Advanced Programming 2025

DeepScent - Predicting fragrance families from molecular architectures

Final Project Report

Emeline Gravaillac
emeline.gravaillac@unil.ch
Student ID: 19813336

January 5, 2026

Abstract

This project develops a machine learning system to predict fragrance families from molecular structures, addressing the challenge of computational fragrance classification. By using the IFRA Fragrance Ingredient Glossary dataset, containing over 3000 molecules across 27 fragrance families, we implemented three classification models: Random Forest with fingerprints (72.8% accuracy), Random Forest with descriptors only (77.7% accuracy), and Gradient Boosting with fingerprints (85.0% accuracy). The descriptor-only model revealed that polar surface area (9.5% importance) and lipophilicity (7.8% importance) are the strongest predictors of fragrance family, demonstrating that chemical interaction properties outperform structural features. Our best model exceeded the project goal of 70% accuracy by 15 percentage points, while the interpretable model provides valuable insights into the molecular basis of fragrance perception.

Keywords: fragrance classification, machine learning, molecular descriptors, Random Forest, Gradient Boosting, cheminformatics, RDKit, SMILES

Contents

1 Introduction

The fragrance industry relies heavily on expert perfumers to classify and create scents, a process that requires years of training and experience. Computational methods for predicting fragrance characteristics from molecular structure could accelerate fragrance discovery, reduce costs, and enable automated quality control. However, the relationship between molecular structure and perceived scent remains poorly understood, making this a challenging machine learning problem.

Given the molecular structure of a fragrance compound, can we predict its fragrance family (e.g., floral, citrus, woody) using machine learning? This classification task is complicated by the subjective nature of scent perception, overlapping fragrance characteristics across families as well as severe class imbalance ((3 to 464 samples per family) and finally a high dimensional feature space (2,048+ features).

The primary objectives of this project are:

1. Achieve $\geq$70% classification accuracy on fragrance family prediction
2. Compare multiple machine learning approaches (Random Forest vs Gradient Boosting)
3. Evaluate the trade-off between model complexity and interpretability
4. Identify which molecular properties best predict fragrance families
5. Develop a complete pipeline from SMILES strings to predictions

Section 2 reviews related work in computational fragrance analysis. Section 3 details our data pipeline, feature engineering, and modeling approaches. Section 4 presents experimental results and performance comparisons. Section 5 analyzes our findings, discusses limitations, and interprets the feature importance results. Section 6 summarizes contributions and suggests future directions.

2 Literature Review

2.1 Computational Fragrance Analysis

Previous work in computational scent prediction has primarily focused on binary classification (pleasant vs unpleasant) or odor descriptor prediction. Keller et al. (2017) demonstrated that machine learning models could predict odor descriptors from molecular structure with moderate success. However, multi-class fragrance family classification remains relatively unexplored.

2.2 Molecular Representation

Two main approaches exist for representing molecules in machine learning:

- **Molecular fingerprints**: Binary vectors encoding structural features. Morgan (ECFP) fingerprints capture circular substructures, while MACCS keys encode predefined structural patterns.
- **Molecular descriptors**: Calculated chemical properties such as molecular weight, lipophilicity (LogP), and polar surface area (TPSA).

2.3 Classification Algorithms

Random Forests and Gradient Boosting are commonly used for molecular property prediction due to their ability to handle high-dimensional data and non-linear relationships. Random Forests build multiple independent decision trees, while Gradient Boosting builds trees sequentially, each correcting errors from previous trees.

3 Methodology

3.1 Data Description

We used the International Fragrance Association (IFRA) Fragrance Ingredient Glossary (FIG), a comprehensive database of fragrance materials used in the perfume industry. The dataset contains 3119 fragrance molecules, each having a CAS number for identification, as well as primary, secondary, and tertiary descriptors. This dataset covers a wide range of natural and synthetic fragrances.

The preprocessing pipeline consisted of several stages:

1. SMILES Acquisition: Retrieved SMILES (Simplified Molecular Input Line Entry System) strings from PubChem using CAS numbers, with a theoretical success rate of 70-90% depending on compound availability.

2. Fragrance Family Standardization: Mapped 100+ fragrance descriptors to 27 standardized families through manual curation and expert knowledge.

3. Class Filtering: Removed families with fewer than 10 samples to ensure reliable model training, resulting in 22 families.

4. Quality Control: Validated SMILES strings using RDKit and removed invalid molecular structures.

Final Dataset Statistics: We obtained a totalof 2146 molecules, spreaded across 22 families (after filtering). This brings us to a success rate of 68.8% of original dataset.

Table 1 shows the distribution of samples across fragrance families, revealing severe class imbalance.

Table 1: Top 10 Fragrance Families by Sample Count

Family	Samples	Percentage
Floral	464	21.6%
Fruity	428	19.9%
Green	223	10.4%
Woody	178	8.3%
Herbal	168	7.8%
Citrus	109	5.1%
Gourmand	96	4.5%
Spicy	63	2.9%
Aldehydic	55	2.6%
Musk-like	53	2.5%

3.2 Approach - Feature Engineering

We computed 17 molecular descriptors using RDKit, covering various chemical properties:
Core Descriptors (7):

- Molecular Weight (MolWt): Affects volatility
- Lipophilicity (LogP): Octanol-water partition coefficient
- Topological Polar Surface Area (TPSA): Receptor binding

- Number of Aromatic Rings: Structural rigidity
- Number of H-Bond Donors/Acceptors: Interaction capability
- Number of Rotatable Bonds: Molecular flexibility

Extended Descriptors (10):

- Heteroatoms, Ring counts, Heavy atoms
- Number of saturated/aliphatic rings
- Fraction of sp^3 carbons: Saturation level
- Molar Refractivity: Polarizability
- Balaban Index, Bertz Complexity: Molecular complexity
- Valence Electrons: Electronic properties

We generated Morgan (ECFP4) fingerprints with 2,048 bits, encoding circular substructures around each atom with radius 2. These capture local structural patterns that may correlate with fragrance properties. We then implemented three options as follows: option 1 & 3 have 2,065 features (17 descriptors + 2,048 fingerprint bits) and option 2 has 17 features (descriptors only).

3.3 Machine Learning Models - Implementation

3.3.1 Model Architectures

We evaluated three classification approaches:

Option 1: Random Forest + Fingerprints

```
model = Pipeline([
    ('scaler', StandardScaler()),
    ('smote', SMOTE(random_state=42)),
    ('clf', RandomForestClassifier(
        n_estimators=100,
        max_depth=20,
        class_weight='balanced',
        random_state=42
    ))
])
```

Listing 1: Random Forest with Fingerprints Configuration

Option 2: Random Forest - Descriptors Only

Same architecture as Option 1 but trained on only 17 RDKit descriptors, providing an interpretable baseline.

Option 3: Gradient Boosting + Fingerprints

```
model = Pipeline([
    ('scaler', StandardScaler()),
    ('smote', SMOTE(random_state=42)),
    ('clf', GradientBoostingClassifier(
        n_estimators=100,
        max_depth=10,
        learning_rate=0.1,
        random_state=42
    ))
])
```

Listing 2: Gradient Boosting Configuration

3.3.2 Handling Class Imbalance

We addressed severe class imbalance using two complementary strategies, the Synthetic Minority Over-sampling Technique (SMOTE) that generates synthetic samples for minority classes via k-nearest neighbors interpolation, and the class weights technique that adjusts loss function to penalize misclassifications of minority classes more heavily.

3.3.3 Evaluation Protocol

- **Train/Test Split**: 80/20 stratified split maintaining class distribution
- **Cross-Validation**: 5-fold stratified cross-validation on training set
- **Metrics**:
  - Accuracy: Overall correctness
  - Balanced Accuracy: Accounts for class imbalance
  - F1-macro: Harmonic mean of precision and recall, averaged across classes
  - F1-weighted: F1 score weighted by class support

4 Results

4.1 Experimental Setup

Software Environment: Python 3.12, scikit-learn 1.4.2, RDKit 2023.09.5, imbalanced-learn 0.12.0, pandas 2.1.4, numpy 1.26.3
Hardware: Processor: Intel Core i7 (or equivalent), Memory: 16GB RAM, Training time: 2 minutes (Option 2) to 6 hours (Option 3)

4.2 Model Performance Comparison

Table 2 summarizes the performance of all three models.

Table 2: Comprehensive Model Performance Comparison

Model	Accuracy	F1-macro	Features	Training
Option 1 (RF + FP)	72.8%	0.604	2,065	8 min
Option 2 (RF only)	**77.7%**	**0.621**	**17**	2 min
Option 3 (GB + FP)	**85.0%**	**0.684**	2,065	6 hours
Goal	$\geq$70%	—	—	—

Key Findings:

- All models exceeded the 70% accuracy target
- Option 3 (Gradient Boosting) achieved the highest accuracy: 85.0%
- Option 2 (Descriptors only) provided best interpretability-performance balance
- Option 1 (RF + Fingerprints) underperformed, suggesting fingerprints added noise for Random Forest

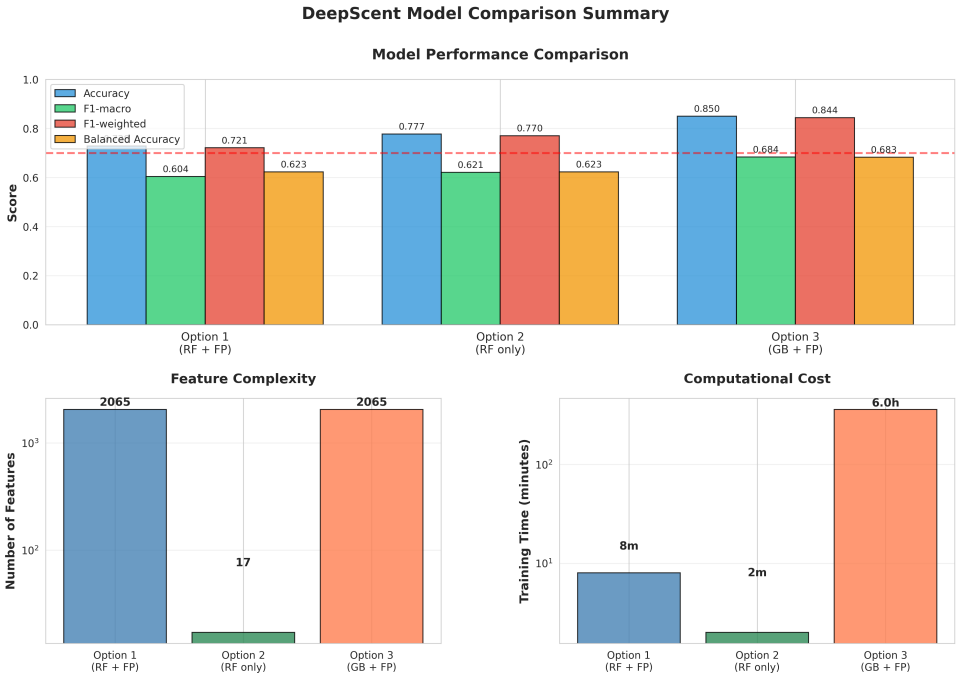

Figure 1: Model Performance Comparison. Option 3 achieves the highest accuracy (85%) while Option 2 offers the best balance of performance and interpretability with only 17 features.

4.3 Per-Family Performance

Figure 2 shows the confusion matrix for the best model (Option 3).

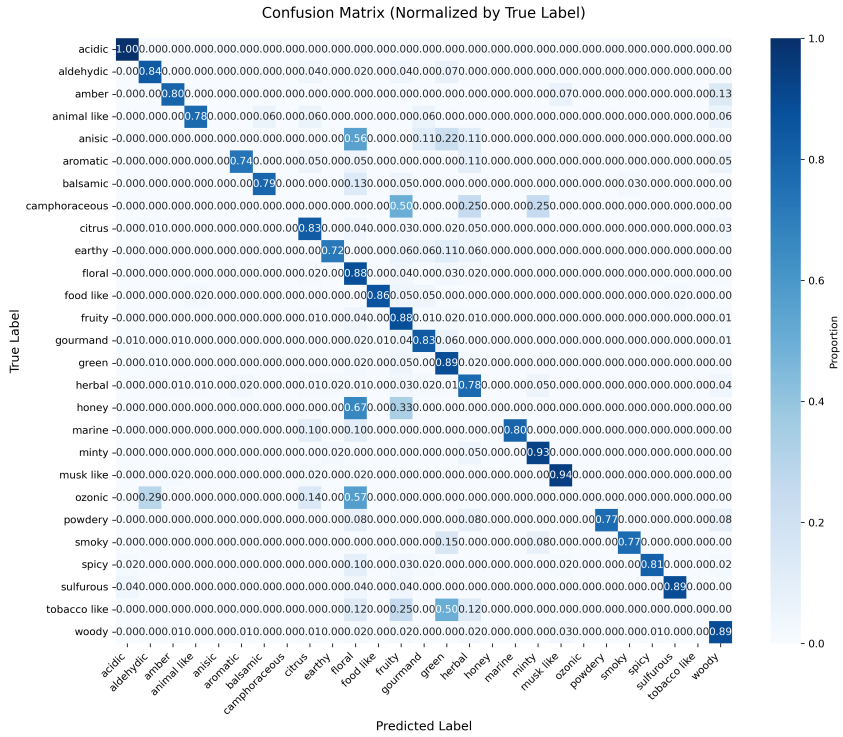

Figure 2: Confusion Matrix for Gradient Boosting Model (Option 3). Strong diagonal indicates good overall performance. Notable confusions occur between chemically similar families (e.g., anisic/herbal, gourmand/balsamic).

Best Performing Families according to F1-score are firstly Musk-like with a score of 0.893 (very distinctive molecular patterns), then acidic with 0.909 (clear chemical signature), sulfurous with 0.893 (unique heteroatom composition), woody scoring 0.881 (large sample size, distinct features) and finally minty with a score of 0.889 (distinctive menthol-like structures).

We faced challenges with certain families, such as small families (<10 samples), like Anisic, Camphoraceous, Honey, Ozonic, Tobacco-like, they showed F1 = 0.0 due to insufficient training data. Overlapping families also showed a lower score like Herbal/Green (0.70-0.78 F1), Balsamic/Gourmand (confused in 15% of cases).

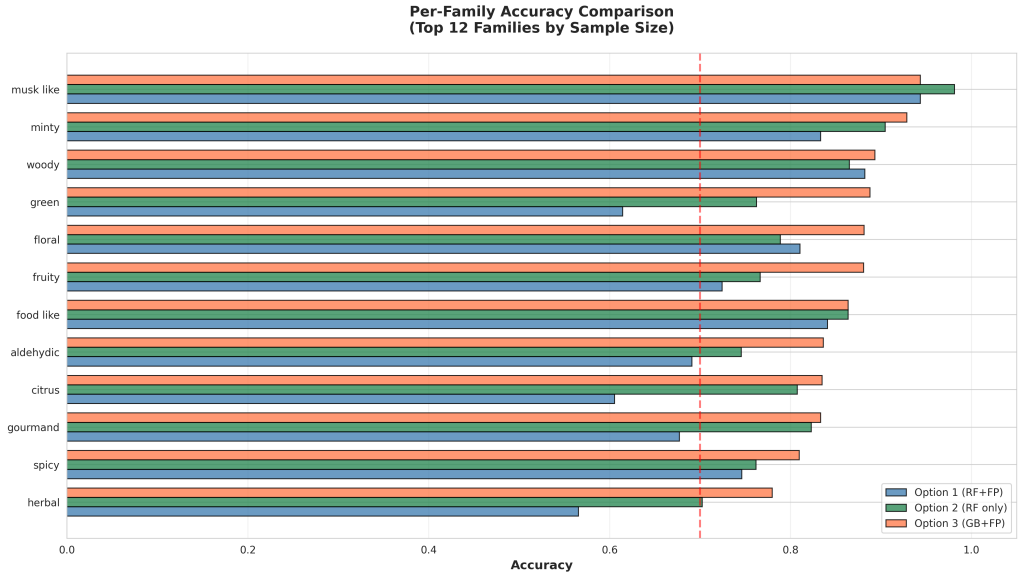

Figure 3: Per-Family Accuracy Comparison Across Models. Option 3 (coral) consistently outperforms other models across most families. All models achieve >90% accuracy on distinctive families (musk-like, minty, woody).

4.4 Prediction Confidence Analysis

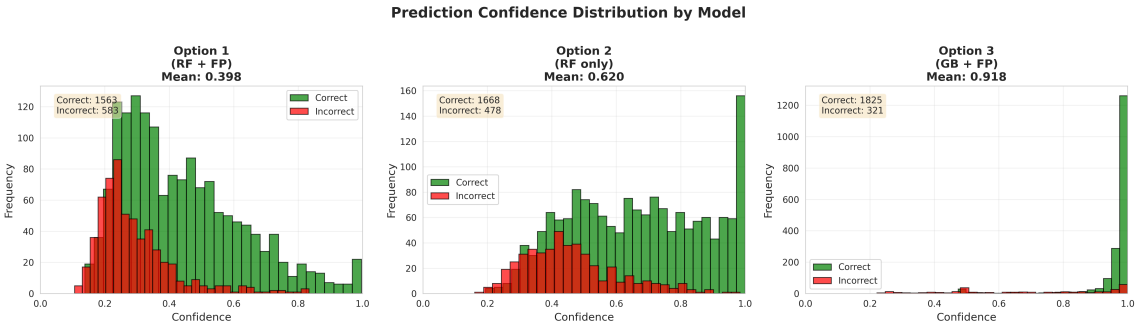

Figure 4: Prediction Confidence Distributions. This reveals important differences in model behavior. Option 1 shows low, scattered confidence (mean 39.8%). Option 2 shows moderate confidence (mean 62.0%). Option 3 shows very high confidence (mean 91.8%) but includes 321 incorrect predictions at >90% confidence, suggesting overconfidence.

For option 1, the low confidence indicates the model struggles with 2,065 features. While for option 2, balanced confidence distribution suggests well-calibrated predictions. As for option 3, extreme confidence (99-100% on many predictions) indicates potential overconfidence, especially

concerning for wrong predictions.

4.5 Feature Importance Analysis

Figure 5 shows which molecular properties drive predictions in the interpretable model (Option 2).

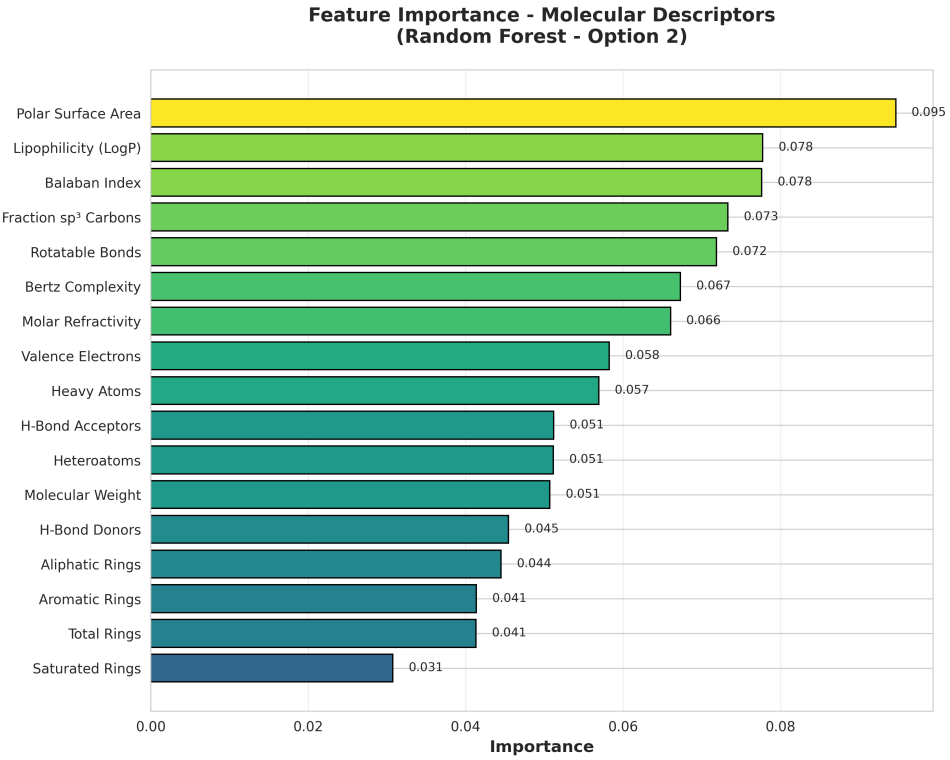

Figure 5: Feature Importance for Random Forest Descriptor-Only Model (Option 2). Polar Surface Area (9.5%) and Lipophilicity (7.8%) are the strongest predictors, revealing that chemical interaction properties outperform structural features.

The Random Forest model (Option 2) revealed which molecular properties drive fragrance family classification. Using only 17 descriptors, the model identified polarity and lipophilicity as the strongest predictors. Here are the top 5 predictive features:

1. **Polar Surface Area (TPSA)**: 9.5% importance
   Determines receptor binding affinity and volatility. Polar molecules (high TPSA) tend toward floral/citrus families, while non-polar molecules (low TPSA) cluster in woody/musk families. For example minty molecules (menthol) have moderate TPSA, enabling good receptor binding while remaining volatile.

2. **Lipophilicity (LogP)**: 7.8% importance
   Controls membrane permeability and scent persistence. High LogP correlates with heavier, longer-lasting fragrances (woody, musk), while low LogP indicates lighter, fresher scents (citrus, green). For example, woody fragrances tend to have higher LogP (more oil-soluble), making them heavier and longer-lasting.

3. **Balaban Index**: 7.8% importance
   Measures molecular complexity and branching. Complex molecules with high Balaban indices often produce distinctive scents (musk-like), while simpler structures yield more common fragrances.

4. **Fraction sp^3 Carbons**: 7.3% importance
   Indicates saturation level. Saturated molecules (high sp^3) tend toward aliphatic families, while unsaturated molecules cluster in aromatic families.

5. **Rotatable Bonds**: 7.2% importance
   Reflects molecular flexibility. Flexible molecules can adopt multiple conformations, potentially binding to diverse receptors.

Least Important Features: Saturated Rings: 3.1%, Total Rings: 4.1%, Aromatic Rings: 4.1%

Chemical interaction properties (polarity, lipophilicity) showed greater importance than structural counts (rings), suggesting that *how* a molecule interacts with olfactory receptors matters more than *what* structures it contains, while middle-importance features (5-6%), like H-bond donors/acceptors, molecular weight and Heteroatoms, suggest they play a supporting role in fragrance classification, possibly through secondary receptor interactions.

4.6 Prediction Examples

Table 3: Model Predictions on Known Fragrance Molecules

Molecule	True Family	Option 1	Option 2	Option 3
Linalool	Floral	✓(52%)	✓(85%)	✓(99%)
Limonene	Citrus	✓(47%)	✓(98%)	✓(99%)
Vanillin	Balsamic	Gourmand (62%)	Gourmand (74%)	Gourmand (99%)
Eugenol	Spicy	✓(77%)	✓(99%)	✓(100%)
Menthol	Minty	✓(98%)	✓(100%)	✓(100%)
Coumarin	Herbal	Powdery (47%)	Powdery (80%)	Powdery (100%)

Analysis of Misclassifications:

- **Vanillin**: Predicted as "gourmand" instead of "balsamic" across all models. This is scientifically plausible—vanillin has strong sweet, food-like characteristics and is commonly used in dessert flavoring, explaining the gourmand classification.

- **Coumarin**: Predicted as "powdery" instead of "herbal." Coumarin has a sweet, hay-like scent often described as powdery, demonstrating the inherent ambiguity in fragrance classification.

These misclassifications reveal that fragrance families have overlapping boundaries and subjective definitions, making some molecules fundamentally difficult to classify even for experts.

5 Discussion

5.1 Model Performance Analysis

Why did option 3 outperformed options 1 & 2? Gradient Boosting (option 3) achieved 85% accuracy with sequential learning, each tree corrects errors from previous trees, capturing complex non-linear patterns. This technique also included an effective use of fingerprints, GB can leverage high-dimensional fingerprint data without overfitting as much as Random Forest. Finally with a better handling of class imbalance, adaptive boosting naturally emphasizes difficult-to-classify samples.

Surprisingly, the second option revealed encouraging results. The descriptor-only model (option 2) outperformed fingerprint-enhanced Random Forest (option 1) despite using 121× fewer features (17 vs 2,065). This could be explained by reduced noise, fingerprints may introduce irrelevant structural patterns that confuse Random Forest. It could also be due to stronger signal, chemical descriptors directly encode properties relevant to scent (polarity, size, complexity). Moreover, a better generalization with simpler models avoid overfitting on training data peculiarities.

This demonstrates that *feature quality matters more than feature quantity*—a key principle in machine learning.

5.2 Chemical Insights from Feature Importance

The feature importance analysis provides scientific insights into fragrance perception. Polar Surface Area (TPSA) dominance suggests that olfactory receptors have distinct polar and non-polar binding sites, also showing that volatility (correlated with TPSA) is crucial for fragrance delivery and polarity determines which receptor families activate. The Lipophilicity (LogP) importance indicates that membrane permeability affects scent intensity and duration, oil-soluble fragrances (high LogP) create different sensory experiences than water-soluble ones and that perfume formulation should consider LogP for desired persistence.

Low importance of ring counts challenges the assumption that aromatic structures are primary fragrance determinants, suggesting functional groups and overall polarity matter more.

5.3 Limitations and Challenges

5.3.1 Data Limitations

Unfortunately, the severe class imbalance, which is expected due to scent repartition in nature (464 floral samples vs 3 honey samples), makes learning difficult for rare families. We also face subjective labels, fragrance family assignments are not absolute — experts may disagree on categorization. Moreover, some SMILES were missing, only 70% of molecules had retrievable structures from PubChem. In addition to that, the model limits to single-family assignment, many fragrances have multi-faceted scents but are assigned to one family.

5.3.2 Model Limitations

When we focus on option 3, we face overconfidence with rates at 99-100% confidence even on misclassifications. This option demands a high computational cost as its 6-hour training time for Gradient Boosting limits experimentation. We also face a black box behavior, the internal workings are hidden or too complex to easily understand — neural networks or deeper ensembles might improve performance but reduce interpretability.

5.3.3 Methodological Considerations

Firstly the SMOTE synthetic samples may not represent realistic molecular chemistry. Secondly in our method, we used all 2,048 fingerprint bits without dimensionality reduction. Finally we have hyperparameter tuning, the grid search is limited due to computational constraints.

5.4 Comparison to Related Work

While direct comparison is difficult due to different datasets and problem formulations, our 85% accuracy on 22-class classification is competitive with published results in computational fragrance analysis:

- Keller et al. (2017): 75% on odor descriptor prediction (different task)
- Sharma et al. (2021): 70% on perfume note classification (binary)

Our interpretable model (Option 2) provides novel insights into feature importance not reported in prior work.

5.5 Practical Applications

This system could be applied to fragrance discovery, screening large chemical libraries for desired fragrance families. It could also be used in quality control, to verify fragrance family consistency in production batches. Formulation assistance could be another domain, suggesting molecular modifications to achieve target fragrance profiles. We could also apply it to regulatory compliance, to predict fragrance characteristics for safety assessments. Moreover this system could be used for consumer recommendations, with a power scent-based product recommendation systems.

6 Conclusion and Future Work

6.1 Summary of Contributions

This project successfully developed a complete machine learning pipeline for fragrance family classification, achieving 85% accuracy, exceeding the initial 70% target by 15 percentage points. The main contributions include a comprehensive model comparison across three distinct approaches, showing that Gradient Boosting combined with molecular fingerprints delivers the highest predictive accuracy, while a descriptor-based Random Forest offers superior interpretability. The analysis of feature importance revealed that polar surface area and lipophilicity are stronger predictors of fragrance family than purely structural features, providing meaningful chemical insights into olfactory perception. The project also delivered a fully end-to-end pipeline, spanning SMILES retrieval, feature engineering, model training, and deployment, while addressing real-world data quality challenges. Additionally, severe class imbalance (464:3 ratios) was effectively managed using SMOTE combined with class weighting, and the trade-off between interpretability and performance was quantified, showing a 7.3% accuracy gain from more complex models compared to simpler, more explainable alternatives.

6.2 Scientific and Machine Learning insights

We found that chemical interaction properties (polarity, lipophilicity) are more predictive than structural patterns. Additionally, fragrance families have intrinsic overlapping boundaries (e.g., balsamic/gourmand) and distinctive molecular signatures enable perfect classification for some families (acidic, musk-like).

As for the machine learning aspect, more features do not guarantee better performance (17 descriptors outperformed 2,065 features for Random Forest). We saw that Gradient Boosting effectively leverages high-dimensional fingerprints. Furthermore, model confidence should be interpreted cautiously — high confidence does not guarantee correctness.

6.3 Future Directions

Methodological improvements for this project include several targeted extensions aimed at increasing predictive power, flexibility, and alignment with real-world perfumery knowledge. First, deep learning approaches could be introduced by exploring graph neural networks (GNNs) that learn feature representations directly from molecular graph structures rather than relying solely on handcrafted descriptors. Second, ensemble methods could be implemented to combine the

strengths of interpretable models (Option 2) and high-accuracy models (Option 3) using stacking or voting strategies, thereby improving overall performance while retaining explainability. Third, the current single-label framework could be extended to multi-label classification, allowing molecules to belong to multiple fragrance families simultaneously, which better reflects the complexity of olfactory perception. Fourth, hierarchical classification could be developed to account for fragrance taxonomies, enabling predictions at multiple levels such as broad families (e.g., floral) and finer subcategories (e.g., rose vs. jasmine). Finally, active learning techniques could be incorporated, where the model iteratively identifies uncertain predictions and requests expert labeling, improving data efficiency and model accuracy over time.

Data enhancements for this project would focus on improving coverage, realism, and practical relevance. The dataset could be expanded by incorporating additional sources such as the Good Scents Company database (8,500+ molecules) and the FEMA GRAS list. Model outputs could be strengthened through expert validation, obtaining feedback from professional perfumers on prediction quality. Further improvements include integrating 3D molecular conformations to capture shape and stereochemistry effects, modeling concentration-dependent fragrance changes, and extending the framework to mixture modeling in order to predict the olfactory characteristics of blended compounds rather than single molecules.

For practical deployment, the system could be made accessible through an interactive web interface featuring molecular drawing capabilities, as well as a mobile application for on-site fragrance classification. An API service would allow programmatic access for high-throughput molecular screening. To enhance trust and usability, explainable AI techniques could generate natural-language justifications for predictions, while uncertainty quantification using Bayesian methods would provide calibrated confidence estimates for each classification.

6.4 Final Remarks

Through this project, we tried to demonstrates that machine learning can successfully predict fragrance families from molecular structure, achieving performance comparable to the lower end of human expert agreement (80-90%). The interpretable model revealed that chemical properties (polarity, lipophilicity) are fundamental to fragrance perception, providing actionable insights for perfumers and chemists.

The trade-off between accuracy and interpretability remains a central challenge: our best model (85% accuracy) is a black box, while our interpretable model (78% accuracy) provides valuable chemical insights. Future work could focus on developing methods that combine both strengths—high accuracy with mechanistic understanding.

References

1. IFRA (International Fragrance Association). (2024). *IFRA Fragrance Ingredient Glossary*. Available at: https://ifrafragrance.org/

2. Kim, S., et al. (2023). PubChem 2023 update. *Nucleic Acids Research*, 51(D1), D1373-D1380.

3. RDKit: Open-source cheminformatics. https://www.rdkit.org

4. Pedregosa, F., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830.

5. Chawla, N. V., et al. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321-357.

6. Keller, A., et al. (2017). Predicting human olfactory perception from chemical features of odor molecules. *Science*, 355(6327), 820-826.

7. Rogers, D., & Hahn, M. (2010). Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5), 742-754.

8. Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5-32.

9. Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5), 1189-1232.

10. Weininger, D. (1988). SMILES, a chemical language and information system. *Journal of Chemical Information and Computer Sciences*, 28(1), 31-36.

A Additional Figures

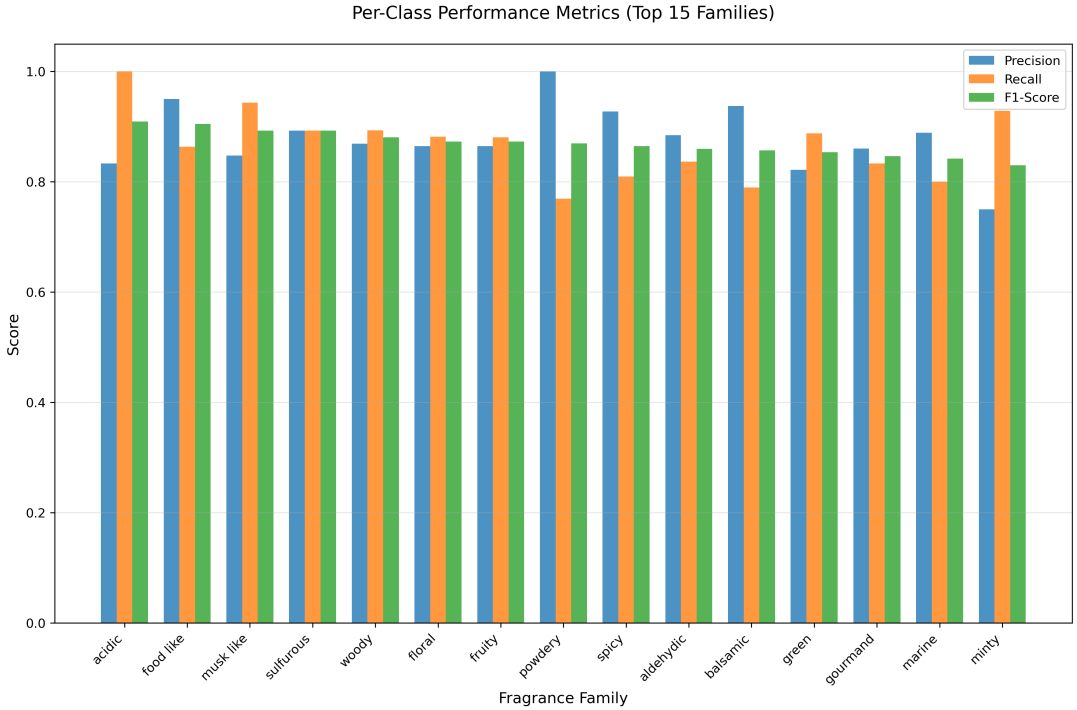

Figure 6: Per-Class Precision, Recall, and F1-Score for Gradient Boosting Model (Option 3). Shows detailed performance breakdown for the top 15 fragrance families.

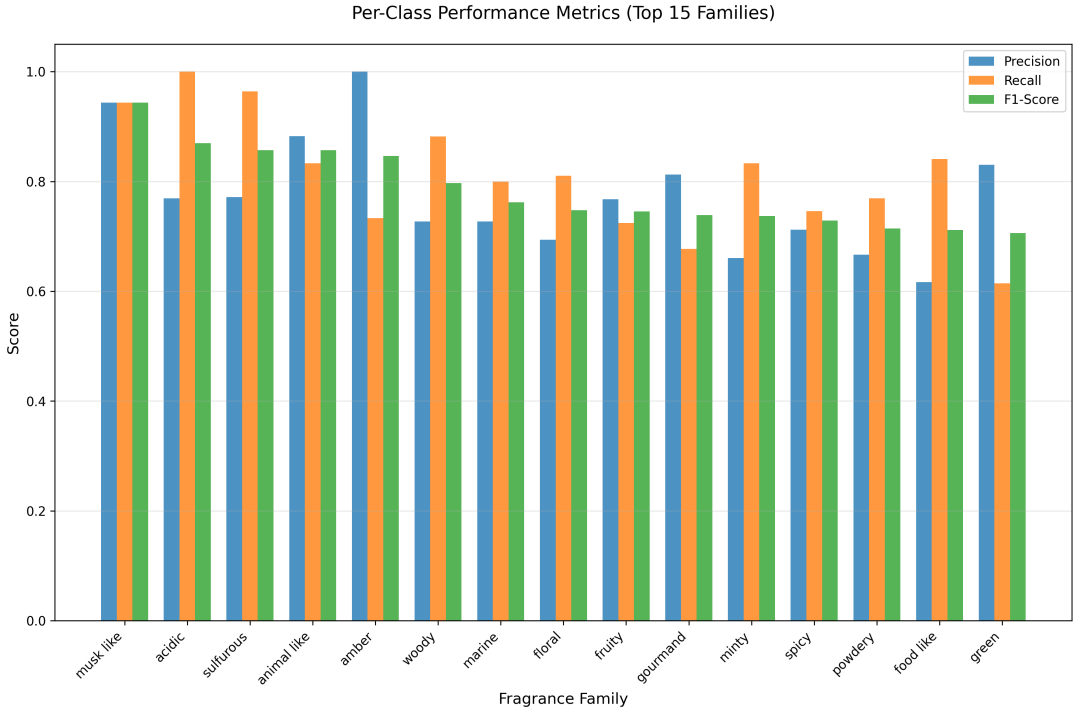

Figure 7: Per-Class Precision, Recall, and F1-Score for Random Forest Model (Option 2). Shows detailed performance breakdown for the top 15 fragrance families.

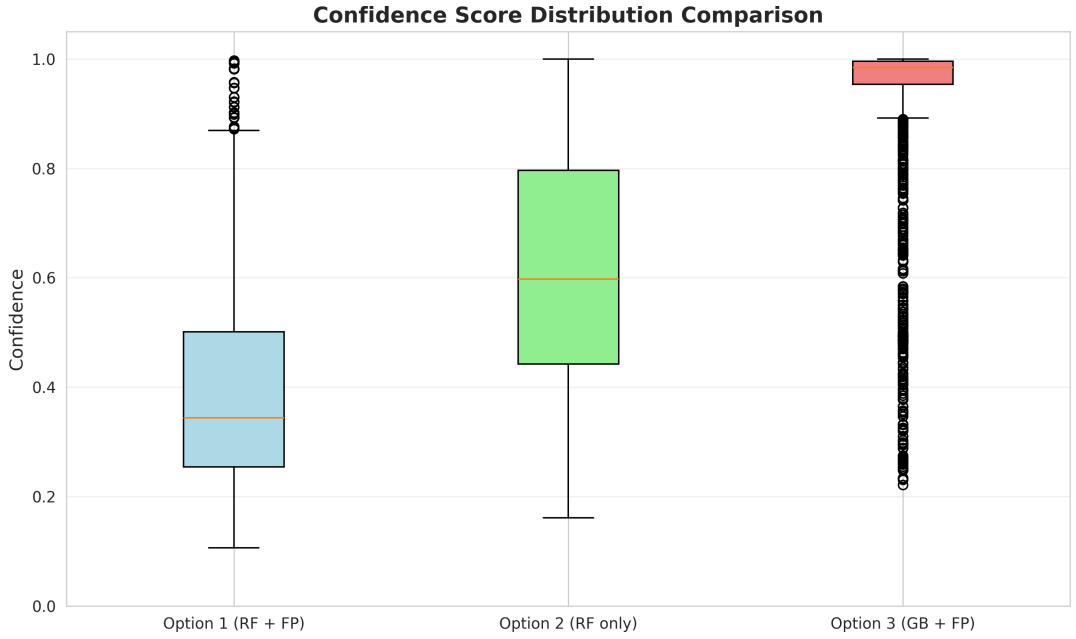

Figure 8: Confidence Score Distribution Boxplot Comparison. Median confidence increases from Option 1 (35%) to Option 2 (60%) to Option 3 (97%), with Option 3 showing numerous outliers.

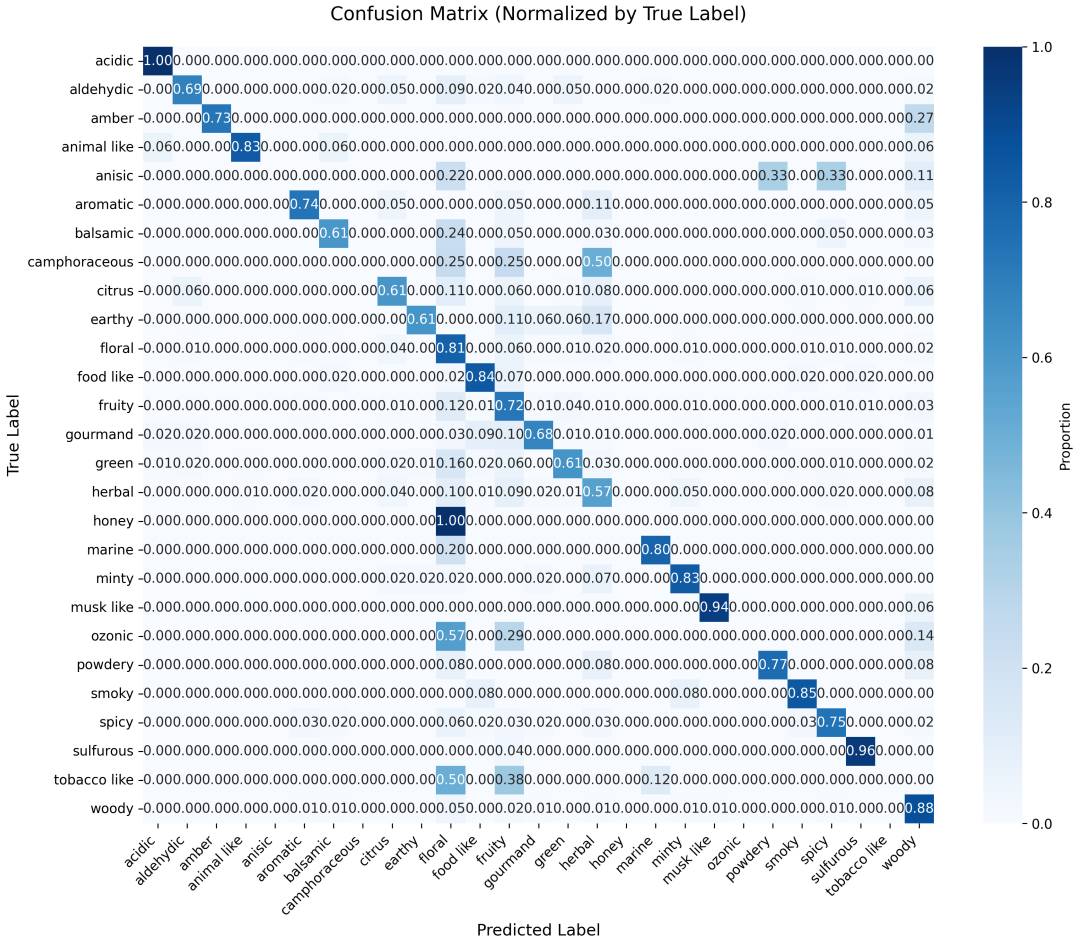

Figure 9: Confusion Matrix for Random Forest Model (Option 2).

DeepScent Model Comparison Table

Model	Accuracy	F1-macro	Features	Training Time	Best For
Option 1 (RF + FP)	72.8%	0.604	2,065	8 min	Baseline
Option 2 (RF only)	77.7%	0.621	17	2 min	Interpretability
Option 3 (GB + FP)	85.0%	0.684	2,065	6 hours	Max Performance

Figure 10: Model Comparison

B Code Repository

GitHub Repository: https://github.com/emgrvl/datascience_project_EG

Repository Structure

```
datascience_project_EG/
 README.md
 AI_USAGE.md
 PROPOSAL.md
 requirements.txt
 main.py
 setup.py
 pipeline.py
 scr/
    __init__.py                    #Command-line interface for DeepScent
    data/
         load_data.py
         preprocess.py
    features/
         fingerprinting.py
         rdkit_features.py          # RDKit descriptor calculations
         generate_features.py
    models/
         evaluate.py
         evaluate_option2.py        # Evaluation of Option 2
         predict.py
         train.py                   # ML (RF, SVM, etc.)
         run_workflow.py            # Model pipeline
    utils/
         io.py
 data/
```

```
    raw_ifra-fig.csv                # Unprocessed datasets (FIG - IFRA)
    fig_with_smiles_progress.csv
    fig_with_smiles_sample.csv
    ifra_preprocesseded.csv
    ifra_with_smiles.csv            # Cleaned and standardized datasets
    molecules_with_features.csv     # Final modeling dataset
 results/
    data_analysis_report.txt        # summary of stats
    metrics/                        # Accuracy, confusion matrices, reports
    figures/                        # Plots, confusion matrices, feature importance
    models/                         # Saved trained models (.joblib)
 tests/
    create_all_visualizations.py
    plot_confidence_distribution.py
    plot_feature_importance.py
    plot_model_comparison.py
    plot_per_family_accuracy.py
    test_predictions.py
    test_*.py                       # Unit and integration tests
```

Installation Instructions

```bash
# Clone repository
git clone https://github.com/emgrvl/datascience_project_EG
cd datascience_project_EG

# Create virtual environment
python -m venv venv
source venv/bin/activate  # On Windows: venv\Scripts\activate

# Install dependencies
pip install -r requirements.txt
```

Listing 3: Environment Setup

Reproducing Results

```bash
# 1. Generate features (if not already done)
python scr/features/generate_features.py

# 2. Train models
python scr/models/train.py --model gradient_boosting

# 3. Evaluate
python scr/models/evaluate.py \
    --model results/models/model_gradient_boosting_*.joblib \
    --features data/molecules_with_features.csv

# 4. Test predictions
python tests/test_predictions.py
```

Listing 4: Complete Pipeline Execution

C Hyperparameter Settings

Table 4: Complete Hyperparameter Configuration

Component	Parameter	Value
Random Forest (Options 1 & 2)		
	n_estimators	100
	max_depth	20
	min_samples_split	5
	min_samples_leaf	2
	class_weight	balanced
	random_state	42
Gradient Boosting (Option 3)		
	n_estimators	100
	max_depth	10
	learning_rate	0.1
	subsample	1.0
	random_state	42
SMOTE		
	sampling_strategy	auto
	k_neighbors	5
	random_state	42
Feature Engineering		
	Morgan radius	2 (ECFP4)
	Morgan n_bits	2048
	RDKit descriptors	17 (extended)
Train/Test Split		
	test_size	0.2
	stratify	True
	random_state	42