

LEVERAGING METADATA AND MULTIMODAL INFORMATION IN DEEP EXTREME CLASSIFICATION

ANSHUL MITTAL



AMAR NATH AND SHASHI KHOSLA
SCHOOL OF INFORMATION TECHNOLOGY
INDIAN INSTITUTE OF TECHNOLOGY DELHI
MAY 2024

© Indian Institute of Technology Delhi (IITD), New Delhi, 2024

LEVERAGING METADATA AND MULTIMODAL INFORMATION IN DEEP EXTREME CLASSIFICATION

by

ANSHUL MITTAL

AMAR NATH AND SHASHI KHOSLA
SCHOOL OF INFORMATION TECHNOLOGY

Submitted

in fulfillment of the requirements of the degree of
Doctor of Philosophy

to the



Indian Institute of Technology Delhi
May 2024

Certificate

This is to certify that the thesis titled **LEVERAGING METADATA AND MULTIMODAL INFORMATION IN DEEP EXTREME CLASSIFICATION** being submitted by **Mr. ANSHUL MITTAL** for the award of **Doctor of Philosophy in Extreme Classification** is a record of bonafide work carried out by him under my guidance and supervision at the Amar Nath and Shashi Khosla School of Information Technology, Indian Institute of Technology Delhi. The work presented in this thesis has not been submitted elsewhere, either in part or full, for the award of any other degree or diploma.

Sumeet Agarwal
Associate Professor
Department of Electrical Engineering
Indian Institute of Technology Delhi
New Delhi - 110016

Manik Varma
Distinguished Scientist
& Vice President
Microsoft Research India
Bengaluru - 560001

Acknowledgements

This doctoral journey has been an incredible voyage of intellectual exploration, personal growth, and immense challenges. The completion of this dissertation would not have been possible without the support and guidance of many wonderful individuals.

First and foremost, I owe a deep debt of gratitude to Dr. Manik Varma, Dr. Sumeet Agrwal, and Dr. Purushottam Kar for their unwavering support, insightful guidance, and trust in my abilities have been instrumental in shaping my research and development as a scholar. Their mentorship extends far beyond academia, as they have also offered invaluable personal support throughout my Ph.D. Their belief in me, even when I doubted myself, propelled me forward, especially when I ventured into seemingly risky projects.

My sincere thanks also go to my exceptional colleagues, Kunal Dhahiya and Deepak Saini. Countless late-night discussions, insightful lunch debates, and collaborative brainstorming sessions fueled my research and enriched my experience. I deeply appreciate their intellectual companionship and camaraderie.

I am incredibly grateful to my collaborators, Sheshansh Agrawal, Shreya Malani, Shikhar Mohan, and Naveen Sachdeva. Collaborating with these brilliant minds has been a rewarding experience, pushing my research boundaries and fostering meaningful intellectual exchange.

Words cannot express my heartfelt gratitude to Prateek Chhabra, Vyom Vashishta, Shruti Garg, and Ritika Singh. Their unwavering friendship, emotional support, and unwavering encouragement were like a beacon of hope during my darkest hours. They kept me grounded, motivated, and helped me navigate the emotional roller coaster of this journey.

Lastly but most importantly, my deepest love and gratitude go to my family: my sister, Dr. Megha, and my mother, Rajni, who have stood as pillars of strength throughout my life. I am forever indebted to the memory of my late father, Lect. Vinay Mittal, whose shield of protection, affection, and unwavering trust continue to inspire resilience in the face of adversity. His spirit motivates me to continue contributing to the world of knowledge.

To all of you, thank you from the bottom of my heart. Your presence has made this incredible journey possible.

Anshul Mittal

Abstract

Deep Extreme Classification (XC) refers to accurately tagging a data point with a relevant subset of labels from a large universe of labels. The challenge is to predict data scarce tail labels. Additionally, tagging every data point comprehensively becomes impractical due to the sheer number of labels, leading to potential omissions or “missing labels”. Metadata about labels, including descriptions, image features, correlations, and relationships, offers valuable insights into document-label connections and holds promise for addressing these challenges. This thesis introduces four algorithms – DECAF, MUFIN, ECLARE, and RAMEN – that leverage the metadata to improve both accuracy and diverse recommendations.

Traditional XC methods like MACH and Parabel struggle with label semantics. However, DECAF tackles this challenge by understanding label meaning through the associated text. By capturing semantic relationships between labels, DECAF delivers accurate recommendations, outperforming baselines by 2-3% in accuracy.

Real-world product recommendations often involve both text and visual data. While vision-based methods like ADDE-O and CLIP excel at handling visuals, they face

difficulties in scaling. MUFIN bridges the gap between existing multi-modal approaches and XC through its scalable training and architecture, surpassing XC and vision-based methods by 4-10% in accuracy.

Leveraging multi-modality enhances accuracy for tail labels, but gains are limited due to significant data imbalance. Methods like X-Transformer and AttentionXML, despite utilizing label semantics, struggle with recommending tail labels. ECLARE harnesses textual descriptions and inferred label correlation graphs to "borrow strength" from more common head labels, allowing for accurate predictions of tail labels, outperforming baseline XC approaches by 4-5% in accuracy.

These label correlations are often incomplete due to missing labels. Graph metadata, such as connections between related products or webpages, can help identify the missing labels. Traditional methods like GraphFormers and PINA rely on computationally expensive graph convolution networks (GCNs) to utilize such data. RAMEN introduces a cost-effective alternative to GCNs, leveraging graph metadata to infer missing labels and subsequently enhancing performance by 3-5% in accuracy.

Collectively, these algorithms demonstrate the power of metadata. They pave the way for accurate and personalized recommendations, improved content tagging, and richer user experiences across various applications. Code for each method is available at the XML Repository [4].

सारांश

दीप एक्सटीम वर्गीकरण (एक्ससी) बड़ी संख्या में लेबलों में से प्रासंगिक उपसमूह के साथ डेटा बिंदु को सटीक रूप से टैग करने को संदर्भित करता है। चुनौती यह है कि डेटा की कमी वाले टेल लेबल की भविष्यवाणी की जाए। इसके अतिरिक्त, लेबलों की भारी संख्या के कारण प्रत्येक डेटा बिंदु को व्यापक रूप से टैग करना अव्यावहारिक हो जाता है, जिससे संभावित चूक या "लापता लेबल" हो जाते हैं। लेबल विवरण, छवि सुविधाओं, सहसंबंधों और संबंधों सहित लेबल के बारे में मेटाडेटा दस्तावेज़-लेबल कनेक्शन में मूल्यवान अंतर्दृष्टि प्रदान करता है और इन चुनौतियों का समाधान करने का वादा करता है। यह थीसिस चार एल्गोरिदम - DECAF, MUFIN, ECLARE और RAMEN - का परिचय देता है जो सटीकता और विविध अनुशंसाओं दोनों को बेहतर बनाने के लिए मेटाडेटा का लाभ उठाते हैं।

MACH और Parabel जैसी पारंपरिक XC विधियां लेबल अर्थशास्त्र के साथ संघर्ष करती हैं। हालांकि, DECAF संबंधित टेक्स्ट के माध्यम से लेबल अर्थ को समझकर इस चुनौती का सामना करता है। लेबल के बीच अर्थ संबंधों को कैप्चर करके, DECAF सटीक अनुशंसाएं प्रदान करता है, जो सटीकता में आधारभूत प्रदर्शन से 2-3% बेहतर प्रदर्शन करता है।

वास्तविक दुनिया में उत्पाद अनुशंसाओं में अक्सर पाठ और दृश्य दोनों डेटा शामिल होते हैं। जबकि दृष्टि-आधारित तरीके जैसे ADDE-O और CLIP दृश्यों को संभालने में उत्कृष्ट हैं, उन्हें स्केलिंग में कठिनाइयों का सामना करना पड़ता है। MUFIN अपने स्केलेबल प्रशिक्षण और आर्किटेक्चर के माध्यम से मौजूदा मल्टी-मोडल दृष्टिकोणों और XC के बीच के अंतर को पाटता है, जो सटीकता में XC और विज़न-आधारित विधियों को 4-10% से अधिक पार कर जाता है।

मल्टी-मोडैलिटी का लाभ उठाना टेल लेबल के लिए सटीकता को बढ़ाता है, लेकिन महत्वपूर्ण डेटा असंतुलन के कारण लाभ सीमित हैं। X- ट्रांसफॉर्मर और अटेंशनएक्सएमएल जैसी विधियां, लेबल अर्थशास्त्र का उपयोग करने के बावजूद, टेल लेबल की सिफारिश करने में संघर्ष करती हैं। ECLARE पाठ विवरणों और अनुमानित लेबल सहसंबंध ग्राफ़ का उपयोग करता है ताकि अधिक सामान्य हेड लेबल से "शक्ति उधार" ले सके, जिससे टेल लेबल की सटीक भविष्यवाणी की जा सके, आधारभूत XC दृष्टिकोणों को सटीकता में 4-5% से अधिक आउटपरफॉर्म किया जा सके।

ये लेबल सहसंबंध अक्सर लापता लेबलों के कारण अधूरे होते हैं। ग्राफ़ मेटाडेटा, जैसे संबंधित उत्पादों या वेबपृष्ठों के बीच संबंध, लापता लेबलों की पहचान करने में मदद कर सकता है। ग्राफफॉर्मर्स और पीआईएनए जैसी पारंपरिक विधियां ऐसे डेटा का

उपयोग करने के लिए कम्प्यूटेशनल रूप से महंगे ग्राफ कनवल्शन नेटवर्क (जीसीएन) पर भरोसा करती हैं। RAMEN GCN के लिए एक लागत- प्रभावी विकल्प प्रस्तुत करता है, जो लापता लेबल का अनुमान लगाने और बाद में सटीकता में 3-5% तक प्रदर्शन को बढ़ाने के लिए ग्राफ मेटाडेटा का लाभ उठाता है।

सामूहिक रूप से, ये एल्गोरिदम मेटाडेटा की शक्ति का प्रदर्शन करते हैं। वे विभिन्न अनुप्रयोगों में सटीक और व्यक्तिगत अनुशंसाओं, बेहतर सामग्री टैगिंग और समृद्ध उपयोगकर्ता अनुभवों का मार्ग प्रशस्त करते हैं। प्रत्येक विधि के लिए कोड XML रिपॉजिटरी में उपलब्ध है।

Contents

Certificate	i
Acknowledgements	iii
Abstract	v
List of Figures	xiii
List of Tables	xxv
1 Introduction	1
1.1 Overview	1
1.2 Importance of metadata	2
2 Related Work	7

2.1	Type of classifiers	7
2.2	Type of metadata	9
3	Dataset and Evaluation	11
3.1	Dataset Preparation	11
3.1.1	Cleaning and processing datasets	15
3.2	XC evaluation metrics	15
4	DECAF: Deep Extreme Classification with Label Features	17
4.1	Intuition and Notation	17
4.2	Architecture and Training	21
4.2.1	Architecture	21
4.2.2	Training	23
4.2.3	Prediction	26
4.3	Results and Discussion	26
4.4	Analysis and Ablation	32
4.5	Conclusion	37

5	MUFIN: Multi-modal Extreme Classification	39
5.1	Intuition and Notations	39
5.2	Architecture and Training	41
5.2.1	Architecture	41
5.2.2	Training	43
5.2.3	Prediction	45
5.3	Results and Discussion	46
5.4	Analysis and Ablation	48
5.5	Conclusion	56
6	ECLARE: Extreme Classification with Label Graph Correlations	59
6.1	Intuition and Notation	59
6.2	Architecture and Training	60
6.2.1	Architecture	60
6.2.2	Training	70
6.2.3	Prediction	72
6.3	Results and Discussion	73

6.4	Analysis and Ablation	78
6.5	Conclusion	84
7	RAMEN: Graph Regularized Encoder Training for Extreme Clas- sification	85
7.1	Intuition and Notation	85
7.2	Architecture and Training	87
7.2.1	Architecture	88
7.2.2	Training	89
7.2.3	Prediction	92
7.3	Results and Discussion	93
7.3.1	Analysis and Ablation	96
7.4	Conclusion	100
8	Conclusion and Future Work	103
	Bibliography	105
A	Hyperparameters	115

A.1	DECAF	115
A.2	MUFIN	115
A.3	ECLARE	117
A.4	RAMEN	117
B	Additional Results	119
B.1	DECAF	119
B.1.1	Further Details about Experiments and Ablation Studies . . .	119
B.2	MUFIN	125
B.3	Label Quantile Creation	125
C	Complexity Analysis	129
C.1	DECAF	129
C.2	MUFIN	131
D	Theoretical Analysis	133
D.1	DECAF	133
D.1.1	Proof of Theorem 1	133

D.2	RAMEN	137
D.2.1	Proof for a Single-layer GCN	138
D.2.2	Extension to GCNs with Multiple Layers	140
List of Publications		141

List of Figures

1.1	The MM-AmazonTitles-300K recommendation task highlights the importance of accurate multi-modal retrieval. When using a decorative motorcycle-shaped alarm clock as the query, MUFIN successfully retrieved visually similar items like a motorcycle-shaped pencil holder and thematically related products like a motorcycle-themed ashtray. However, relying solely on visual cues ignored thematically linked items, while text-based retrieval primarily focused on "motorcycle," leading to accessories for real motorcycles.	4
-----	---	---

1.2	A snapshot from the LF-WikiSeeAlsoTitles-320K dataset for the article on “Cladistics”, a method of biological classification that groups organisms based on common ancestry, related article “Common descent” is tagged but the ground truth is incomplete and has missing labels such as “Crown group” which refers to hierarchy of organisms. Traversal on hyperlink edges can help discover missing labels but must be used with caution since they can also lead to irrelevant labels such as “Vestigial organs”.	5
3.1	Number of training points per label for two datasets. Labels are ordered from left to right in increasing order of popularity. 60–80% labels have < 5 training points (the bold horizontal line indicates the 5 training point level).	12
4.1	DECAF’s frugal prediction pipeline scales to millions of labels. Given a document $\mathbf{x}$, its text embedding $\hat{\mathbf{x}}$ (see Fig 4.3 (Left)) is first used by the shortlister $\mathcal{S}$ to shortlist the most probable $\mathcal{O}(\log L)$ labels while maintaining high recall. The ranker $\mathcal{R}$ then uses label classifiers (see Fig 4.3 (Right)) of only the shortlisted labels to produce the final ranking.	18

- 4.2 (Left) DECAF uses a lightweight architecture with a residual layer to embed both document and label text (see Fig. 4.3). (Right) Combination blocks are used to combine various representations (separate instances are used in the text embedding blocks ($\mathcal{E}_D, \mathcal{E}_L$) and in label classifiers ($\mathcal{C}_L$)). 19
- 4.3 (Left) Document text is embedded using an instance $\mathcal{E}_D$ of the text embedding block (see Fig. 4.2). Stop words (e.g. *and*, *the*) are discarded. (Right) DECAF critically incorporates label text into classifier learning. For each label $l \in [L]$, a one-vs-all classifier $\mathbf{w}_l$ is learnt by combining label text embedding $\mathbf{z}_l^1$ (using a separate instance $\mathcal{E}_L$ of the text embedding block) and a refinement vector $\mathbf{z}_l^2$. Note that $\mathcal{E}_D, \mathcal{E}_L, \mathcal{C}_L$ use separate parameters. However, all labels share the blocks $\mathcal{E}_L, \mathcal{C}_L$ and all documents share the block $\mathcal{E}_D$ 20
- 4.4 Quantile analysis of gains offered by DECAF in terms of contribution to P@5. The label set was divided into 5 bins with increasing label frequency (see Appendix B.3 for binning details). Quantiles increase in mean label frequency from left to right. DECAF consistently outperforms other methods on all bins with the difference in accuracy being more prominent on bins containing data-scarce tail labels (e.g. bin 5). 31

-
- 4.5 Impact of the number of instances in DECAF’s ensemble on performance on the LF-AmazonTitles-131K dataset. DECAF offers maximum benefits using a small ensemble of 3 instances after which benefits taper off. 35
- 4.6 A comparison of recall when using moderate or large fanout on the LF-WikiSeeAlso-320K dataset. The x-axis represents various values of beam-width B and training recall offered by each. A large fanout offers superior recall with small beam width, and hence small shortlists lengths. 35
- 5.1 **(a)** The embedding block $\mathcal{E}$ uses a multi-modal self-attention block $\mathcal{A}_S$ to represent a datapoint (or a label) as a bag of embeddings $\hat{\mathbf{X}}^1$ that can be optionally aggregated into a single normalized vector $\hat{\mathbf{x}}^1$. **(b)** One-vs-all classifier vectors $\mathbf{w}_l$ are learnt for each label by combining the vector representation for that label $\hat{\mathbf{z}}_l^1$ with a normalized free vector $\mathfrak{N}(\boldsymbol{\eta}_l)$ ($\mathfrak{N}$ is normalization). **(c)** The relevance score between a datapoint i and label l is computed using the label classifier $\mathbf{w}_l$ and a vector representation of the datapoint $\hat{\mathbf{x}}_i^{2,l}$ that is adapted to the label l by using the cross-attention block $\mathcal{A}_C$. **(d)** The attention block is instantiated twice, once to implement self-attention as $\mathcal{A}_S : X \mapsto \mathcal{A}(X, X)$ and once to implement cross-attention as $\mathcal{A}_C : (X, Z) \mapsto \mathcal{A}(X, Z)$. The two instantiations use distinct parameters. 40

- 5.2 t-SNE representations for classifiers $\mathbf{w}_l, l \in [L]$ learnt by MUFIN show that labels belonging to the same category are clustered together. Only the 5 most popular categories are shown. 52
- 5.3 MUFIN outperforms baseline methods on almost all categories. Only the top 5 categories are shown here to avoid clutter. Tab. B.2 contains results on all categories. 52
- 5.4 Analyzing the performance of MUFIN and other methods on popular vs. rare labels. Labels were divided into 5 bins in increasing order of popularity (left-to-right)(see Appendix B.3 for binning details). The plots show the overall R@10 of each method (histogram group “complete”) and the contribution of each bin to this value. The results indicate that MUFIN’s performance on popular labels (histogram group 1) does not come at the cost of performance in rare labels. Other methods seem to exhibit a trade-off between rare and popular labels. 53

- 5.6 The impact of multi-modal cross-attention in MUFIN. 5.5a shows the cross-attention heat map between the datapoint X and a relevant label Z_+ . 5.5b shows that cross-attention is able to identify objects in X among objects in the background of the relevant label Z_+ . 5.5c shows how this helps boost the scores assigned to relevant labels. $\hat{\mathbf{x}}^1$ ($\bullet$) represents the non-adapted vector embedding of the datapoint X . $\mathbf{w}_+(\blacktriangle)$ and $\mathbf{w}_-(\blacktriangle)$ represent the classifier vectors for the relevant and irrelevant labels Z_+ and Z_- respectively. Similarly, $\hat{\mathbf{x}}^{2,+}(\star)$ and $\hat{\mathbf{x}}^{2,-}(\star)$ represent the vector embedding of the datapoint adapted to Z_+ and Z_- respectively. Notice how adaptation moves $\hat{\mathbf{x}}^{2,+}(\star)$ closer to $\mathbf{w}_+(\blacktriangle)$ allowing the relevant label Z_+ to get a higher score. On the other hand, adaptation has no effect when done with respect to an irrelevant label (note that $\hat{\mathbf{x}}^1(\bullet)$ and $\hat{\mathbf{x}}^{2,-}(\star)$ do indeed almost overlap). 5.5c was plotted by projecting the vectors $\hat{\mathbf{x}}^1, \mathbf{w}_+, \mathbf{w}_-, \hat{\mathbf{x}}^{2,+}, \hat{\mathbf{x}}^{2,-}$ onto $\mathbb{R}^2$ using a t-SNE embedding. 54
- 6.1 Prediction Pipeline: ECLARE uses a low-cost prediction pipeline that can make millisecond-level predictions with millions of labels. Given a document/query $\mathbf{x}$, its text embedding $\hat{\mathbf{x}}$ (see Fig 6.2) is used by the shortlister $\mathcal{S}$ to obtain the $\mathcal{O}(\log L)$ most probable labels while maintaining high recall using label correlation graphs via GAME. Label classifiers (see Fig 6.2) for only the shortlisted labels are then used by the ranker $\mathcal{R}$ to produce the final ranking of labels. 60

- 6.2 (Top) Document Embedding: ECLARE uses the light-weight embedding block $\mathcal{E}$ to embed documents, ensuring rapid processing at test time. Stop words such as *for*, *of*) are discarded. (Bottom) Label classifiers: ECLARE incorporates multiple forms of label metadata including label text (LTE) and label correlation graphs (GALE), fusing them with a per-label refinement vector $\hat{\mathbf{z}}_l^3$ using the attention block to create a one-vs-all classifier $\mathbf{w}_l$ for each label $l \in [L]$. Connections to and from the attention block are shown in light gray to avoid clutter. A separate instance of the embedding block is used to obtain document embeddings ($\mathcal{E}_D$), LTE ($\mathcal{E}_L$) and GALE ($\mathcal{E}_G$) embeddings. The embedding block is used in document and label embeddings. The attention block is used to fuse multiple label representations into a single label classifier. 61
- 6.3 An execution of Algorithm 1 on a subset of the LF-WikiSeeAlsoTitles-320K dataset starting from the label “Kangal Dog”. (Left) ECLARE uses document-label associations taken from ground-truth label vectors (black color) to infer indirect correlations among labels. (Middle) Random walks (distinct restarts colored differently) infer diverse correlations. (Right) These inferred correlations (marked with the color of the restart that discovered them) augment the meager label co-occurrences present in the ground truth (marked with dotted lines). . 64

-
- 6.4 A comparison of meta-label clusters created by ECLARE compared to those created by DECAF. Note that the clusters for DECAF are rather amorphous, combining labels with diverse intents whereas those for ECLARE are much more focused. We note that other methods such as ASTEC and AttentionXML offered similarly noisy clusters. It is clear that clustering based on label centroids that are augmented using label correlation information creates far superior clusters that do not contain noisy and irrelevant labels. 65
- 6.5 To avoid head labels from distorting label correlation patterns, labels with > 500 training points are forcibly disconnected from the label correlation graph and also clustered into *head* meta-labels separately. Consequently, the **GALE** and **GAME** steps have a trivial action on head labels. 66

- 6.6 Analyzing the performance of ECLARE and other methods on popular vs rare labels. Labels were divided into 5 bins in increasing order of popularity. The plots show the overall P@5 for each method (histogram group “complete”) and how much each bin contributed to this value. ECLARE clearly draws much of its P@5 performance from the bin with the rarest labels (histogram group 5), i.e ECLARE’s superiority over other methods in terms of P@5 comes from predicting challenging rare labels and not easy-to-predict popular labels. ECLARE’s lead over other methods is also more prominent for rare bins (histogram groups 4, 5) (see Appendix B.3 for binning details). 77
- 6.7 Analysing the dominance of various components in the label classifier $\mathbf{w}_l$ for rare vs popular labels. For rare labels (on the left), components that focus on label metadata i.e. $\hat{\mathbf{z}}_l^1, \hat{\mathbf{z}}_l^2$ gain significance whereas for popular labels (on the right), the unrestricted refinement vector $\hat{\mathbf{z}}_l^3$ becomes dominant. This illustrates the importance of graph augmentation for data-scarce labels for which the refinement vectors cannot be trained adequately owing to lack of training data. 81

-
- 7.1 The model architecture adopted by RAMEN. A shared encoder is used to embed both data points and labels (a). A cross-attention architecture (b) is used to obtain a label-adapted representation of the data point, which is then scored by a classifier vector. This score is combined with the base label-datapoint similarity score to yield the final score. 86
- 7.2 Quantile wise-comparison of RAMEN and other methods. RAMEN gives consistent gains in each bin (see Appendix B.3 for binning details). The left-most bin contains the tail labels whereas the rightmost bin contains the head labels. 98
- 7.3 t-SNE of retrieved labels of document “terrestrial locomotion” and “Cladistics” by RAMEN, NGAME, GraphFormers. RAMEN has higher number of correct retrievals and retrieves a healthy mix of both head (H) and tail (T) labels, while other methods retrieve fewer relevant labels and focus on only tail labels. 98

B.1	Predictions by MUFIN for sample test points in the MM-AmazonTitles-300K dataset. $\mathbf{Z}_1$ (resp. $\mathbf{Z}_{-1}$) in the title indicates that the retrieved product (label) was relevant (resp. irrelevant) for the query datapoint in the ground truth. Not only does MUFIN mostly recommend products relevant according to the ground truth, but the occasional recommendation not in the ground truth list is often a good recommendation nevertheless. For instance, in the first row, the last recommendation does not appear in the ground truth but is a chair cover relevant to the query. (Figure best viewed under magnification)	126
C.1	Scalable prediction pipeline deployed by MUFIN	132

List of Tables

3.1	Dataset statistics summary for benchmark datasets. For Polyvore-Disjoint, a ‘-’ indicates that only unseen labels were available for recommendation at test time.	12
3.2	Dataset statistics of meta-data graph using in bench marking	13
3.3	Dataset Statistics for methods using BoW features.	13
3.4	Image statistics for datasets used in benchmarking.	13
4.1	A comparison of DECAF on publicly available short-text datasets. . .	27
4.2	A comparison of DECAF’s performance on full text datasets.	28

4.3	DECAF’s predictions on selected test points. Document and label names ending in “...” were abbreviated due to lack of space. Please refer to Table B.1 in the for the complete table. Predictions in black and a non-bold/non-italic font were a part of the ground truth. Those in bold italics were part of the ground truth but never seen with other the ground truth labels in the training set i.e. had no common training points. Predictions in light gray were not a part of the ground truth. DECAF’s exploits label metadata to discover semantically correlated labels.	32
4.4	Augmenting existing BoW-based XML methods by incorporating label metadata leads to 1.5% increase in the accuracy as compared to base method. However, DECAF could be up to 7% more accurate compared to even these.	33
4.5	Using strategies used by existing XML algorithms for shortlisting labels instead of $\mathcal{S}$ hurts both both shortlist recall ($R@20$) and final prediction accuracy ($P@k$, $PSP@k$).	34
4.6	Analyzing the impact for alternative design and algorithmic choices for DECAF’s components.	34

5.1	Results on MM-AmazonTitles-300K. MUFIN outperforms state-of-the-art XC methods by 3–11% in P@1 as well as R@10. MUFIN also outperforms state-of-the-art vision+language pre-training strategies by 3–12% in P@1 and 4–13% in R@10. Recall that the multi-modal techniques were augmented with MUFIN’s pipeline. B.2 shows that MUFIN’s lead rises significantly if the methods are not offered these augmentations. The column t_{pred} reports the per-datapoint prediction times in milliseconds for various methods. MUFIN offers millisecond level prediction comparable to or better than existing methods. . . .	49
5.2	Results on the FITB task on Polyvore-Disjoint. MUFIN is 3-4% more accurate compared to the next best method.	49
5.3	An ablation study exploring alternate architecture and training choices for MUFIN. Choices made by MUFIN could lead to 3-4% gain in P@1 and 1-2% gain in R@10 than alternatives.	50
6.1	Results on public benchmark datasets.	74
6.2	An ablation study showing loss of mutual information (lower is better) using various clustering strategies as well as fanouts. Lowering the number of metalabels $K = \mathcal{C} $ hurts performance. Competing methods that do not use graph-augmented clustering offer poor LMI, especially MACH that uses random hashes to cluster labels.	75

6.3	An ablation study exploring the benefits of the GAME step for other XC methods. Although ECLARE still provides the leading accuracies, existing methods show consistent gains from the use of the GAME step.	75
6.4	An ablation study exploring alternate design decisions. Design choices made by ECLARE for its components were found to be optimal among popular alternatives.	79

- 6.5 A subjective comparison of the top 5 label predictions by ECLARE and other algorithms on the WikiSeeAlso-350K. Predictions typeset in black color were a part of the ground truth whereas those in light gray color were not. ECLARE is able to offer precise recommendations for extremely rare labels missed by other methods. For instance, the label “Dog of Osu” in the first example is so rare that it occurred only twice in the training set. This label does not have any token overlaps with its document or co-occurring labels either. This may have caused techniques such as DECAF that rely solely on label text, to miss such predictions. The examples also establish that incorporating label co-occurrence allows ECLARE to infer the correct intent of a document or a user query. For instance, in the second example, all other methods, including DECAF, either incorrectly focus on the tokens “Academy Awards” in the document title and start predicting labels related to other editions of the Academy Awards, or else amorphous labels about entertainment awards in general. On the other hand, ECLARE is able to correctly predict other labels corresponding to award ceremonies held in the same year as the 85th Academy awards, as well as the rare label “List of ... Best Foreign Language Film” 82

7.1	Results on short-text benchmark datasets. RAMEN-M1 indicates accuracies obtained after module M1 i.e. without training the classifier and cross-attention architecture to offer a fair comparison to other methods that do not adopt a classifier.	94
7.2	Results on full-text benchmark datasets. RAMEN is up to 15% more accurate as compared to both text-based and graph-based baselines. RAMEN-M1 indicates accuracies obtained after module M1 i.e. without training the classifier and cross-attention architecture to offer a fair comparison to other methods that do not adopt a classifier. Note that including the classifier and cross-attention architecture offers up to 6% and 4% improvement in PSP@1 and P@1 respectively.	95
7.3	A subjective comparison of predictions made by RAMEN, the leading text-based method NGAME, and the leading graph-based method GraphFormers on LF-WikiSeeAlsoTitles-320K. Labels that are a part of the ground truth are formatted in black color. Labels not a part of the ground truth are formatted in light gray color. Relevant labels that are missing from the ground truth are marked in bold black. RAMEN could make predict highly relevant labels such as “Crown group”, which were missing from the ground truth as well as omitted by other methods.	97

7.4	Comparing the difference in false negative rates between RAMEN and NGAME on LF-WikiSeeAlsoTitles-320K. Results are presented in a quantile-wise manner. The #5K label quantile contains the most popular/head labels whereas the #729K quantile contains the most rare/tail labels. See Appendix 3 for definitions of metrics and quantiles. RAMEN’s false negative rate is always superior to that of NGAME but the gap widens significantly when compared on tail labels.	97
7.5	Ablations were done on module M1 (RAMEN-M1) on LF-WikiSeeAlsoTitles-320K to understand the impact of design choices on the quality of encoder training. RAMEN’s design choices are seen to be optimal and offer 2.5–13% improvement in the P@1 metric over alternate design choices.	100
7.6	Results of RAMEN as a regularizer in baseline algorithms on LF-WikiSeeAlsoTitles-320K. RAMEN’s regularization algorithm improves the performance of the base algorithm by 1–2%	101
7.7	Impact of metadata on the performance of RAMEN on LF-WikiSeeAlsoTitles-320K. RAMEN’s performance on P and PSP decreases as we decrease the volume of metadata.	101
7.8	RAMEN’s encoder $\mathcal{E}$ can predict links in the meta-data graph with extremely high recall (R@100).	101

- A.1 Parameter settings for DECAF on different datasets. Apart from the hyperparameters mentioned in the table below, all other hyperparameters were held constant across datasets. All ReLU layers were followed by a dropout layer with 50% drop-rate in Module-I and 20% for the rest of the modules. Learning rate was decayed by a decay factor of 0.5 after interval $0.5 \times$ epoch length. Batch size was taken to be 255 for all datasets. Module I used 20 epochs with initial learning rate of 0.01. In Module II, 10 epochs were used with an initial learning rate of 0.008 for all datasets. 116
- A.2 Hyper-parameters used to train MUFIN on the public datasets. MUFIN uses the AdamW optimizer and learning rate scheduler with a warm start of 1000 iterations. 116
- A.3 Parameter settings for ECLARE on different datasets. Apart from the hyperparameters mentioned in the table below, all other hyperparameters were held constant across datasets. All ReLU layers were followed by a dropout layer with 50% drop-rate in Module-I and 20% for the rest of the modules. Learning rate was decayed by a decay factor of 0.5 after interval $0.5 \times$ epoch length. Batch size was taken to be 255 for all datasets. Module I used 20 epochs with initial learning rate of 0.01. In Module II, 10 epochs were used with an initial learning rate of 0.008 for all datasets. ECLARE uses Random walk of length 400 and restart probability of 80%. 117

A.4	Hyper-parameter values for RAMEN on all datasets to enable reproducibility. RAMEN code will be released publicly. Most hyperparameters were set to their default values across all datasets. LR is learning rate. Multiple clusters were chosen to form a batch hence $B > C$. Clusters were refreshed after 5 epochs. Cluster size C was doubled after every 25 epochs. Margin $\gamma = 0.3$ was used for contrastive loss. For training M2 number of positive samples and negative samples were kept at 2 and 12 respectively. A cell containing the symbol $\uparrow$ indicates that that cell contains the same hyperparameter value present in the cell directly above it.	118
-----	---	-----

B.1	A subjective comparison of DECAF’s prediction quality as compared to state-of-the-art deep learning, as well as BoW approaches on examples taken from the test sets of two datasets. Predictions in black color in non-bold/non-italic font were a part of the ground truth. Predictions in bold italics were such that their co-occurrence with other ground truth labels in the test set was novel, i.e. those co-occurences were never seen in the training set. Predictions in light gray color were not a part of the ground truth. DECAF offers much more precise recommendations on these examples as compared to other methods, for example AttentionXML, whose predictions on the last example are mostly irrelevant, e.g. focusing on labels such as “Early United States commemorative coins”, instead of those related to the New Zealand dollar. DECAF is able to predict labels that never co-occur in the training set due to its inclusion of label text in to the classifier. For example, in the last example, the label “Australian dollar” never occurred with the other ground truth labels in the training set i.e. had no common training instances with the rest of the ground truth labels. Similarly, in the first example, the label “Panzer Dragoon Orta” never occurred together with other ground truth labels yet DECAF predicted these labels correctly while the other XML algorithms could not do so.	121
-----	---	-----

B.2	MUFIN’s performance on various categories of the MM-AmaonTitles-300K dataset of which a snapshot was provided in Fig. 5.3. For each category, the best performance is highlighted in bold black font, the second-best performance is left in normal black font and the third-/fourth- performances are stylized in light gray. Note that for most categories, MUFIN is indeed the best method and MUFIN ($\alpha = 1$) is the second-best method. In particular, MUFIN could give accuracy gains up to 6% on various categories (e.g. Musical Instruments). . . .	127
-----	---	-----

