

Efficient Fusion Techniques for Result Diversification and Image Interestingness Tasks

Prabavathy Balasundaram^{1,*,\dagger}, G Gnana Sai^{2,*,\dagger}, Kishore N^{2,\dagger}, Odirva M^{2,\dagger},
Makesh Vaibhav A.G^{2,\dagger}, Naren Srinivasan Murali^{2,\dagger} and Parlapalli Sai Harshith^{2,\dagger}

¹Faculty, Department of Computer Science, Sri Sivasubramaniya Nadar College of Engineering, Chennai, Tamil Nadu, India

²UG Student, Sri Sivasubramaniya Nadar College of Engineering, Chennai, Tamil Nadu, India.

Abstract

Result diversification aims to retrieve a set of images that is both relevant and diverse, effectively capturing the essence of a given query. Image interestingness aims to fulfill the need for accurately assessing and predicting the level of interest in images, enabling better user experience and content organization. These two tasks can use inducer fusion, which combines the outputs of multiple inducers to improve the accuracy and robustness of prediction models. In this work, independent and ensemble ML techniques were used to solve the challenges in inducer fusion. Experimental validation was carried out on *Result diversification* and *Image interestingness* datasets of ImageCLEF2023-Fusion task. Our research contributes to advancing the field of inducer fusion and improving the performance of result diversification and image interestingness tasks.

Keywords

Result diversification, Image interestingness, Inducer fusion, Machine learning techniques, Ensemble Machine learning techniques

1. Introduction

With the exponential growth of digital imagery on the internet, effective image retrieval systems have become indispensable for users seeking visual content. Traditional image search engines primarily rely on content-based features and textual metadata to generate a ranked list of visually similar images. However, this approach often falls short in providing diverse search results, leading to redundancy and limited exploration of the search space. The [1]-[2]-diversification task was introduced to address this limitation and encourage the development of techniques that enhance result diversification.

The proliferation of visual content on various platforms necessitates effective techniques for predicting image interestingness. Accurately determining the level of interestingness associated with images holds immense value in applications such as image search, recommendation systems,

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece

*Corresponding authors.

^{\dagger}These authors contributed equally.

✉ prabavathyb@ssn.edu.in (P. Balasundaram); gnanasai2111012@ssn.edu.in (G. G. Sai);
kishore21110289@ssn.edu.in (K. N); olirva2110544@ssn.edu.in (O. M); makesh2110629@ssn.edu.in (M. V. A.G);
naren2110695@ssn.edu.in (N. S. Murali); saiharshith2110036@ssn.edu.in (P. S. Harshith)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

and content curation. The ability to automatically rank and retrieve interesting images not only enhances user satisfaction but also streamlines information retrieval processes. In recent years, substantial progress has been made in the development of computational models and techniques for image interestingness prediction. However, the diverse and subjective nature of interestingness poses significant challenges. To address these challenges, Constantin et al [9] introduced the *Interestingness10k* dataset, which serves as a standardized benchmark for evaluating image interestingness prediction methods.

This paper presents a study on result diversification and image interestingness predictions using fusion techniques. Furthermore, the research objective is to investigate the effectiveness of inducer fusion, a technique that combines the outputs of multiple inducers, in enhancing prediction performance. Inducer fusion aims to leverage the strengths of individual inducers and mitigate their weaknesses, ultimately resulting in a more accurate and robust prediction model.

2. Related Work

The existing work related to **Result diversification** and **Image Interestingness** are summarised below.

Hai-tao, Yu et al [3] proposes a framework dubbed MO4SRD for search result diversification. While the current methods in use rely on a sequential selection procedure, MO4SRD suggests a score-and-sort approach based on direct metric optimization. It represents the diversity score of each document using probability distributions, enabling the development of different variations of diversity metrics. A probabilistic neural scoring function that takes into account cross-document interaction and permutation equivariance is incorporated into the system. MO4SRD is tested on the four standard test datasets released in the diverse tasks of TREC Web Track from 2009 to 2012 which suggests that it performs better than the current approaches.

Shreya Sriram et al [4] proposed an ensembled approach for web search result diversification using neural network models. The data was obtained from the Retrieving Diverse Social Images Task dataset. Different networks namely, Multilayer Perceptron, Ridge Regressor using Grid Search and Keras Regressor using Sequential model were ranked based on MAE. These ranks and the models were fed as input for the Voting Regressor. The performance of the voting regressor can again be measured with MAE. Among the ten best submissions done to ImageClef 2022, the best F1 score and CR score were 0.5604 and 0.4384 respectively.

Lekshmi Kalinathan et al [5] presented a fusion approach for web search result diversification using machine learning algorithms. The data was obtained from the Retrieving Diverse Social Images Task dataset. A voting regressor of three predictor models K Nearest Regressor, Decision Tree Regressor and SVM was used to predict the similarity scores of the models in the validation dataset. Of the 10 best submissions done to ImageClef 2022, the best F1 score and CR score were found to be 0.5634 and 0.4414 respectively.

Maria, Shoukat et al [6] presented investigation on predicting media interestingness scores using a novel late fusion framework. The individual inducers' scores are extracted from the Interestingness10k dataset which are provided by the task organizers. The proposed framework combines multiple algorithms and employs two fusion strategies: naive fusion and merit-based

fusion. The results revealed that the proposed late fusion framework consistently outperformed alternative approaches, exhibiting superior predictive accuracy and robustness. Overall, this paper offers a comprehensive exploration of media interestingness prediction, providing a valuable contribution to the existing literature.

Ying, Dai et al [7] have proposed two image interestingness models with different convolutional neural network architectures and improves on their image aesthetic score (AS) prediction by an ensemble. The models are trained on two datasets, CUHK-PQ and XihAA datasets. One model extracts the subject of the image for predicting the image’s aesthetic score, and the other extracts the holistic composition for the prediction. It is found that these models trained on the XiheAA dataset seem to learn the latent photography principles, though it cannot be said that they learn the aesthetic sense. The aggregated model improves the F1 value by 5.4% and 33.1% compared to the first and second model respectively.

V. Kalakota et al [8] proposed a model to retrieve diverse images of a particular landmark location that cover different aspects of a query. Images required are obtained from the Flickr Div150Cred dataset. Flickr Baseline Ranking Algorithm and a re-ranking strategy are applied to retrieve the most relevant images out of all the possible set of images using the provided textual metadata. A fusion-based strategy is employed to ensemble several cluster models and a final summary of the query location is produced by selecting images from different clusters. The model is evaluated based on P@10, CR@10, F1@10, P@20, CR@20 and F1@20. The proposed method achieved a start-of-the-art performance on precision scores and F1 Score for images retrieved 30 and above. Cluster Recall scores still need slight improvement for 10 or 20 images being retrieved. Future work will be devoted to improving cluster recall metric without affecting the initial precision scores.

3. Task and Dataset Description

3.1. Result Diversification

The dataset used for this task is extracted from [2]. The data corresponds to the Retrieving Diverse Social Images Task dataset [10]. An inducer is a model which predicts images related to a query. The outputs from 56 inducers, representing a total of 123 queries are split into devset (56 inducers for 60 queries) for training and testset (56 inducers for 63 queries) for testing. The

Table 1
Attributes of Inducer File for Result Diversification

Field	Representation
query id	represents the unique id of the query
photo id	the unique id of the photo represented by the entry
rank	rank of the photo
sim	similarity score of the photo to the query
run name	a general name for the inducer

task is to diversify the results of image search. This fusion task is a retrieval task, where the

similarity scores of each image with the query is generated. Each entry or row in these files is of the format as given below in the Table 1.

3.2. Media Interestingness

The Media interestingness fusion task corresponds to the problem of predicting the interestingness of a particular image. An inducer is responsible to determine the interestingness of the given images. The output of the inducer consists of the relevant images, their interestingness classification and score. However, a single inducer is disadvantageous for application in certain areas due to low precision and lack of performance. To tackle this problem, ensembling, a technique that aggregates the predictions of several inducers, is used. The ensembled system is expected to be superior when compared to the highest-performing individual inducer. The data

Table 2

Attributes of Inducer File for Media Interestingness

Field	Representation
video id	the id of the video from which image is extracted
image id	the id of the image
classification	classification of the image
score	the interestingness score of the image
run name	a general name for the inducer

for this task is extracted and corresponds to the Interestingness10k dataset [Constantin2021b]. The output data from 29 inducers, representing visual interestingness predictions for 2435 images, is stored in separate text files for each inducer. Each entry of these files is as per the format given in Table 2.

4. Methodologies used

Different machine learning algorithms like Elastic net, Gradient Boosting Regressor, Decision Tree were employed for the result diversification task and XGBoost Classifier, k-Nearest Neighbors Classifier and Decision Tree were employed for the image interestingness task.

4.1. XGBoost Classifier

XGBoost is an efficient machine learning algorithm known for its ensembling-based approach. It combines multiple decision tree models sequentially, leveraging gradient boosting to improve predictions continuously by addressing errors made by previous trees. XGBoost avoids overfitting and provides a range of hyperparameters for optimisation through regularisation approaches. It has advanced capabilities like handling missing values and parallel processing and can handle large-scale datasets effectively. Metrics including accuracy, precision, recall, and F1-score are used in evaluation

4.2. K-Nearest Neighbors Classifier

k-Nearest Neighbors (k-NN) is a machine learning algorithm used for classification and regression tasks. The k-Nearest Neighbours in the training set are taken into account for predicting the value or class of a new instance. In terms of classification, it chooses the neighbor's majority class, and in terms of regression, it takes the average of those values. The bias-variance trade-off and complexity of the model are influenced by the choice of k. Since it is non-parametric, k-NN can be applied to a variety of situations, although it is sensitive to irrelevant features and distance metrics.

4.3. Elastic Net

The Elastic Net is a regression technique that combines L1 (Lasso) and L2 (Ridge) regularisation methods to achieve a balance between feature selection and feature grouping. The model introduces two hyperparameters, alpha and l1 ratio, which control the extent of L1 and L2 regularisation applied during training. By adjusting these hyperparameters, the Elastic Net model can effectively handle both feature selection and grouping, resulting in more accurate and interpretable regression models. After training the model on the provided hyperparameters, predictions are made on the test set. The performance of the model is evaluated using mean absolute error (MAE). This metric provides insights into the accuracy and goodness of fit of the Elastic Net model.

4.4. Gradient Boosting

The Gradient Boosting is a powerful machine learning algorithm used for regression tasks. It combines multiple weak predictive models like decision trees in an ensemble to make accurate predictions. The algorithm works by sequentially fitting the models to the residuals of the previous model, allowing it to gradually improve its performance by focusing on the remaining errors. This iterative process effectively captures complex patterns and relationships in the data. The Gradient Boosting Regressor utilises gradient descent optimization to minimise a loss function, such as mean squared error, and find the best fitting model. The algorithm also incorporates regularisation techniques to prevent over-fitting and enhance generalisation.

4.5. Decision Tree

The decision tree algorithm is a powerful supervised learning method that constructs a tree-like model as shown in Figure 1. In this model, internal nodes represent features or attributes, branches represent decision rules, and leaf nodes correspond to predicted values. The algorithm initiates by selecting the best attribute to split the dataset, evaluating various attributes and measuring their impact on reducing the target variable's impurity. This attribute selection process is recursively applied to subsets of data until a predefined stopping criterion is satisfied. Upon constructing the tree, each leaf node is assigned a predicted value based on the average of the target variable. This enables the model to make predictions on new, unseen instances by traversing the tree from the root node to a leaf node, guided by the instance's attribute values.

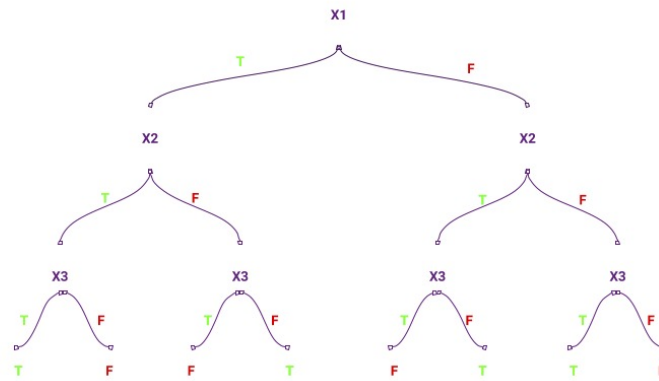


Figure 1: Decision Tree

4.6. Voting Regressor

The Voting Regressor shown in Figure 2 is a machine learning ensemble technique. It combines multiple base models, including ElasticNet, Gradient Boosting Regressor, and Decision Tree Regressor, to make predictions. Each base model contributes to the final prediction by voting or averaging their individual predictions. The Voting Regressor leverages the strengths of each base model, resulting in improved overall prediction accuracy and robustness. Here, the Voting Regressor is trained on the training data and used to predict the target variable on the test set. The performance of the model is evaluated using the Mean Squared Error (MSE).

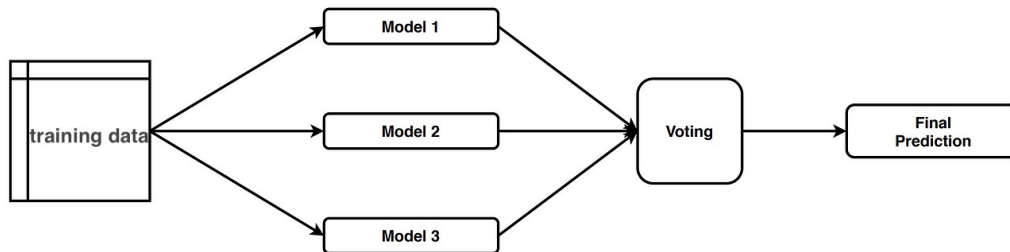


Figure 2: Voting Regressor

4.7. Stack Ensemble Regressor

The Stack Ensemble combines multiple base models, including ElasticNet, GradientBoostingRegressor, and DecisionTreeRegressor, to create a more robust and accurate predictive model. The base models are trained individually on the training data, and their predictions are then used as input features for a meta-model. The meta-model learns to combine the predictions of the base

models to make the final prediction. By leveraging the strengths of different base models, the Stack Ensemble aims to improve the overall predictive performance. The ensemble model is trained on the training data and evaluated on the test data using Mean Squared Error.

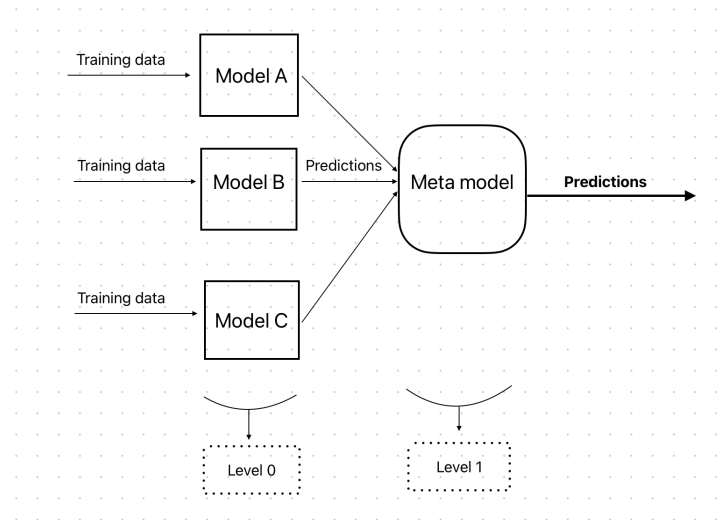


Figure 3: Stack Ensemble

5. Result Analysis of Fusion Technique for Result Diversification Task

This section discusses about the implementation fusion techniques with the analysis of the results using evaluation metrics namely Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE).

5.1. Implementation

The inducers' data contains information about query_id, inter, photo_id, rank, sim and run_name. The rank and sim are extracted from the inducers data. The missing values of the rank and sim attributes are filled using the SimpleImputer method. The data is then split into 80% training set and 20% testing set. Three regression models **M1**, **M2** and **M3** are built to study how the similarity scores are assigned.

The model **M1** is implemented using the *ElasticNet Regressor* where the *alpha parameter* controls the regularization strength. It helps to prevent overfitting by shrinking the coefficients towards zero. The *l1_ratio* parameter determines the balance between the *L1* and *L2* penalties. If *l1_ratio* is 1 it indicates L1 regularization (*Lasso*) and if it 0 it indicates L2 regularization (*Ridge*). If the value is in between 0 and 1, it represents a combination of both the penalties. The model

is trained on the training data using the fit method, which estimates the coefficients that best fit the data.

The model **M2** is implemented using *GradientBoostingRegressor* which calculates the gradients of the loss function with respect to the predictions made by the weak learners. In the code, the gradients are implicitly computed during the training process of the *GradientBoostingRegressor*.

The model **M3** is implemented using *Decision Tree Regressor*. The fit (X,y) method is used to train the *Decision Tree Regressor* on the given training data. The predict(X) method is used to make predictions on new data using the trained *Decision Tree Regressor*.

The model **M4** is implemented using *Voting Regressor*. The *VotingRegressor* model is created by passing the base models **M1**, **M2**, **M3** as estimators to the '*VotingRegressor*' class. The voting method used is 'hard', which means the final prediction is based on the majority vote of the base models.

The ensemble model is created using the *StackingRegressor* from scikit-learn. The estimators are defined as a list of tuples, where each tuple contains the name of the models **M1**, **M2**, and **M3**. The final estimator, which is the *gradient boosting regressor*, is used to build a meta model. An ensemble model **M5** is obtained from the models **M1**, **M2**, and **M3** using the final estimator.

5.2. Results and discussion

The built models **M1**, **M2**, **M3**, **M4**, and **M5** are used to classify the test dataset. The predicted values are compared with the actual values and various evaluation metrics are computed as shown in Table 3 to assess the performance of these models. These metrics provide insights into the model's ability to correctly classify and predict the correct results based on rank and similarity scores. After analysing the results for various evaluation metric in the Table 3 it is clear that **M1** model is the yielding best results among all the model.

Table 3

Assessment of Models using Evaluation Metrics

Model	MAE	MSE	RMSE
ElasticNet(M1)	0.0039	0.0004	0.0070
Gradient Boosting(M2)	0.0314	0.0162	0.1274
Decision Tree(M3)	0.0002	0.0013	0.0360
Voting(M4)	0.0114	0.0019	0.0445
Stacking(M5)	0.0315	0.0165	0.1287

The **M1** model has been tested with the CLEF test data and the F1@20 and CR@20 metrics are used to compare and analyze the performance of the results. F1@20 combines precision and recall into a single score, providing a balanced measure of the system's performance. CR@20 calculates the proportion of relevant items or documents that are retrieved within the top 20 ranked results. A higher CR@20 score indicates a system's ability to retrieve more relevant items within the top-ranked results. An F1@20 of 0.5708 and CR@20 of 0.449 is obtained in

the top 10 results. Table 4 illustrates the F1@20 and CR@20 evaluated for the 10 best file submissions.

Table 4
Test results for Result Diversification

Run	F1@20	CR@20
1	0.4993	0.3901
2	0.5183	0.3992
3	0.5383	0.428
4	0.547	0.4137
5	0.5623	0.439
6	0.5408	0.4295
7	0.5708	0.449
8	0.5364	0.4165
9	0.5404	0.4215
10	0.5343	0.414

6. Result Analysis of Fusion Technique for Image Interestingness Task

This section discusses about the implementation fusion techniques with the analysis of the results using evaluation metrics namely Accuracy, Precision, Recall, F1 score, Mean Absolute Error, Balanced Accuracy.

6.1. Implementation

The inducers' data, contains information about video and image identifiers, classification labels, and interestingness scores. The interestingness scores and classification labels are extracted from the inducers' data. The interestingness scores are stored in a numpy array, while the classification labels are stored in a separate array. The data is then split into 80% training dataset and 20% testing dataset. Three classifier models M1, M2, and M3 were built to study the nature of classification of the images.

The **M1** classifier is implemented using the *XGBoost algorithm* with grid search to find the best combination of hyperparameters such as *maxdepth*, *learningrate*, and *nestimators*. The GridSearchCV function from sklearn is used to perform the grid search, with the F1 score as the evaluation metric.

The **M2** classifier is implemented using the *decision tree algorithm* with grid search to find the optimal combination of hyperparameters such as *maxdepth*, *minsamplesplit*, and *minsamplesleaf*.

The **M3** classifier is implemented using the *K-nearest neighbours algorithm* with grid search to find the optimal combination of hyperparameters such as *nneighbors*, weights, and *p*, representing the number of neighbors, the weight function used in prediction, and the power parameter for the Minkowski distance, respectively.

A Voting Classifier is created with all the models **M1**, **M2**, and **M3**. A grid search is performed to find the optimal combination of the voting scheme and weights. The best *Voting Classifier model (M4)* is obtained based on the grid search results.

The ensemble model is created using the *StackingClassifier* from scikit-learn. The estimators are defined as a list of tuples, where each tuple contains the name of the models **M1**, **M2**, and **M3** and the corresponding best model instance. The final estimator, which is the decision tree classifier, is used to build a meta model. An ensemble model **M5** is obtained from the models **M1**, **M2**, and **M3** using the final estimator.

6.2. Results and discussion

The built models **M1**, **M2**, **M3**, **M4**, and **M5** are used to classify the test dataset. The predicted labels are compared with the actual labels and various evaluation metrics are computed as shown in Table 5 to assess the performance of these models. These metrics provide insights into the model's ability to correctly classify and predict the interestingness of media content. After analyzing the results for various evaluation metrics in the Table 5 it is clear that the **M4** model is the yielding best results among all the models.

Table 5
Assessment of Models using Evaluation Metrics

Model	Accuracy	Precision score	F1 score	Mean per class error	Balanced accuracy score
XGBoost(M1)	0.8705	0.8450	0.6826	0.1295	0.3637
DT (M2)	0.8637	0.8006	0.6763	0.1363	0.3924
kNN(M3)	0.8691	0.8261	0.6849	0.1309	0.3794
Voting(M4)	0.8756	0.8801	0.6887	0.1244	0.3428
Stacking (M5)	0.8461	0.7430	0.6394	0.1539	0.4091

The **M4** model has been tested with the CLEF test data and MAP@10 metric is used to compare and analyze the performance of the results. The Mean Average Precision at 10 ranges from 0 to 1, where a higher value indicates better performance. It considers the order and relevance of the recommended items, giving more weight to relevant items appearing at higher positions in the recommendations. A MAP@10 of 0.1331 is obtained in the top 10 results. Table 6 illustrates the MAP@10 score evaluated for the 10 best file submissions.

Table 6
Test results for Image Interestingness

Run	MAP@10
1	0.0667
2	0.1331
3	0.1252
4	0.1303
5	0.0977
6	0.0666
7	0.0603
8	0.0666
9	0.096
10	0.0969

7. Conclusion

In order to improve the predictions of the results of the inducers in the result diversification task, three base regressors and two ensemble models were implemented. The model was trained on data from 56 different inducers, containing 134,400 training values and tested on data from 56 inducers, containing 33,600 testing values. The base regressors obtained RMSE values of 0.0070, 0.1274 and 0.0360 each. The ensemble models obtained RMSE scores of 0.0445, and 0.1287 each. The best model is chosen based on the RMSE score. The model is then used to predict the values of 56 inducers containing 176,400 values and among the ten best submissions, the best F1 score and CR score are 0.5708 and 0.4295 respectively.

In order to improve the predictions of the results of the inducers in the image interestingness task, three base classifiers and two ensemble models were implemented. The model was trained on data from 29 different inducers, containing 43,546 training values and tested on data from 29 inducers, containing 10,886 testing values. The base classifiers obtained Accuracy of 0.8705, 0.8637 and 0.8691 each. The ensemble models obtained Accuracy of 0.8756, and 0.8461 each. The best model is chosen based on the Accuracy. The model is then used to predict the values of 29 inducers containing 16,182 values and among the ten best submissions, the best MAP@10 score is 0.1331.

8. References

- [1] Bogdan Ionescu, Henning Müller, Ana-Maria Drăgulinescu, Wen-wai Yim, Asma Ben Abacha, Neal Snider, Griffin Adams, Meliha Yetisgen, Johannes Rückert, Alba García Seco de Herrera, Christoph M. Friedrich, Louise Bloch, Raphael Brüngel, Ahmad Idrissi-Yaghir, Henning Schäfer, Steven A. Hicks, Michael A. Riegler, Vajira Thambawita, Andrea Storås, Pål Halvorsen, Nikolaos Papachrysos, Johanna Schöler, Debesh Jha, Alexandra-Georgiana Andrei, Ahmedkhan Radzhabov, Ioan Coman, Vassili Kovalev, Alexandru Stan,

- George Ioannidis, Hugo Manguinhas, Liviu Daniel Ștefan, Mihai Gabriel Constantin, Mihai Dogariu, Jérôme Deshayes, Adrian Popescu, “Overview of the ImageCLEF 2023: Multimedia Retrieval in Medical, Social Media and Recommender Systems Applications,” *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, Proceedings of the 14th International Conference of the CLEF Association (CLEF 2023), Thessaloniki, Greece, September 18-21, 2023.
- [2] Liviu-Daniel Ștefan, Mihai Gabriel Constantin, Mihai Dogariu, Bogdan Ionescu, “Overview of ImageCLEFfusion 2023 Task - Testing Ensembling Methods in Diverse Scenarios,” *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, Proceedings of the 14th International Conference of the CLEF Association (CLEF 2023), Thessaloniki, Greece, September 18-21, 2023.
- [3] Hai-Tao Yu, “Optimize What You Evaluate With: Search Result Diversification Based on Metric Optimization,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 9, pp. 10399–10407, 2022.
- [4] Shreya Sriram, Ramachandran Balasundaram P, L. Kalinathan, “Ensembled Approach for Web Search Result Diversification Using Neural Networks,” *CLEF2022 Working Notes*, CEUR Workshop Proceedings, CEUR-WS.org, Bologna, Italy, 2022.
- [5] L. Kalinathan, P. Balasundaram, Sriram, “A Fusion Approach for Web Search Result Diversification Using Machine Learning Algorithms,” *CLEF2022 Working Notes*, CEUR Workshop Proceedings, CEUR-WS.org, Bologna, Italy, 2022.
- [6] Maria Shoukat, Khubaib Ahmad, Naina Said, Nasir Ahmad, Mohammed Hassanuzaman, Kashif Ahmad, “A Late Fusion Framework with Multiple Optimization Methods for Media Interestingness,” *arXiv preprint arXiv:2207.04762*, 2022.
- [7] Ying Dai, “Building CNN-Based Models for Image Aesthetic Score Prediction Using an Ensemble,” *Journal of Imaging*, vol. 9, no. 2, pp. 2–30, 2023.
- [8] Vaibhav Kalakota, Ajay Bansal, “Diversifying Relevant Search Results from Social Media Using Community Contributed Images,” *IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC)*, 2021.
- [9] Mihai Gabriel Constantin, Liviu-Daniel Ștefan, Bogdan Ionescu, Ngoc QK Duong, Claire-Hélène Demarty, and Mats Sjöberg, “Visual interestingness prediction: A benchmark framework and literature review,” *International Journal of Computer Vision*, 129:1526–1550, 2021.
- [10] Bogdan Ionescu, Mircea-Radu Rohm, Bogdan Boteanu, Adrian L. Gînscă, Mihai Lupu, and Henning Müller, “Benchmarking Image Retrieval Diversification Techniques for Social Media,” *IEEE Transactions on Multimedia*, 23:677–691, 2020.