length=10, num_walks=15, p=0.5, q=2.0), and

(2) **Ten hand-crafted structural features** including degree, PageRank, betweenness centrality, and clustering coefficient. Combined approach stacked both representations (74 dimensions total) with **StandardScaler normalization**.

Price Features:

Twelve engineered features including price transformations (raw, log, square root), text-price interactions (price per word, price × description length), statistical categories (top/bottom 10th percentiles), and behavioral pattern indicators. Applied **selective standardization**: continuous features (first 8 dimensions) scaled with StandardScaler while preserving binary indicators (last 4 dimensions) unchanged.

Why Feature Selection Was Critical:

- **Chi-squared for TF-IDF**: Optimal for sparse, non-negative features, removing noise while preserving discriminative terms
- **Mutual information for mixed features**: Handles both positive TF-IDF and potentially negative Word2Vec features effectively
- **Two-stage selection strategy**: Reduced computational complexity before expensive feature combination operations
- **Dimensionality management**: Prevented curse of dimensionality while maintaining information quality

2.2 Model Selection and Ensemble Strategy

We evaluated multiple algorithms per modality, focusing on probability calibration due to the log-loss metric. Our meta-learning ensemble used stacking: (1) trained specialized models on each modality using 80% of training data, (2) generated predictions on held-out 20% to create 48-dimensional meta-features (16 classes × 3 modalities), (3) trained XGBoost meta-model to learn optimal combinations.

3. Results

3.1 Individual Modality Performance

Text Classification with TF-IDF Features:

- LogisticRegression: 0.3220
- RandomForestClassifier: 1.2664
- XGBClassifier: 0.7495
- **CalLinearSVC: 0.2996** (Best)
- CalibratedRandomForest: 1.0440
- CalibratedXGBClassifier: 0.5803

Text Classification with Word2Vec Features:

- **LogisticRegression: 0.5711** (Best for Word2Vec)

- RandomForestClassifier: 0.6029
- XGBClassifier: 0.6726
- CalLinearSVC: 0.6078
- CalibratedRandomForest: 0.6662
- CalibratedXGBClassifier: 0.5718

Text Classification with Neural Networks:

- MLP with Word2Vec features: 0.3659
- **MLP with TF-IDF (SVD-reduced): 0.3358** (Best neural approach)

Analysis: TF-IDF dramatically outperformed Word2Vec embeddings, with CalLinearSVC achieving the best text performance (0.2996). Calibrated models consistently outperformed their non-calibrated counterparts. Neural networks showed competitive performance but didn't surpass well-calibrated traditional methods.

Graph Classification with Node2Vec Features:

- LogisticRegression: 0.4380
- RandomForestClassifier: 0.3657
- **CalibratedRandomForest: 0.2768** (Best)
- CalibratedLinearSVC: 0.4801

Graph Classification with Custom Features:

- LogisticRegression: 2.0800
- RandomForestClassifier: 1.3889
- **CalibratedRandomForest: 1.3495** (Best for custom features)
- CalibratedLinearSVC: 2.0691

Analysis: Node2Vec embeddings significantly outperformed hand-crafted features across all algorithms. The best Node2Vec result (0.2768) was nearly 5x better than the best custom features result (1.3495), demonstrating the power of learned representations for graph data.

Price Classification:

- LogisticRegression: 2.6017
- RandomForestClassifier: 2.5153
- **CalibratedRandomForest: 2.1779** (Best)
- CalibratedLinearSVC: 2.2602

Analysis: Price features alone showed limited discriminative power, as expected due to category overlap and missing data. Tree-based methods outperformed linear models, suggesting non-linear price-category relationships.

3.2 Ensemble Results

Base Models Ensemble (Single Features): Using best performers from each modality:

- Text: CalLinearSVC with TF-IDF
- Graph: CalibratedRandomForest with Node2Vec
- Price: CalibratedRandomForest
- Meta-model: XGBClassifier
- **Result: 0.2428** (validation)

Optimized Models Ensemble (Single Features): Same base models with tuned meta-model:

- **Result: 0.2453, 0.2006 (Kaggle)** (Best)

Optimized Models Ensemble (Combined Features): Enhanced feature representations:

- Text: Combined TF-IDF + Word2Vec (SelectKBest k=20,000)
- Graph: Combined Node2Vec + Custom features
- Price: Same engineered features
- Meta-model: Tuned XGBClassifier
- **Result: 0.2385 (validation)** - slightly worse due to overfitting

Key Insights from Ensemble Results:

1. **Combined features only used with tuned models**: The combination strategy was only applied to optimized meta-models, not base classifiers
2. **Validation vs Kaggle discrepancy**: Significant gap (0.2385 vs 0.2006) indicates validation methodology issues
3. **Marginal improvement from feature combination**: Only 0.043 improvement (0.2428 → 0.2385) suggests diminishing returns
4. **Meta-model optimization importance**: Tuned XGBoost parameters were crucial for extracting value from combined features

Why Combined Features Showed Limited Improvement:

- **Feature redundancy**: TF-IDF and Word2Vec captured overlapping semantic information
- **Increased complexity**: Higher-dimensional meta-features (48D vs 36D) approached optimal complexity threshold
- **Computational overhead**: Significantly longer training time for minimal gains
- **Selection limitations**: Even with mutual information selection, some noise remained in combined feature space

4. Analysis: What Worked and What Didn't

4.1 Successful Strategies

Feature Engineering Excellence: Multi-modal approach captured complementary information effectively. Enhanced TF-IDF with multi-vectorizer strategy and n-grams significantly outperformed simpler approaches. Node2Vec embeddings proved superior to hand-crafted graph features by learning latent user behavior patterns.

Calibration Focus: Key insight that probability calibration matters more than model complexity for log-loss optimization. CalibratedClassifierCV consistently improved performance across all modalities.

Meta-Learning Success: Ensemble learned optimal confidence levels and combinations, with each modality contributing unique discriminative information.

Systematic Feature Selection: Two-stage selection strategy (chi-squared → mutual information) optimized for different feature types while managing computational complexity.

4.2 Challenges and Limitations

Computational Constraints: Limited computational resources prevented extensive neural network exploration and deeper hyperparameter tuning. Graph feature extraction required 35+ minutes, limiting experimentation. With more resources, we would explore deeper neural architectures for text features, potentially combining CNNs for character-level patterns with transformers for semantic understanding.

Validation Discrepancy: Validation scores in notebook (0.24+) didn't match Kaggle performance (0.2006), likely due to data leakage in validation splits or different train/test handling procedures. This made local optimization challenging and potentially misleading.

Class Imbalance Impact: High imbalance ratio (38.32) required careful handling through class weighting and calibration. Minority classes with <1,500 samples proved difficult to learn effectively, and log-loss metric was particularly sensitive to minority class predictions.

Feature Combination Limitations: Despite sophisticated selection methods, combining features within modalities yielded minimal improvements, suggesting feature redundancy and potential overfitting in high-dimensional spaces.

5. Technical Contributions

Advanced Feature Engineering: Novel multi-vectorizer TF-IDF approach capturing word, character, and brand patterns. Systematic feature selection pipeline optimized for mixed feature types and computational efficiency.

Sophisticated Ensemble: Meta-learning implementation with proper validation splits and extensive XGBoost hyperparameter tuning including regularization parameters. Demonstrated effective combination of heterogeneous modalities.

Systematic Evaluation: Rigorous framework with stratified splits, metric-specific optimization through calibration, and comprehensive algorithm comparison across modalities.

Scalable Methodology: Pipeline designed to handle large-scale multi-modal data with efficient feature selection and model training procedures.

6. Possible Improvements

Given more computational resources, we would pursue:

Advanced Neural Architectures: Transformer-based models for text processing, graph neural networks for product relationships, and end-to-end multi-modal fusion networks learning optimal feature combinations.

Enhanced Feature Engineering: Temporal patterns in co-viewing data, advanced graph community detection, and external data integration (reviews, ratings, category hierarchies).

Improved Ensemble Methods: Dynamic weighting based on input characteristics, hierarchical classification exploiting category relationships, and better uncertainty quantification methods.

7. Conclusion

This project successfully demonstrates multi-modal machine learning for product classification, achieving strong performance (0.2006 on Kaggle). Key insights include: calibration matters more than complexity for log-loss, learned embeddings outperform hand-crafted features, systematic feature selection is crucial for high-dimensional data, and intelligent ensembles effectively combine complementary information sources.

Despite computational constraints limiting neural network exploration, our systematic methodology provides a solid foundation for scaling to larger datasets and offers a template for similar e-commerce classification challenges. The finding that feature quality trumps quantity in ensemble learning provides valuable guidance for future multi-modal projects.