Large-scale Non-linear Multimodal Semantic Embedding

Jorge Andrés Vanegas Ramírez

Universidad Nacional de Colombia
Departamento de Ingeniería de Sistemas e Industrial
Bogotá, Colombia
2017
Large-scale Non-linear Multimodal Semantic Embedding

Jorge Andrés Vanegas Ramírez

Submitted to the Engineering School of the Universidad Nacional de Colombia, in partial fulfillment of the requirements for the degree of:

Doctor of Engineering
Systems and Computer Engineering

Advisor:
Fabio A. Gonzalez Ph.D.

Co-Advisor:
Hugo Jair Escalante Ph.D.

Research area:
Computer Science

Research Group:
MindLab Research Group

Universidad Nacional de Colombia
Departamento de Ingeniería de Sistemas e Industrial
Bogotá, Colombia
2017
Dedication

To my parents Gustavo and Belén
Acknowledgements

This thesis would not have been possible without the help and support of many people. First and foremost, I would like to thank my thesis advisors, Dr. Fabio Gonzalez and Dr. Hugo Escalate for their invaluable guidance, constant encouragement, understanding, and patience. Also, I would like to express my sincere gratitude to all the people who are part of the research group MindLab, who have become not only colleagues but also good friends. Special thanks to my colleagues Viviana Beltran, Victor Contreras and John Arevalo who contributed to my research in a variety of ways.

I greatly appreciate the support received through the collaborative work undertaken with the Laboratory of Language Technologies of the Computational Sciences Department at the National Institute of Astrophysics, Optics, and Electronics (INAOE).

I also want to thank COLCIENCIAS, the institution that mainly funded this thesis by the doctoral fellowship grant 617/2013. Also thanks to Universidad Nacional de Colombia, their professors, infrastructure and funding support for the doctoral thesis project and international mobility.

Finally, but most important of all, I am grateful to my parents, Gustavo Vanegas and Belén Ramírez as well as my brothers, who have always given me their unconditional support, encouragement, and advice.
Abstract

The main goal of this thesis is to investigate effective and efficient methods to combine complementary evidence, and model the relationships between multiple modalities of multimedia data in order to improve the access and analysis of the information, to finally obtain valuable insights about the data. In this thesis is proposed to use multimodal latent semantic as the strategy that allows us to combine and to exploit the different views from this heterogeneous source of knowledge, by modeling relations between the different modalities and finding a new common low-dimensional semantic representation space. For a richer modeling, it is proposed the usage of kernel-based methods that usually present accurate and robust results. Unfortunately, kernel-based methods present a high computational complexity that makes them infeasible for large data collections. This drawback implies one of the most important challenges addressed in this thesis, which was to investigate alternatives to handle large-scale datasets with modest computational architectures.

In this thesis, several kernelized semantic embedding methods based on matrix factorization have been proposed, developed and evaluated. Thanks to the non-linear capabilities of the kernel representations, the proposed methods can model the complex relationships between the different modalities, allowing to construct a richer multimodal representation even when one of the modalities presents incomplete data. Besides, the proposed methods have been designed under a scalable architecture based on two main strategies: online learning and learning-in-a-budget that allow preserving low computational requirements in terms of memory usage and processing time.

An extended experimental evaluation shows that the proposed multimodal strategies achieve the state-of-the-art in several data analysis tasks, such as multi-labeling and multi-class classification and cross-modal retrieval and under different learning setups, such as supervised, semi-supervised, and transductive learning. Furthermore, thanks to the online learning and learning-in-a-budget strategies proposed in this thesis, the scalability capabilities are preserved allowing to deal with large-scale multimodal collections.

Keywords:
Multi-modal information, Multimodal Data Analysis, Machine Learning, latent semantic embedding, kernel methods, large-scale datasets.
Resumen

El objetivo principal de esta tesis es investigar métodos eficaces y eficientes para combinar evidencia complementaria de múltiples modalidades de información multimedia y modelar las relaciones entre éstas, con el fin de mejorar el acceso y el análisis de los datos contenidos. En esta tesis se pretende utilizar la estrategia de semántica latente multimodal, la cual permite combinar y explotar las diferentes vistas de esta fuente de información heterogénea, modelando las relaciones entre las diferentes modalidades y encontrando un nuevo espacio común de representación semántica de baja dimensionalidad. Para un modelado más rico, se propone el uso de métodos basados en kernel los cuales usualmente presentan resultados precisos y robustos. Desafortunadamente, los métodos basados en kernel presentan una alta complejidad computacional que los hace inviables para grandes colecciones de datos. Este inconveniente implica uno de los desafíos más importantes abordados en esta tesis, que fue investigar alternativas para manejar conjuntos de datos de gran escala con modestas arquitecturas computacionales.

En esta tesis, han sido propuestos, desarrollados y evaluados varios métodos kernelizados de semántica latente basados en factorización de matrices, donde, gracias a las capacidades no lineales de las representaciones basadas en kernels, los métodos propuestos pueden modelar las relaciones complejas entre las diferentes modalidades, lo que permite construir una representación multimodal enriquecida, incluso cuando una de las modalidades presenta datos incompletos. Además, los métodos propuestos han sido diseñados bajo una arquitectura escalable basada en dos estrategias principales: el aprendizaje en línea y el aprendizaje bajo presupuesto que permiten preservar bajos requerimientos computacionales en términos de uso de memoria y tiempo de procesamiento.

Una extensiva evaluación experimental muestra que las estrategias multimodales propuestas logran el estado del arte en varias tareas de análisis de datos, tales como la anotación multi-etiqueta y la clasificación multi-clase, así como la búsqueda y recuperación intermodal, y bajo diferentes configuraciones de aprendizaje, tales como aprendizaje supervisado, semi-supervisado y transductivo. Además, gracias a las estrategias de aprendizaje en línea y de aprendizaje bajo presupuesto propuestas en esta tesis, se preservan las capacidades de escalabilidad, lo que permite tratar con colecciones multimodales de gran escala.

Palabras clave: Información multimodal, análisis de datos multimodales, aprendizaje de máquina, indexación semántica latente, métodos del kernel, conjuntos de datos a gran escala.
# Content

## Acknowledgements

## Abstract

## Resumen

1 **Introduction**
   1.1 Motivation ............................... 2
   1.2 Problem statement ........................ 4
   1.3 Objectives ............................... 5
      1.3.1 General aim ........................... 5
      1.3.2 Specific objectives ................... 5
   1.4 Contributions ............................ 5
      1.4.1 Other contributions ................... 7
   1.5 Thesis Organization ........................ 9

2 **Basic Notions and Definitions**
   2.0.1 Latent semantic techniques ............... 10
   2.1 Kernel Methods ............................ 11
   2.2 Semi-supervised and transductive learning .... 11
   2.3 Transductive learning ........................ 11
   2.4 Performance measures ........................ 12
      2.4.1 classification measures ................ 12
      2.4.2 Information retrieval measures .......... 13
      2.4.3 Average Precision ........................ 13
      2.4.4 Mean Average Precision .................. 13

3 **A Review of Multi-modal Data Analysis**
   3.1 Multimodal data fusion .................... 14
      3.1.1 Multimodal fusion using kernel representation .... 14
      3.1.2 Multimodal semantic embedding ............. 15
      3.1.3 Multimodal semantic embedding from tensor representations .... 16
   3.2 Non-linear multimodal semantic embedding ........ 17
# Table of Contents

3.3 Large scale machine learning ................................................. 18
  3.3.1 Large scale semantic embedding learning .......................... 18
  3.3.2 Scalability in Kernel-based methods .............................. 19
3.4 Conclusions ......................................................................... 20

4 Multimodal matrix factorization for dimensionality reduction 21
  4.1 Two-way multimodal online matrix factorization for multi-label annotation 22
  4.1.1 Introduction .............................................................. 22
  4.1.2 Two-way multimodal matrix factorization ......................... 23
  4.1.3 Experiments and results ............................................. 28
  4.2 Semi-supervised dimensionality reduction via multimodal matrix factorization 30
  4.2.1 Introduction .............................................................. 30
  4.2.2 Related work ............................................................ 31
  4.2.3 Semi-supervised two-way multimodal online matrix factorization 31
  4.2.4 Experiments and results ............................................. 32
  4.2.5 Conclusions and future Work ..................................... 33

5 Transductive non-linear semantic embedding for multi-class classification 37
  5.1 Introduction ..................................................................... 37
  5.2 Related work .................................................................... 38
  5.3 Semi-supervised online kernel matrix factorization ................ 40
  5.3.1 Semi-supervised and transductive formulation .................. 41
  5.3.2 Online formulation ...................................................... 42
  5.3.3 SS-OKMF algorithm ................................................... 43
  5.3.4 Transductive learning .................................................. 43
  5.3.5 Prediction .................................................................... 43
  5.4 Experiments and results .................................................... 44
  5.4.1 Experimental setup ..................................................... 44
  5.4.2 Transductive classification .......................................... 45
  5.5 Conclusions ..................................................................... 52

6 Semi-supervised online kernel semantic embedding for multi-label annotation 53
  6.1 Introduction ..................................................................... 53
  6.2 Multi-label annotation based on semantic embedding methods .......... 54
  6.3 Semi-supervised online kernel matrix factorization for multi-label annotation 56
  6.3.1 Loss function ............................................................. 57
  6.3.2 Kernel learning .......................................................... 58
  6.3.3 Prediction .................................................................... 59
  6.3.4 Implementation details ............................................... 59
  6.4 Experiments and Results .................................................. 59
  6.4.1 Experimental setup ..................................................... 59
<table>
<thead>
<tr>
<th>Content</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.4.2 Determining the hyperparameters</td>
<td>60</td>
</tr>
<tr>
<td>6.4.3 Multi-label annotation performance</td>
<td>60</td>
</tr>
<tr>
<td>6.4.4 Multi-label Annotation Performance using kernel learning</td>
<td>60</td>
</tr>
<tr>
<td>6.4.5 Time-consuming and complexity</td>
<td>62</td>
</tr>
<tr>
<td>6.5 Conclusions</td>
<td>63</td>
</tr>
<tr>
<td>7 Online kernel semantic embedding for cross-modal retrieval</td>
<td>65</td>
</tr>
<tr>
<td>7.1 Introduction</td>
<td>65</td>
</tr>
<tr>
<td>7.2 Related work</td>
<td>66</td>
</tr>
<tr>
<td>7.3 Method</td>
<td>68</td>
</tr>
<tr>
<td>7.3.1 Semantic alignment</td>
<td>69</td>
</tr>
<tr>
<td>7.3.2 Label prediction and alignment</td>
<td>69</td>
</tr>
<tr>
<td>7.3.3 Learning with negative samples</td>
<td>69</td>
</tr>
<tr>
<td>7.3.4 Cross-modal reconstruction</td>
<td>70</td>
</tr>
<tr>
<td>7.3.5 Loss function</td>
<td>70</td>
</tr>
<tr>
<td>7.3.6 On-line multimodal kernel semantic embedding algorithm for cross-modal retrieval (MKSE-CM)</td>
<td>71</td>
</tr>
<tr>
<td>7.3.7 Cross-modal retrieval process</td>
<td>72</td>
</tr>
<tr>
<td>7.3.8 Implementation details</td>
<td>73</td>
</tr>
<tr>
<td>7.4 Experimental evaluation</td>
<td>73</td>
</tr>
<tr>
<td>7.4.1 Experimental setup</td>
<td>73</td>
</tr>
<tr>
<td>7.4.2 Dataset</td>
<td>73</td>
</tr>
<tr>
<td>7.4.3 Performance measurement</td>
<td>73</td>
</tr>
<tr>
<td>7.4.4 Hyper-parameter exploration</td>
<td>74</td>
</tr>
<tr>
<td>7.4.5 Kernel selection</td>
<td>74</td>
</tr>
<tr>
<td>7.4.6 Baseline methods</td>
<td>74</td>
</tr>
<tr>
<td>7.4.7 Results and discussion</td>
<td>75</td>
</tr>
<tr>
<td>7.4.8 Distance metric evaluation</td>
<td>75</td>
</tr>
<tr>
<td>7.4.9 The contribution of each objective function</td>
<td>76</td>
</tr>
<tr>
<td>7.4.10 Kernel evaluation</td>
<td>76</td>
</tr>
<tr>
<td>7.4.11 Budget impact</td>
<td>77</td>
</tr>
<tr>
<td>7.4.12 Comparison with the state-of-the-art methods</td>
<td>78</td>
</tr>
<tr>
<td>7.4.13 Evaluation with high quality descriptors</td>
<td>78</td>
</tr>
<tr>
<td>7.4.14 Image representation</td>
<td>79</td>
</tr>
<tr>
<td>7.4.15 Text representation</td>
<td>79</td>
</tr>
<tr>
<td>7.5 Conclusion and future work</td>
<td>80</td>
</tr>
<tr>
<td>8 Conclusions</td>
<td>82</td>
</tr>
<tr>
<td>8.1 Multimodal representations</td>
<td>82</td>
</tr>
<tr>
<td>8.2 Learning multimodal non-linear relationships</td>
<td>82</td>
</tr>
</tbody>
</table>
8.3 Large-scale multimodal kernel semantic embedding learning . . . . . . . . . 83
   8.3.1 Scalable semantic space construction . . . . . . . . . . . . . . . . . . 83
   8.3.2 Scalability of kernel-based methods . . . . . . . . . . . . . . . . . . 83
8.4 Future research directions . . . . . . . . . . . . . . . . . . . . . . . . . . . 84
   8.4.1 Regarding the budget construction . . . . . . . . . . . . . . . . . . 84
   8.4.2 Combination with deep learning models . . . . . . . . . . . . . . . . 84
   8.4.3 hyper-parameter exploration and selection . . . . . . . . . . . . . . 85

Bibliography 86
List of Figures

2-1 Example of ROC curve and its AUC representation ................. 13
4-1 Average classification accuracy and average required time for training in MNIST and CovType .................................................. 35
5-1 Semi-supervised Kernel Matrix Factorization Model .................. 41
5-2 Average classification accuracy for the MNIST dataset when incrementing the budget size .................................................... 50
5-3 Classification accuracy in G241C dataset for different training configurations 51
6-1 Conceptual model and the actual implementation of SS-OKSE ........ 55
6-2 Model implementation of SS-OKSE with kernel hyper-parameter learning. 58
6-3 Performance comparison in multi-label annotation. (hi: Histogram intersection kernel) .................................................... 61
6-4 Gamma learning progression for Chi-squared kernel in Delicious dataset 62
6-5 Performance comparison in multi-label annotation. Comparative using kernel hyper-parameter learning ......... 63
7-1 Conceptual model and the actual implementation of OKSE .......... 68
7-2 Proposed model: On-line Multimodal Kernel Semantic Embedding for Cross-modal retrieval (MKSE-CM) ................................. 71
7-3 Impact of the budget construction to the final performance .......... 78
# List of Tables

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-1</td>
<td>Execution time using GPU and CPU using the library pylearn2</td>
<td>28</td>
</tr>
<tr>
<td>4-2</td>
<td>Characteristics of the selected datasets to evaluate our method</td>
<td>29</td>
</tr>
<tr>
<td>4-3</td>
<td>Performance in F-Measure for each method</td>
<td>30</td>
</tr>
<tr>
<td>4-4</td>
<td>Dataset information and data partition for each dataset</td>
<td>32</td>
</tr>
<tr>
<td>4-5</td>
<td>Classification accuracy for different percentages of annotated instances in training</td>
<td>34</td>
</tr>
<tr>
<td>5-1</td>
<td>Error rate for 6 benchmark datasets in the transductive setup for different methods in the state of the art for ( l = 10 ) labeled samples</td>
<td>47</td>
</tr>
<tr>
<td>5-2</td>
<td>Error rate for 6 benchmark datasets in the transductive setup for different methods in the state of the art for ( l = 100 ) labeled samples</td>
<td>48</td>
</tr>
<tr>
<td>5-3</td>
<td>Main properties of transductive datasets</td>
<td>48</td>
</tr>
<tr>
<td>6-1</td>
<td>Dataset information a data partition for Delicious and Ohsumed datasets</td>
<td>62</td>
</tr>
<tr>
<td>6-2</td>
<td>Training time comparison in CPU and GPU devices</td>
<td>64</td>
</tr>
<tr>
<td>7-1</td>
<td>Different distance metrics used in the retrieval process</td>
<td>73</td>
</tr>
<tr>
<td>7-2</td>
<td>MAP scores by using different distance metrics in the retrieval process</td>
<td>75</td>
</tr>
<tr>
<td>7-3</td>
<td>MAP scores for cross-retrieval over the semantic space</td>
<td>76</td>
</tr>
<tr>
<td>7-4</td>
<td>MAP scores for cross-retrieval over the semantic label space</td>
<td>77</td>
</tr>
<tr>
<td>7-5</td>
<td>Comparison in mAP scores of MKSE-CM with different kernel functions.</td>
<td>77</td>
</tr>
<tr>
<td>7-6</td>
<td>Comparison in mAP scores of MKSE-CM against baseline methods.</td>
<td>79</td>
</tr>
<tr>
<td>7-7</td>
<td>Comparison in mAP scores of MKSE-CM against baseline methods using high quality descriptors</td>
<td>80</td>
</tr>
</tbody>
</table>
1 Introduction

The continued progress in digital data acquisition, storage and communication technology continuously generates huge amounts of multimedia data (i.e., text, images, audio, video, among others), which are made widely available thanks to Internet supported sharing platforms. This kind of information is a valuable source of knowledge, that usually is presented at different perspectives, views or modalities, that only can be properly exploited with the development of effective tools that properly integrate the complementary information obtained from each perspective of this heterogeneous kind of data.

The main goal of this thesis is to investigate effective and efficient methods to combine complementary evidence and model the relationships between multiple modalities of multimedia data in order to improve the access and analysis of the information to finally obtain valuable insights about the data. Multimodal latent semantic representation is a successful strategy that allows to combine and exploit the different views from these heterogeneous source of knowledge, by modeling relations between the different modalities and finding a new common low dimensional semantic representation space. However, an important limitation of approaches based on this strategy is that most of these models assume linear dependencies between the data modalities, giving a restricted and poor approximation to the inherent complex non-linear relationships present in this kind of data.

In this thesis, some alternatives to model these complex relationships have been investigated, especially, approaches based on kernel modeling have been proposed and evaluated. Classically, kernel methods have been shown to be very effective to solve non-linear problems such as classification, annotation, and clustering among others, in an efficient way, thanks to the ‘kernel trick’. However, kernel methods present high computational complexity. This drawback implies one of the most important challenges addressed in this thesis. Therefore, one of the main objectives in this thesis was to investigate new alternatives to handle large-scale datasets with modest computational architectures, which implied the design of scalable solutions that can process multimodal collections of hundreds of thousands or even millions of documents.

1.1 Motivation

Due to the fast advance in digital data acquisition, storage and distribution; large collections of multimodal data are becoming more common. These collections comprise a rich source of knowledge, therefore the development of effective and efficient searching, browsing and
1.1 Motivation

retrieval tools may result in benefits in multiple domains such as medicine, crime prevention, e-education, and publishing, among others [28].

Visual data is one of the richest sources of information, unfortunately, in most of the cases, it is not properly exploited due to the inherent complexity in analyzing this kind of data. For instance, in most of the existing retrieval systems for multimodal data with visual information, such as images or videos, manually-attached labels are used to identify the visual content. Moreover, the usual way to define the searching intention is using keywords as queries. This approach presents two main problems: first, manual keywords assignment is required, which is an arduous process that makes very difficult to ensure a reliable annotation for each visual instance in large collections; and second, the visual content is very rich, and in many cases it is not possible to describe all the information contained with a limited number of annotations, this is because an image can present different semantic hierarchies and the visual content can be interpreted in different ways depending on the search intention.

CBR (Content-based Retrieval) is a good alternative to allow access to unannotated visual instances [120]. This approach allows us to use a visual example, such as an image or a short video clip, to define the search intention. Under this setup, the system relies mainly on processing the visual contents to find relevant results. Unfortunately, this approach raises a very important issue, well known as the “semantic gap” [120], which implies that matching visual similarities does not necessarily leads to results with semantic validity. (It refers to the significant gap between the high-level concepts, which a human perceives and the low-level features which are used to describe the visual content).

In order to overcome the visual semantic gap, many multimodal semantic analysis strategies have been proposed in the last years [23, 24, 99]. These strategies try to minimize this problem by taking advantage of complementary information from other data modalities. This is because, usually, the visual information does not appear in isolation, but rather with other kinds of data, for instance, in web pages, it is common to find images accompanied with unstructured related text, and videos commonly contain other modalities such as audio and text data from the captions. This is useful because multimodal data originated from the same source tend to be correlated [81]. However, correlations among modalities could be very complex and, in general, non-linear.

The purpose of multimodal semantic analysis is to generate a data-driven representation, which simultaneously incorporates semantic descriptions from all the available data modalities. So, instead of requiring human intervention to define and assign labels to describe the visual content, an algorithm is designed to automatically discover semantic categories from the combination of all modalities, and incorporate that information in the multimodal document signature.
1.2 Problem statement

This thesis aims to investigate efficient and effective methods that take as input visual data with other associated data modalities from a multimodal collection and produce a common semantic representation that models the relationships between these different modalities. Classical latent semantic models, can boost the performance in tasks like content-based retrieval, exploration and automatic annotation and classification, among others, by modeling relations between the different modalities. Nevertheless, one important limitation of most approaches based on latent semantic analysis is that they assume linear dependencies between the data modalities. However, it is reasonable to expect that there is an inherent non-linearity in the nature of the data. Therefore, this kind of linear models can only give a limited and restricted approximation of the nature of the data and the complex relationships between the different modalities. This important issue motivates the main research question in this thesis that is:

• **How to properly model non-linear relationships between different data modalities?**

By modeling non-linear relationships we aim to create an effective multimodal semantic model, which can deal with the complexity of the original data structures. A popular and successful alternative to model these non-linearities is using kernel-based methods. This kind of methods have succeeded in a number of tasks such as classification, annotation and clustering, but have not been extensively explored for semantic embedding, therefore, a second research question is:

• **How to perform non-linear semantic embedding using kernel methods?**

Kernel-based methods also present other important characteristics, for instance, with an adequate selection of the kernel it is possible to achieve robustness to contamination of the training sample [66] by making it unaffected by outliers. So, another important question is:

• **In which scenarios non-linear semantic embeddings exhibit clear advantages over linear semantic embedding?**

The richness of large collections of multimodal data is precisely its huge amount of data that can describe the wide variability with which the information can be presented. This would be a great resource to feed sophisticated algorithms to model a rich semantic representation. Kernel-based methods have obtained accurate and robust results in many tasks involving non-linearities, and would be a promising tool to model semantic representations, but, unfortunately kernel-based methods present an important drawback, as more data is used to train a model, much more computational resources are required, growing in an accelerated way, making them infeasible for large collections. Consequently, another fundamental research question is:
1.3 Objectives

1.3.1 General aim

To propose a method for non-linear semantic embedding for large-scale multi-modal data based on kernel methods.

1.3.2 Specific objectives

- To propose and/or adapt adequate feature extraction methods for each kind of modality.
- To design non-linear semantic analysis methods that combine visual representations with other complementary modalities representations.
- To propose and/or adapt scalability strategies for non-linear semantic analysis.
- To evaluate the performance of the method in different data analysis and information retrieval problems.

1.4 Contributions

The main contributions of this work are the development of models to learning multimodal relationships between different modalities based on semantic embedding. Several strategies to model a common semantic space were proposed and evaluated over classification and multi-label tasks, and in supervised, semi-supervised and transductive learning setups.

1. Preliminary linear models for multimodal semantic embedding based on matrix factorization were proposed. The most remarkable characteristic of the proposed models is the reformulation of the matrix factorization problem from the classical optimization based on multiplicative rules to additive rules that allows performing the optimization based on stochastic gradient descent. This allowed the implementation of highly scalable algorithms.
1.1. Inspired by the idea of autoencoders in neural networks, an implicit calculation of the semantic space was proposed.


1.2. A reformulation of the previous algorithm that allows modeling the semantic space by using labeled and unlabeled samples was proposed.


1.3. An alternative online matrix factorization method that defines the same non-negative restriction an optimizes the same loss function of the classical non-negative matrix factorization method was proposed.


The source code of the methods proposed in these articles is available at: [https://bitbucket.org/lvbeltranb/sem_models/](https://bitbucket.org/lvbeltranb/sem_models/)

2. A non-linear strategy to model the common semantic space based on kernel methods was proposed. This method takes advantage of two main strategies to achieve high scalability: online learning and learning-in-a-budget.


The source code is available at: [https://bitbucket.org/javanegasr/ssokmf](https://bitbucket.org/javanegasr/ssokmf)

3. The previous strategy to model the common semantic space has some disadvantages related to the explicit construction of the semantic space. For this reason, a new kernel strategy that learns the semantic projection and back-projection from and to the semantic space was proposed.

3.1. An initial version of the method was presented in the Iberoamerican Congress on Pattern Recognition. This article has won the IAPR Best Paper Award.

---


3.2. An extended version that learns the kernel function parameters was proposed.


The source code of the methods proposed in these articles is available at: [https://bitbucket.org/javanegasr/ss-okse](https://bitbucket.org/javanegasr/ss-okse)

4. A new semantic embedding strategy that allows modeling a common semantic representation based on two unstructured data modalities was proposed. For this purpose, the last matrix factorization strategy was extended to modeling simultaneously a factorization problem for each modality while a semantic alignment strategy enforces that both factorizations share the same semantic representation. This strategy was evaluated under the challenging task of cross-modal retrieval.


The source code of the method proposed in this article is available at: [https://bitbucket.org/javanegasr/cross-ocmf](https://bitbucket.org/javanegasr/cross-ocmf)

### 1.4.1 Other contributions

Additional papers were published as a result of collaborations and primary studies performed in different and related research areas.

5. Some indexing strategies based on matrix factorization models were evaluated for different data modalities of different nature, such as images, videos, and text.


6. Several strategies for text representation were explored and evaluated in the task of event extraction.

7. Another multimodal semantic representation strategy for multimodal biomedical data was proposed.


8. In collaboration with the Laboratory of Language Technologies of the Computational Sciences Department at the National Institute of Astrophysics, Optics, and Electronics (INAOE), some methods for automatic image annotation and captioning were proposed, implemented and evaluated.

8.1. An unsupervised approach for image captioning based on a two steps image-textual retrieval process was proposed.


8.2. Some papers were publish as a result of our participation in the ImageCLEF challenge in the task of scalable concept image annotation, which consist in the automatic annotation of images using noisy web page data.


9. An efficient implementation of an online kernel matrix factorization by exploiting graphics processing units (GPUs) was evaluated.


---

1.5 Thesis Organization

The remaining chapters of the thesis are organized as follows:

- **Chapter 2: Review of Multi-modal Data Analysis.** This chapter discusses previous works related to multimodal methods, including non-linear strategies.

- **Chapter 3: Multimodal Matrix Factorization for Dimensionality Reduction.** This section presents an initial approach for dimensionality reduction that takes advantage of annotated data to model a semantic low-dimensional space representation. One of the remarkable characteristics of the proposed method is its online learning formulation that allows it to deal with large collections of data by achieving a significant reduction in computational requirements.

- **Chapter 4: Online kernel semantic embedding.** This section presents the first attempt to modeling non-linear relationships based on kernel methods. In this approach, a learning-in-a-budget strategy is proposed. This mitigates one of the main drawbacks of kernel-based methods that is the high memory requirements caused by the kernel matrix (Gram Matrix) that grows quadratically with the total number of training instances.

- **Chapter 5: Two-way Online Kernel Semantic Embedding.** This section presents another kernel-based method strategy that follows a different approach based on fully stochastic gradient descent learning, which can be directly implemented in standard data-flow architectures, this characteristic allows to make several extensions based on deep learning strategies.

- **Chapter 6: Conclusions.** The final chapter presents the main conclusions and discussions of this dissertation, summarizes the main contributions, and highlights the most important findings. Also, some future research directions are presented and discussed.
2 Basic Notions and Definitions

This chapter recalls briefly some notations, concepts, and definitions which are required for the following chapters. The first part of this chapter describes a set of classic related methods and concepts that serve as basics to understand the motivation and the formulation of the proposed methods in this thesis, and in the final part, the main evaluation metrics to assess the quality of the proposed methods are explained.

2.0.1 Latent semantic techniques

Latent semantic embedding is a successful approach initially proposed for information retrieval and later extended to several data analysis tasks. This approach implies to find a transformation from the original data representation to a lower rank approximation. This strategy presents three important advantages: first, perform this transformation implies a considerable reduction in computational requirements, second, this approach extracts implicit similarities of correlated data helping to reduce the effects caused by the polysemy and the synonymy, and finally, this process dismiss the inherent noise of the data.

Several alternatives have been proposed to achieve a semantic representation. Among the most relevant, we can mention the classical LSA method, which is based on the technique of singular value decomposition (SVD), Additionally, there are other alternatives that follow a probabilistic strategy, such as, the PLSA (Probabilistic Latent Semantic Analysis) [56] and the NMF (Non-negative matrix factorization) [75].

Non-negative matrix factorization

Nonnegative Matrix Factorization (NMF) finds a low rank approximation by performing a matrix decomposition. For an input matrix $X \in \mathbb{R}^{n \times l}$, containing $l$ data samples with $n$ features in its column vectors, NMF construct a low rank approximation of the data using non-negativity constraints:

$$X \approx WH$$

$W, H \geq 0$

where $W \in \mathbb{R}^{n \times r}$ is the basis of the vector space in which the data will be represented and $H \in \mathbb{R}^{r \times l}$ is the new data representation using $r$ latent factors. One of the reasons
for the success of NMF is its ability to find parts of objects. Furthermore, in comparison with the standard latent semantic indexing or singular value decomposition (SVD) that had orthonormal restrictions but no constraints in sign NMF gives more interpretability and finds better structural patterns.

2.1 Kernel Methods

The name of kernel methods comes from the use of kernel functions, which are functions that return the inner product between mapped data points in a higher dimensional space [113]. The remarkable characteristic of Kernel methods is that they provide a structured way to enable linear algorithms to address non-linear problems in a transformed feature space, where this transformation is typically nonlinear and to a higher dimensional space. The viability of kernel methods is that by using kernel functions an explicit projection of the data into the higher dimensional feature space is not necessary, but instead, an implicit computing of the inner products between the images of all pairs of data in the feature space is calculated. This strategy is called the “kernel trick”.

2.2 Semi-supervised and transductive learning

Semi-supervised learning falls between unsupervised learning (without any labeled training data) and supervised learning (with completely labeled training data). Many machine-learning researchers have found that unlabeled data, when used in conjunction with a small amount of labeled data, can produce considerable improvement in learning accuracy. The acquisition of labeled data often requires a skilled human agent or a physical experiment and in many situations the cost associated to construct a fully labeled training set infeasible, whereas acquisition of unlabeled data is relatively inexpensive. In such situations, semi-supervised learning can be of great practical value.

2.3 Transductive learning

In the traditional supervised learning (inductive learning) the result of the inference relies only on the training set and it is always independent of the test set. In transductive learning, the algorithms use the test instances by comparing them to all of the training instances. Usually, this can produce better performance in analysis tasks, but at a much higher computational cost.
2.4 Performance measures

Throughout this thesis, several methods have been proposed, which have been evaluated in different data analysis tasks such as clustering, multi-labeling, classification, and information retrieval, for this purpose several performance metrics have been used. In this section, we present the most relevant.

2.4.1 classification measures

F-Measure

F-Measure (also know as F1 score or F-score ) is a measure of accuracy. It considers both the precision that is the number of correct positive results divided by the number of all positive results returned by the classifier, and recall that is the number of correct positive results divided by the number of all relevant samples. The F1 score summarizes the results by calculating their harmonic average, where an F1 score reaches its best value at 1 (perfect precision and recall) and worst at 0.

\[
F_1 = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}
\]  (2-1)

Error rate

Error rate is the percentage of misclassifications over all the dataset, also can be seen as the reciprocal of the accuracy.

ROC curve

The ROC curve (receiver operating characteristic curve) is a graphical plot that illustrates the behavior of a binary classifier as its discrimination threshold is varied. The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings:

True positive rate (TPR), also known as sensitivity corresponds to the proportion of positive data points that are correctly considered as positive, with respect to all positive data points. In other words, the higher TPR, the fewer positive data points we will miss.

False positive rate (FPR), also known as fall-out corresponds to the proportion of negative data points that are mistakenly considered as positive, with respect to all negative data points. In other words, the higher FPR, the more negative data points will be misclassified.

A common way to summarize the performance of a method is measuring the area under the ROC curve (AUC). An area of 1 represents a perfect test; an area of .5 represents a worthless test.
2.4 Performance measures

Figure 2-1: Example of ROC curve and its AUC representation.

Figure 2-1 presents an example of ROC curve, the dark area corresponds to the AUC. The dashed line in the diagonal presents the ROC curve of a random predictor that is commonly used as a baseline, this has an AUC of 0.5.

2.4.2 Information retrieval measures

Evaluation measures for an information retrieval system are used to assess how well the search results satisfied the user’s query intent.

2.4.3 Average Precision

Average precision (AP) is the average of precision values at all ranks where relevant documents are found. The average precision for the top \( N \) documents is defined as follows:

\[
AP@N = \frac{1}{m} \left[ \sum_{k=1}^{N} P(k) \cdot rel(k) \right] 
\]  

(2-2)

where \( rel(k) \) is an indicator that says whether that \( k \)-th document is relevant (\( rel(k) = 1 \)) or not (\( rel(k) = 0 \))

2.4.4 Mean Average Precision

Mean Average Precision (MAP) is the standard single-number measure for comparing search algorithms. The calculation of MAP consists of determining the AP for each individual query \( q_i \), and then calculating the average (mean) of APs on the set of queries \( Q \):

\[
MAP(Q) = \frac{1}{|Q|} \sum_{i} AP(q_i) 
\]  

(2-3)
3 A Review of Multi-modal Data Analysis

This chapter discusses previous works related to multimodal data fusion and analysis, including non-linear strategies and related strategies to deal with large-scale problems.

3.1 Multimodal data fusion

There are many alternatives to fusing the different modalities of multimodal data [70]. The most general classification is given by the level of the fusion: early fusion (or feature level) and late fusion (or decision level). In early fusion, the features extracted from the input data are first combined and sent as a new input to a single analysis unit that performs the analysis task, and in late fusion, initial analysis is performed and local decisions are taken for each modality and then these local decisions are combined using a decision fusion. This review focuses on early fusion strategies and especially in strategies that model the correlation between different modalities. The reason to focus only on early fusion strategies is that this kind of methods can capture the true essence of multimodal data collaboration as all the features are combined in a unified representation.

3.1.1 Multimodal fusion using kernel representation

There are many works that use kernel representation as an alternative to fusing heterogeneous feature representations. In this approach, the fusion is performed at the kernel-level instead of feature-level. This presents two main advantages: first, kernel-level fusion avoids the sparsity of data which often occurs in the combined feature space given by feature-level fusion; and second the optimized weights of the basic kernels may provide information on the relative importance of each data modality. In order to determine the optimal combination of individual modalities, Wu et al. [145] proposed the Super-kernel Fusion, taking into account three important aspects: modality independence, the curse of dimensionality, and fusion-model complexity. Yen et al. [82] evaluated a similar approach in content-based image retrieval task, by proposing a semi-supervised technique for fusing multiple visual features and label information in the domain of kernel matrices in order to achieve a flexible and unified view for handling multimodal information and preserving the intrinsic structure of each information modality. The proposed method combines all the corresponding kernel
matrices into an informative one, based on Kernel Alignment [35] to estimate the degree of agreement between two kernel matrices. Sahbi et al. [109] present a different strategy for kernel information fusion, which does not try to fuse kernels from different modalities, but try to enrich the kernel representation from one modality using semantic information extracted from the other one. This kernel method called Context-dependent Kernel contains context information, i.e., this kernel not only contains similarity information since visual representation but also contains contextual cues resulting from surrounding tags. In this way, this method enriches the kernel representation by combining the visual representation with textual information; in this case, the text representation is conformed by surrounding tags. This method constructs a new kernel representation by constructing images relationships by finding shared tags between images and constructing an adjacency matrix which modeled a multimodal database as a graph where nodes are images and edges represent the shared tags. The proposed kernel design method is based on the optimization of an objective function mixing a fidelity term, a context criterion and a regularization term. The fidelity term takes into account the visual content of images, so highly visually similar contents encourage high kernel values.

3.1.2 Multimodal semantic embedding

One commonly used strategy for modeling the relationships between different modalities is the construction of a common semantic space where each representation of each modality can be projected. One of the first attempts published using this strategy is the seminal work of Barnard et al. [8], which introduced the multimodal Latent Dirichlet Allocation (mmLDA) algorithm to learn the joint distribution of image regions and words. Several subsequent works have proposed related probabilistic algorithms to approach this problem, including multilayer Probabilistic Latent Semantic Analysis (mmPLSA) [80], and topic regression multimodal LDA [99]. The core concept underneath all these methods is to model latent variables for discovering the hidden structure of the data, using different prior distributions and different dependencies between variables. This idea has shown to be a powerful tool for learning multimodal relationships, however, the main bottleneck of all these models is their computational cost, which makes them difficult to implement and scale for large data collections. Several matrix factorization algorithms have also been proposed for modeling latent structures, and have been used for learning multimodal interactions. For instance, Chandrika and Jawahar [24] evaluated a multimodal Latent Semantic Indexing (mmLSI) algorithm, based on Singular Value Decomposition (SVD). Also, several algorithms for learning multimodal relationships using Nonnegative Matrix Factorization (NMF) have been proposed by Caicedo et al. [23] and Akata et al. [2]. The main difference between both algorithms, the asymmetric NMF (aNMF) and the multimodal NMF (mNMF) is the underlying cost function that each algorithm optimizes, the former uses NMF under the Kullback Leibler Divergence (NMF-KLDiv) and the latter uses NMF under the Frobenius
NMF-KLDiv has been demonstrated to be equivalent to the PLSA algorithm since they optimize the same objective function [36]. Wang et al. [137] present the Orthogonal Nonnegative Matrix Tri-factorization (O-NMTF) that is a variation of NMF that allows incorporating intra-type information and assigns orthogonality constraints on factor matrices for manifold regularization. This sets interesting theoretical insights to model matrix decompositions for learning latent factors. However, even though matrix factorization can provide a promising framework for multimodal learning, these algorithms still require significant computational resources for large-scale learning.

Another popular approach for multimodal data analysis based on matrix factorization is using statistical correlation analysis methods. Canonical Correlation Analysis (CCA) and Partial Least Squares (PLS) methods are some of the most common approaches to solve cross-modal retrieval problem based on this approach. Both methods try to learn linear transformations that maximize the correlation between two data modalities and simultaneously create a common semantic space. Several works have been proposed based on CCA, for instance, Chen et al. [29] propose a variation of CCA to solve the problem of cross-modal retrieval between images and text, Aryafar et al. [6] use CCA to model correlation between audio signals and lyrics in order to outperform the performance in content-based retrieval. Sargın et al. [110] evaluated CCA in audio-visual synchronization and fusion for the speaker identification problem, and Lazaridou et al. [72] presented another approach to combining prior linguistic information and visual knowledge in order to learn to associate new images with words. More recently, based on PLS, He et al. [54, 52] have proposed a method to address the of cross-modal retrieval in texts and images.

### 3.1.3 Multimodal semantic embedding from tensor representations

A tensor is a multi-way array. This is the natural generalization of vectors (one-way tensor) and matrices (two-way tensor). Due to its multi-way structure, the tensor is the most natural manner to represent multimodal data, providing powerful tools for analysis and fusion of multimodal data, based on a strong mathematical backbone for the discovery of underlying hidden complex data structures. Tensors have been applied in many fields, such as computer vision, signal processing, and information retrieval, among others [84, 71]. Several generalized strategies for low-rank decomposition for tensor representation have been proposed. For instance, Wu et al. [144] proposed a tensor strategy, which represents a video by a joint representation of its three different modalities: image representation of a frame, audio representation and text representation of captions. The paper proposed a transductive support tensor machines algorithm to train classifiers to detect semantic concepts in video shots. In a similar way, Yanan Liu et al. [86], proposed to represent a video shot as a 3rd-order tensor, taking advantage of its sparse representation, in order to overcome the curse of dimensionality problem caused by the traditional basic way to join video features based on concatenated vectors. Recently, Fu et al. [42] proposed the Transductive Multi-
3.2 Non-linear multimodal semantic embedding

The set of multimodal semantic embedding methods presented in the previous section have an important limitation. That is, that all of them are based on models that assume linear dependencies between the data modalities, giving a restricted and poor approximation to the inherent complex non-linear relationships presented in multimodal data. There are some works that overcome this problem by proposing a kernelized version of classical embedding algorithms. For instance, Wang et al. [136] proposed a supervised learning method for dimensionality reduction called kernel maximum margin projection (KMMP), which unlike traditional dimensionality reduction algorithms such as principal component analysis (PCA) and linear discriminant analysis (LDA), which only see the global Euclidean structure, KMMP is designed for discovering the local manifold structure by reproducing a non-linear embedding. KMMP is a kernel generalization of the maximum margin projection (MMP) that discovers the local manifold structure by maximizing the margin between positive and negative examples at each local neighborhood. Also, in the case of matrix factorization, some kernelized versions have been proposed: Zitnik et al. [162] present a kernelized variant of matrix tri-factorization and Gonen et al. [44] proposed kernelized Bayesian matrix factorization method that is able to integrate multiple modalities sources by coupling matrix factorization with multiple kernel learning. Also, several approaches have been presented based on a kernelized version of canonical correlation analysis (KCCA) [122, 60]. Despite the effectiveness shown by these methods, they require keeping in memory all the dataset covariance matrix. In many implementations, these methods employ Singular Value Decomposition (SVD), whose computational cost limits their ability to scale on large datasets. More recently, Costa et al. [32] proposed an ensemble method called Semantic Correlation Matching (SCM) that maximize the correlation in the projections of different modalities and
combines label information.

In the area of neural networks, autoencoders and cross autoencoders are deep learning methods that show the ability to learn compact representations of the original data, embedding them into a new latent space. Feng et al. [41, 40] proposed an autoencoder and cross-autoencoder models where the correlation between hidden layers are maximized. In the same area of neural networks, some deep variations of canonical correlation analysis have been proposed [112, 3, 4], showing that the use of several layers helps to better model the complex non-linear relationships between the different modalities.

Some recent works proposed supervised alternatives [112, 98, 32, 121] to model a common space that rely on rich semantic annotation, such as tags, keywords or captions that usually appear in the multimedia content on the web (for instance, news archives, Wikipedia pages and blog posts, among others), this kind of methods present the best performance, showing that with the use of semantic information a richer common semantic representation can be constructed.

### 3.3 Large scale machine learning

With the increasingly large size of datasets, a phenomenon usually called *big data*, several machine learning researchers have concentrated their attention on improving the scalability of machine learning methods. Semantic embedding learning and kernel-based methods are not the exceptions, however, the scalability of these two approaches have been addressed separately. The following subsections review the efforts in this direction.

#### 3.3.1 Large scale semantic embedding learning

The main drawback of current multimodal learning strategies is that the associated algorithms are memory and computationally intensive [24], making them difficult to use in a large-scale setup. For instance, the work of Romberg et al. [106] aims to build a multimodal index for a collection of 10 million Flickr images using a PLSA-based algorithm. However, only a small sample of 10,000 images could be applied to the learning algorithm, losing the potential of such a vast amount of data. Some works have investigated strategies for efficient parallelization of probabilistic models, overcoming underlying problems such as sharing data across the different nodes and other memory restrictions [144], similar approaches have been proposed parallelization strategies for matrix factorization algorithms [43, 83], however, large computational resources dedicated to decompose big matrices are still required. Besides, generic distributed matrix factorization algorithms are not applicable for a multimodal learning setup, which has two input matrices instead of one. There are some works that try to make semantic embedding approaches suitable for large-scale collections by reformulating the original algorithms. For example, Hsan et al. [129] modified the original algorithm called Modified Multi-stage Convex Relaxation (MMCR) in some way that is possible to reduce
3.3 Large scale machine learning

the time complexity and the amount of storage, achieving a significant reduction in time complexity. A different approach is presented by Mairal et al. [88] who proposed an online algorithm based on stochastic gradient descent for non-negative matrix factorization based on sparse coding by adding a sparsity-inducing penalty to the objective function, which allows processing datasets of millions of instances with low memory consumption. Caicedo et al. [23] also proposed an online matrix factorization algorithm for multimodal analysis, showing a speedup of about 25x over classical matrix factorization algorithms, due to a significant gain in the convergence speed with very low memory requirements since they do not need to keep in memory all training data. Also, Jason Weston et al. [141] present a scalable architecture, proposing methods that learn to represent images and annotations jointly in a low dimensional embedding space. To make training time efficient, they also propose a loss function based on stochastic gradient descent (SGD) approach. This formulation allowed them to use up to 10 million of samples for training. These works demonstrate the potential of online learning for matrix factorization by requiring little computational resources, in terms of memory demand and time complexity.

3.3.2 Scalability in Kernel-based methods

Kernel-based methods are a good alternative for modeling non-linear dependences, but these methods present extra computational requirements, mainly generated by the construction of the Kernel matrix that follows a quadratic space complexity with relation to the number of example instances, leading to large storage requirements and high computational cost for training and testing, making them infeasible for a large-scale setup. Several alternatives have been proposed to address this drawback [155]. Initial attempts were focused on parallelized architectures to compute the kernel matrix [101], unfortunately, this approach still requiring larger computational resources and implies to deal with communication overhead and the inherent complexity of distributed systems. Subsequent proposals have been based on using a bounded number of training samples (Sparse Kernel Learning in Batch Setting) [65, 33], and more recent proposals combine this strategy with an online learning setup: Online Sparse Kernel Learning [154] and Online Kernel Learning on a Budget [59, 158]. The main idea of Online Sparse Kernel Learning is to adopt a sampling strategy for reducing the number of example instances, and the objective of Online Kernel Learning on a Budget is to generate a kernel-based method with a fixed number of instances. Finally, the most recent proposals seek to find a low-range approximation to kernel matrix [87, 38, 78, 115], unfortunately, finding a low-range approximation is also a computationally demanding task that has many restrictions for a large-scale setup. For instance, the best rank-k approximation can be obtained by SVD (Singular Value Decomposition) but this is also computationally prohibitive when the number of instances grows to tens of thousands, One alternative is to use the randomized SVD [50] but it still needs computation and storage of the entire dense kernel matrix. In order to try to overcome the prohibitive time and space complexity
of SVD, the Nyström method has been proposed [142]. This method generates a low-rank approximation based on a sampled subset of columns of the kernel matrix. Also, many strategies have been proposed to improve over the basic Nyström approximation: ensemble Nyström approximations [69], Nyström with k-means to obtain benchmark points [152], and randomized Nyström [79] and combinations of these [78]. Another way to approximate the kernel matrix is by using Random Kitchen Sinks (RKS) [102, 119, 7], which approximates the Gaussian kernel function based on its Fourier transform. Le et al. [73] sped-up the RKS by using fast Hadamard matrix multiplications. Recently, Si et al. [115] proposed a method that considers the low-rank and clustering structure of the kernel matrix, by performing Nyström approximations in sub-blocks, allowing outperforms state-of-the-art low-rank kernel approximation methods in terms of speed and memory usage.

### 3.4 conclusions

Motivations for data fusion are numerous. As we can see in the revision of previous work, there is a wide range of works that attempt to fusing multimodal data to address several modeling and data analysis tasks such as classification, multi-label annotation, clustering and multimodal/cross-modal retrieval among others. Among all the data fusion alternatives, semantic-based methods present several advantages that allow eliminating the redundancy and the noise presented in the manifold structure of the original high-dimensional feature representation and tackle the curse of dimensionality by compressing the information in a more representative and expressive reduced new presentation. However, the main bottleneck of all these models is their computational cost, which makes them difficult to implement and scale for large data collections.

On the other hand, according to the state-of-the-art review, a good alternative to model non-linearities are the strategies based on kernel methods, but this kind of strategies are computationally expensive in terms of memory and time requirements, making them infeasible for very large collections of images. Fortunately, there are some strategies that could alleviate this computational cost.

This review stands the motivations of using as alternatives semantic embedding for fusing multimodal data and kernel methods for non-linear modeling, even though the limitations of these two approaches regarding their scalability are clearly evidenced. Although some alternatives to make them scalable are presented in the literature, there are not works that present scalable alternatives when kernel-based methods and semantic embedding strategies are used together. This is because the scalability issues from these two approaches are of different nature. This is the main contribution of this thesis, which is to provide a unified framework that combines the advantages of semantic embedding and kernel methods in a unique learning strategy that tackles the scalability issues of both approaches simultaneously.
4 Multimodal matrix factorization for dimensionality reduction

Dimensionality reduction and manifold learning are techniques widely used today in many machine learning tasks such as regression, annotation, classification, clustering, pattern recognition and information retrieval, among others. These techniques help to eliminate the redundancy and the noise presented in the manifold structure of the original high-dimensional feature representation and tackling the curse of dimensionality by compressing the information in a more representative and expressive reduced set of variables that preserve the most important characteristics and properties of the initial representation. This is done by finding a transformation that does not alter the relevant information presented by the initial data set. These techniques can be used in unsupervised as well as supervised and semi-supervised approaches, where, unsupervised dimensionality reduction is mainly used with the aim of exploring the data structure and extracting meaningful information from data without any prior information. In contrast, in supervised dimensionality reduction, specific targets (labeled instances) of interest are used to guide the dimensionality reduction process. Even though supervised approaches can exploit the labeled data in order to improve classification performance, they require every training instance to be labeled. A proper annotation of a whole dataset is an arduous process, and for large-scale real-world collections is infeasible to ensure a reliable annotation for each instance. So, in many cases, we are in a situation where we have a big quantity of potential data for training our algorithms, but only a small fraction of properly annotated instances can be used. Even so, non-annotated evidence would present valuable information about the manifold structure of the data that should be exploited in some way. This motivates the implementation of semi-supervised approaches.

This chapter presents a dimensionality reduction algorithm based on matrix factorization evaluated under supervised and semi-supervised learning setups in two different data analysis tasks: multi-label annotation and multi-class classification.

This work has been presented in Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, 20th Iberoamerican Congress, CIARP 2015, and in Proceedings of the 4th International Conference on Pattern Recognition Applications and Methods, ICPRAM 2015 (see [11, 133]).
4.1 Two-way multimodal online matrix factorization for multi-label annotation

This section presents a matrix factorization algorithm for multi-label annotation. The multi-label annotation problem arises in situations such as object recognition in images where we want to automatically find all of the objects present in a given image. The solution consists in learning a classification model able to assign one or many labels to a particular sample. The method presented in this section learns a mapping between the features of the input sample and the labels, which is later used to predict labels for unannotated instances. The mapping between the feature representation and the labels is found by learning a common semantic representation using matrix factorization. An important characteristic of the proposed algorithm is its online formulation based on stochastic gradient descent which can scale to deal with large datasets. According to the experimental evaluation, which compares it with state-of-the-art multi-label latent space embedding (MLLSE) algorithms the proposed method presents a competitive performance improving, in some cases, previously reported results.

4.1.1 Introduction

Multi-label annotation has been an active research area in the last years due to its potentially impact in an increasing number of new applications like music categorization [128], functional genomics [156], video content analysis [138], noise detection [100], image understanding [143] and image search [116], among others [130]. The problem of multi-label annotation (or classification) refers to the problem where a single instance can be simultaneously assigned to multiple classes. This differs from multi-class classification where each sample is assigned to only one class. It means that, in multi-class classification, classes are assumed mutually exclusive, but in multi-label classification classes are often correlated.

A common approach to address multi-label annotation is to handle this problem as a conventional classification problem. i.e., multiples classifiers are trained, and only one binary classifier is used per label. In this way a new instance is annotated by independently applying the set of classifiers. Due to that one classifier is required for each label, this approach may not scale well when there is a large number of labels.

An alternative to deal with the large number of labels is to find a compact representation of them by using a dimensionality reduction method. This approach is followed by multi-label latent space embedding methods, which have shown competitive results.

In this section we describe a method for multi-label annotation based on semantic embedding. The proposed method finds a common semantic space for the original features representation of an instance and its corresponding labels to model a direct mapping between the feature representation and annotation labels. An important characteristic of the proposed method is its formulation as an online learning algorithm based on stochastic gradient decent, which
allows it to deal with large collections of data, achieving a significantly reduction in memory requirements and computational load.

### 4.1.2 Two-way multimodal matrix factorization

If we describe the feature representation of an instance as an $n-$dimensional vector, we can represent the entire collection by a matrix $X_v \in \mathbb{R}^{n \times l}$, where $l$ is the number of elements. In the same way we can represent the labels associated to a specific instance by an $m-$dimensional binary indicator vector, where $m$ is the total number of possible labels, and in the $j-th$ position in the vector we have a value of 1 if the $j-th$ label is assigned to the image or 0 otherwise. So, we can construct a label indicator matrix $X_t \in \mathbb{R}^{m \times l}$.

In this section, we propose a model that finds a mapping $F : \mathbb{R}^n \rightarrow \mathbb{R}^r$, from the sample representation space to a semantic space, and simultaneously finds a back-projection from the semantic space to the original space $G : \mathbb{R}^r \rightarrow \mathbb{R}^n$, where $n \gg r$. So we want to find two linear transformations what allows to project the original data representation to a lower-dimensional space (semantic representation) and at the same time learns to reconstruct from this lower-dimensional representation the original data.

If we assume that both $F$ and $G$ are linear mappings with coefficients $W_v$ and $\tilde{W}_v$ respectively, for an entire collection we want to find a reconstruction of the original feature representation as follows:

$$X_v \approx \tilde{W}_v W_v X_v \quad (4-1)$$

where $W_v \in \mathbb{R}^{r \times n}$ is an encoder matrix that projects the original representation to a lower-dimensional semantic space and $\tilde{W}_v \in \mathbb{R}^{n \times r}$ is a decoder matrix that reconstruct the original data. In the same way for the label information, we have:

$$X_t \approx \tilde{W}_t W_t X_t \quad (4-2)$$

where $W_t \in \mathbb{R}^{r \times m}$, $\tilde{W}_t \in \mathbb{R}^{m \times r}$ are the encoder and decoder matrices for the label information respectively.

Our main purpose is to learn a mapping between the original features and label information. Therefore, we also seek that the previous transformation matrices also satisfy the following condition:

$$X_t \approx \tilde{W}_t W_v X_v \quad (4-3)$$

This condition forces both the original representation and the label representation to share the same semantic space and defines a mapping between both representations.
Finally, we can formulate this problem as an optimization problem by minimizing the following loss function:

$$L \left( X_v, X_t, W_v, W_t, \tilde{W}_v, \tilde{W}_t \right) = \alpha \left\| X_v - \tilde{W}_v W_v X_v \right\|_F^2 + (1 - \alpha) \left\| X_t - \tilde{W}_t W_t X_t \right\|_F^2$$

$$+ \delta \left\| X_t - \tilde{W}_t W_v X_v \right\|_F^2$$

$$+ \beta \left( \| W_v \|_F^2 + \| \tilde{W}_v \|_F^2 + \| W_t \|_F^2 + \| \tilde{W}_t \|_F^2 \right)$$

(4-4)

where $\alpha$ controls the relative importance between the reconstruction of the instance representation and the label representation, $\delta$ controls the relative importance of the mapping between instance features and label information and, $\beta$ controls the relative importance of the regularization terms, which penalizes large values for the Frobenius norm of the transformation matrices.

**Gradient descent solution**

The problem above has a non-convex objective function (eq. 4-4). However, this function is differentiable for all the unknown parameters and the solution can be computed using a gradient descent approach:

$$\theta^i = \theta^i - \gamma^i \nabla L \left( \theta^i \right)$$

(4-5)

where $\gamma^i$ is the step-size in the $i$-th iteration used to update each parameter $\theta$ and the gradients of the loss function for each parameter in the model are as follows:

$$\nabla_{W_v} L = -2\alpha \left( X_v - \tilde{W}_v W_v X_v \right) X_v^T W_v^T + 2\beta \tilde{W}_v$$

(4-6)

$$\nabla_{W_t} L = -2\alpha \tilde{W}_v \left( X_v - \tilde{W}_v W_v X_v \right) X_v^T - 2\delta \tilde{W}_v \left( X_t - \tilde{W}_t W_v X_v \right) X_v^T + 2\beta W_v$$

(4-7)

$$\nabla_{\tilde{W}_v} L = -2(1 - \alpha) \left( X_t - \tilde{W}_t W_t X_t \right) X_t^T W_t^T - 2\delta \left( X_t - \tilde{W}_t W_v X_v \right) X_v^T W_v^T + 2\beta \tilde{W}_t$$

(4-8)

$$\nabla_{\tilde{W}_t} L = -2(1 - \alpha) \tilde{W}_t \left( X_t - \tilde{W}_t W_t X_t \right) X_t^T + 2\beta W_t$$

(4-9)

**Online formulation**

The previous subsection presents a strategy to find the coding and decoding matrices by using gradient descent technique. Unfortunately, this strategy by itself is not suitable for large scale data sets, due to its formulation have high memory requirements, since all training samples in the dataset are required in each iteration. For this reason, we want to extend this problem as an online learning approach based on stochastic approximations. The main
idea of online learning based on a stochastic approximation is to update the solution using a single training sample at a time. In this way, we can scan the whole dataset with low memory requirements. Following this approach, the final updating rules only depend on the $i$-th sample $(x_i^v, x_i^t)$: visual and text features for the $i$-th image) and the corresponding gradient functions are as follows.

\[ \nabla \tilde{W}^i_L = -2\alpha \left( x_i^v - \tilde{W}_v^i W_v^i x_i^v \right) (x_i^v)^T (W_v^i)^T + 2\beta \tilde{W}_v^i \]  
\[ \nabla W_v^i L^i = -2\alpha \tilde{W}_v^i \left( x_i^v - \tilde{W}_v^i W_v^i x_i^v \right) (x_i^v)^T - 2\delta \tilde{W}_v^i \left( x_i^v - \tilde{W}_v^i W_v^i x_i^v \right) (x_i^v)^T + 2\beta W_v^i \]  
\[ \nabla \tilde{W}^i L^i = -2(1 - \alpha) \left( x_i^t - \tilde{W}_t^i W_t^i x_t \right) x_i^T W_t^T - 2\delta \left( x_i^t - \tilde{W}_t^i W_t^i x_t \right) (x_i^v)^T W_v^T + 2\beta \tilde{W}_t^i \]  
\[ \nabla W_v^i L^i = -2(1 - \alpha) \tilde{W}_v^i \left( x_i^t - \tilde{W}_t^i W_t^i x_t \right) (x_i^t)^T + 2\beta W_t^i \]  

where $x_i^v$ and $x_i^t$ are vectors of features and label representation, respectively, for one instance. But also, this method can be generalized by using several samples grouped in mini-batches, this helps to a faster execution and numerical stability [34].

**Adaptive step-size**

A potential problem with gradient descent is to get stuck in a local minima, we can alleviate this problem by the inclusion of a momentum term [108]. The main idea about using momentum is to stabilize the parameter change by making non-radical updates using a combination of the previous update and the gradient. So in this way the original update term:

\[ \Delta W^i = -\gamma^i \nabla W L \left( \theta^i \right) \]  

take the form:

\[ \Delta W^i = -\gamma^i \nabla W L \left( \theta^i \right) + p \Delta W^{(i-1)} \]  

where $p$ is the momentum parameter which tries to preserve a portion of the previous update.

**Online learning algorithm**

The final algorithm for learning process (Algorithm 1) is as follows: starts by a random initialization of the transformation matrices, and for each iteration a mini-batch of instances with its corresponding features and label representation are randomly sampled from the
training set, then, the gradients of the lost function are calculated for each transformation matrix (the gradient of the lost functions is calculated by taking into account only the current observations), and the new transformation matrices are calculated by using the update terms based on momentum. Finally, the algorithm ends when a predefined maximum number of epochs is reached.

**Prediction**

Once the parameters have been learned (coding and decoding matrices) we can use this model to predict the label representation \( \tilde{x}_t \) from the feature representation \( x_v \) of a new unannotated sample, as follows:

\[
\tilde{x}_t = \tilde{W}_tW_v x_v
\]  

(4-18)

The transformation of the input features generates an \( m \)-dimensional vector with a smoothed label representation, which can be interpreted as a probability distribution which denotes the probability that the \( j \)-th label is assigned to an instance. The final decision to assign a label would be taken by defining a threshold, so we assign 1 to the \( j \)-th label if \( \tilde{x}_{t,j} \geq \text{threshold} \), or we can assign 1 to the top-\( k \) labels with the highest values in the vector.

**Implementation details**

We used the Pylearn2 library [48] to implement and perform the training of our method. This is a machine learning research library built on top of Theano [14] that facilitates the use of the GPU in a transparent way. Their emphasis on modularity allow us the reuse of code components and there is almost no restrictions on their use. Furthermore, it provides a way of specifying all parameters for a specific and complete experiment without exposing any specific implementation details. It can be done by using the YAML language. Two of the main advantages of using Theano and pylearn2 are: first, it allows to specify our models symbolically and the library optimizes the code for both CPU and GPU. Second, that we can change the objective function anytime we want and compute the gradients in an easy way.

Due to these facilities, this is a convenient library to test our method, mainly, due to the improvement in resource management in GPU and CPU, but also, to the fact that our method is trained with gradient descent algorithm. This help us to test our method in a large scale context.

As mention above, we use the library pylearn2 to take advantage of the computation and use of resources using a GPU. Table 4-1 shows the total execution time for some parameter configurations using the GPU and the CPU. The reported time includes loading time for the dataset, training time and evaluation of the performance with f-score measure.
Algorithm 1: Two-way multimodal online matrix factorization algorithm for learning state:

1 **input** \( r \): latent space size, \( \gamma^0 \): initial step size, \( \text{epochs} \): number of epochs, \( X_v \in \mathbb{R}^{n \times l} \), \( \alpha, \delta, \beta \)

2 **Random initialization of transformation matrices:**

3 \( \tilde{W}_v^0 = \text{random\_matrix} (r, n) \)

4 \( W_v^0 = \text{random\_matrix} (n, r) \)

5 \( \tilde{W}_t^0 = \text{random\_matrix} (r, m) \)

6 \( W_t^0 = \text{random\_matrix} (m, r) \)

7 **for** \( k = 1 \) **to** \( \text{epochs} \) **do**

8 **for** \( j = 1 \) **to** \( l \) **do**

9 \( i = k \times j \)

10 \( x_v^i, x_t^i \leftarrow \text{sample\_without\_replacement}(X_v, X_t) \)

11 **Compute gradients:**

12 \( g_{\tilde{W}_v}^i = \nabla_{\tilde{W}_v} L (x_v^i, x_t^i, W_v^i, \tilde{W}_v^i, W_t^i, \tilde{W}_t^i) \)

13 \( g_{W_v}^i = \nabla_{W_v} L (x_v^i, x_t^i, W_v^i, \tilde{W}_v^i, W_t^i, \tilde{W}_t^i) \)

14 \( g_{\tilde{W}_t}^i = \nabla_{\tilde{W}_t} L (x_v^i, x_t^i, W_v^i, \tilde{W}_v^i, W_t^i, \tilde{W}_t^i) \)

15 \( g_{W_t}^i = \nabla_{W_t} L (x_v^i, x_t^i, W_v^i, \tilde{W}_v^i, W_t^i, \tilde{W}_t^i) \)

16 **Update term calculation using momentum:**

17 \( \Delta \tilde{W}_v^i = -\gamma g_{\tilde{W}_v}^i + \alpha \Delta \tilde{W}_v^{(i-1)} \)

18 \( \Delta W_v^i = -\gamma g_{W_v}^i + \alpha \Delta W_v^{(i-1)} \)

19 \( \Delta \tilde{W}_t^i = -\gamma g_{\tilde{W}_t}^i + \alpha \Delta \tilde{W}_t^{(i-1)} \)

20 \( \Delta W_t^i = -\gamma g_{W_t}^i + \alpha \Delta W_t^{(i-1)} \)

21 **Update transformation matrices:**

22 \( \tilde{W}_v^{(i+1)} = \tilde{W}_v^i + \Delta \tilde{W}_v^i \)

23 \( W_v^{(i+1)} = W_v^i + \Delta W_v^i \)

24 \( \tilde{W}_t^{(i+1)} = \tilde{W}_t^i + \Delta \tilde{W}_t^i \)

25 \( W_t^{(i+1)} = W_t^i + \Delta W_t^i \)

26 **end for**

27 **end for**

28 **return** \( \tilde{W}_v^N, W_v^N, \tilde{W}_t^N, W_t^N \)
Table 4-1: Execution time using GPU and CPU to run 120, 15 and 1 epochs using the library pylearn2. Execution time execution includes loading time for the dataset, training time and evaluation of the performance with f-score measure.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Epochs</th>
<th>120</th>
<th>15</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corel</td>
<td>GPU</td>
<td>0:19:40</td>
<td>0:01:15</td>
<td>0:00:45</td>
</tr>
<tr>
<td></td>
<td>CPU</td>
<td>0:42:47</td>
<td>0:02:43</td>
<td>0:00:47</td>
</tr>
<tr>
<td>Bibtex</td>
<td>GPU</td>
<td>0:40:56</td>
<td>0:01:40</td>
<td>0:01:22</td>
</tr>
<tr>
<td></td>
<td>CPU</td>
<td>2:08:47</td>
<td>0:06:39</td>
<td>0:01:45</td>
</tr>
<tr>
<td>MediaMill</td>
<td>GPU</td>
<td>0:54:58</td>
<td>0:07:06</td>
<td>0:03:21</td>
</tr>
<tr>
<td></td>
<td>CPU</td>
<td>4:33:19</td>
<td>0:18:18</td>
<td>0:04:20</td>
</tr>
</tbody>
</table>

The time reported shows that even when running few epochs, the total execution time is less using GPU than CPU. When running much more epochs and when the dataset gets bigger, the performance in time becomes much more significant. To perform the parameter exploration, this is very useful, due to the fact, that we have to explore around seven parameters to obtain the best results.

### 4.1.3 Experiments and results

The objective of this section is to evaluate the performance of the proposed algorithm in different multi-label annotation task. The performance of the proposed algorithm is compared with several baselines using 3 standard multi-label datasets with different sizes and different dimension for features representation.

#### Experimental setup

In order to compare our method, we used the same experimental setup as in [94], i.e. we use 80% of the images for training and the remaining 20% for test. Results were compared against 8 MLLSE algorithms (OVA, CCA, CS, PLST, MME, ANMF, MNMF, OMMF). The proposed method has a set of parameters that impact the quality of the resulting model. These parameters were experimentally tuned by using a random 5-fold cross validation in the training set. We have two parameters that control the importance of the two different modalities in our method and a third parameter that controls the relative importance of the regularization terms. These first two parameters are $\alpha$ and $\delta$. The parameter $\alpha$ controls the relative importance of the modalities in an independent way. It showed to have low values for the visual modality and high values for the textual modality. The parameter $\delta$ controls the relative importance of the mapping between instance features and label information and it showed to have high values. This setup, shows how the annotation task is favored, by one hand, giving more importance to the textual modality (label representation) and second, by
Table 4-2: Selected datasets to evaluate our method. The characteristics described in the table are: total number of possible labels (Labels), features dimensionality (Features), average number of labels per instance (Label cardinality) and total number of instances in the dataset (Examples).

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Corel5k</th>
<th>Bibtex</th>
<th>MediaMill</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labels</td>
<td>374</td>
<td>159</td>
<td>101</td>
</tr>
<tr>
<td>Features</td>
<td>500</td>
<td>1,836</td>
<td>120</td>
</tr>
<tr>
<td>Label cardinality</td>
<td>3,522</td>
<td>2,402</td>
<td>4,376</td>
</tr>
<tr>
<td>Examples</td>
<td>5,000</td>
<td>7,395</td>
<td>43,907</td>
</tr>
</tbody>
</table>

imposing a strong independence between the modalities.

Datasets

We wanted to evaluate our method in annotation tasks, due to this, we considered three standard multi-label and publicly available datasets with different sizes (Corel5k, Bibtex and MediaMill) that have been used in previous works using F1 score to evaluate the tagger performance. The datasets are distributed by the Mulan framework authors [132]. Table 4-2 summarizes the main characteristics of these datasets.

Corel 5k is widely used in keyword based image retrieval and image annotation tasks. It contains around 5000 images manually annotated with 1 to 5 keywords. A standard set of 499 images are used as test, and the rest is used for training. The vocabulary contains 374 words.

Bibtex contains 7395 bibtex entries that have been tagged by users of a social network using 159 tags. Each bibtex entry contains a small set of textual elements representing the author, the title, and the conference or journal name. The text is represented as bag-of-words, with a feature space with dimensionality equal to 1836.

MediaMill consists of patterns about multimedia files. It dataset includes 43907 sub-shots with 101 classes, where each image is characterized by a 120-dimensional vector.

Annotation performance

We used a threshold strategy to evaluate the performance of our method in the same way as is described in [94]. This is, we assign 1 to the label $j$ of the instance $x_n$ if $x_{nj} > \text{threshold}$. We evaluated the performance of our method in each one of the datasets, calculating the F-Measure. Table 4-3 shows the results for each baseline method and the dimension of the embedding space. In Corel5k and MediaMill datasets, we got the best results in comparison with the other methods and in Bibtex we got a competitive result, being surpassed only by OMMF method.
Table 4-3: Performance in F-Measure for each method. The best performance for each dataset, is presented in bold. values in parentheses are the dimension of the generated embedding space.

<table>
<thead>
<tr>
<th>Method</th>
<th>Corel5k</th>
<th>Bibtex</th>
<th>MediaMill</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVA</td>
<td>0.112</td>
<td>0.372</td>
<td>--</td>
</tr>
<tr>
<td>CCA</td>
<td>0.150</td>
<td>0.404</td>
<td>--</td>
</tr>
<tr>
<td>CS</td>
<td>0.086 (50)</td>
<td>0.332 (50)</td>
<td>--</td>
</tr>
<tr>
<td>PLST</td>
<td>0.074 (50)</td>
<td>0.283 (50)</td>
<td>--</td>
</tr>
<tr>
<td>MME</td>
<td>0.178 (50)</td>
<td>0.403 (50)</td>
<td>0.199 (350)</td>
</tr>
<tr>
<td>ANMF</td>
<td>0.210 (30)</td>
<td>0.297 (140)</td>
<td>0.496 (350)</td>
</tr>
<tr>
<td>MNMF</td>
<td>0.240 (35)</td>
<td>0.376 (140)</td>
<td>0.510 (350)</td>
</tr>
<tr>
<td>OMMF</td>
<td>0.263 (40)</td>
<td>0.436 (140)</td>
<td>0.503 (350)</td>
</tr>
<tr>
<td><strong>Our Method</strong></td>
<td><strong>0.283 (100)</strong></td>
<td><strong>0.422 (300)</strong></td>
<td><strong>0.540 (300)</strong></td>
</tr>
</tbody>
</table>

4.2 Semi-supervised dimensionality reduction via multimodal matrix factorization

4.2.1 Introduction

Unsupervised dimensionality reduction is mainly used with the aim of exploring the data structure and extracting meaningful information from data without any prior information. In contrast, in supervised dimensionality reduction specific targets (labeled instances) of interest are used to guide the process of dimensionality reduction. Even though supervised approaches can exploit the labeled data in order to improve classification performance, they require every training instance to be labeled. But a proper annotation of a whole dataset is an arduous process, and for large-scale real-world collections is infeasible to ensure a reliable annotation for each instance. So, in many cases we are in a situation where we have a big quantity of potential data for training our algorithms but only a small fraction with annotations can be used. Even so, non annotated data present valuable information about the manifold structure of the data that should be exploited in some way. This section presents a semi-supervised dimensionality reduction method based on matrix factorization that can be used in training datasets that are not fully annotated by using the information from annotated instances to preserve the separability between elements from different classes, but also using the non-annotated elements to estimate the intrinsic manifold structure of the data.
4.2 Semi-supervised dimensionality reduction via multimodal matrix factorization

4.2.2 Related work

There are a high number of linear techniques that perform dimensionality reduction by embedding the data to a lower semantic space, among the unsupervised approaches stand out principal component analysis (PCA) [63], factor analysis (FA) and independent component analysis (ICA) [123]. Other approaches like locality preserving projection (LPP) [93] and neighborhood preserving embedding (NPE) [53] try to preserve the local neighborhood structure. Some dimensionality reduction techniques can take into account domain knowledge. This domain knowledge can be expressed in different forms, such as, class labels, pairwise constraints or another kind of prior information. Fisher’s linear discriminant analysis (LDA) [39] was one of the first techniques to take advantage of class observation to preserve the separability of the original classes. Also, there are semi-supervised alternatives that learn from a combination of both labeled and unlabeled data. For instance, semi-supervised discriminant analysis (SDA) [20] and the soft label based linear discriminant analysis SL-LDA [157] use the labeled data to maximize the separability between classes and uses the unlabeled data to estimate the intrinsic manifold structure of the data. Also, there are some non-linear alternatives (isometric feature mapping [126], locally linear embedding [107] and Laplacian Eigenmaps [10], among others). Unfortunately the modeling of these non-linearities leads to high computational complexities that make them prohibitive to use in large-scale collections. The method introduced in this section, presents two characteristics that make it highly scalable: first, it is based on linear transformations, and second, its algorithm is formulated as an online-learning approach, which only needs to keep small portions of the training data in main memory and requires little time to reach a predefined expected risk.

4.2.3 Semi-supervised two-way multimodal online matrix factorization

In this section, we are interested in scenarios where we have a large number of instances for training \( k \) instances), but only a restricted \( l \) number of them are properly labeled. For this purpose, we present a reformulation of the TWOMF (Two-way Multimodal Online Matrix Factorization) framework (see Equation 4-19). This reformulation that takes advantage of both annotated and non-annotated instances presents the following loss function:

\[
L = \alpha \sum_{i=1}^{k} \left\| x^i_v - \tilde{W}_v W_v x^i_v \right\|_F^2 + (1 - \alpha) \sum_{i=1}^{l} \left\| x^i_t - \tilde{W}_t W_t x^i_t \right\|_F^2 + \delta \sum_{i=1}^{l} \left\| x^i_t - \tilde{W}_t W_v x^i_v \right\|_F^2 + \beta \left( \left\| W_v \right\|_F^2 + \left\| \tilde{W}_v \right\|_F^2 + \left\| W_t \right\|_F^2 + \left\| \tilde{W}_t \right\|_F^2 \right) \tag{4-19}
\]

where, \( x^i_v \) is the feature vector of the \( i \)-th instance in the data collection \( X_v \in \mathbb{R}^{n \times l} \) and \( x^i_t \) is the corresponding binary label vector, \( \alpha \) controls the relative importance between the reconstruction of the instance representation and the label representation, \( \delta \) controls the
4 Multimodal matrix factorization for dimensionality reduction

Table 4-4: Dataset information and data partition for each dataset

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Original dataset partitions</th>
<th>Low-scale partitions</th>
<th>Large-scale evaluation</th>
<th>#Dim</th>
<th>#Class</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Train Test</td>
<td>Train Test</td>
<td>Train Test</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Covtype</td>
<td>581012</td>
<td>8000 8000</td>
<td>100000 2000</td>
<td>54</td>
<td>7</td>
</tr>
<tr>
<td>MNIST</td>
<td>60000 10000</td>
<td>8000 8000</td>
<td>60000 10000</td>
<td>784</td>
<td>10</td>
</tr>
<tr>
<td>Letters</td>
<td>20000</td>
<td>8000 8000</td>
<td>–</td>
<td>16</td>
<td>26</td>
</tr>
<tr>
<td>USPS</td>
<td>4649 4649</td>
<td>4649 4649</td>
<td>–</td>
<td>256</td>
<td>10</td>
</tr>
</tbody>
</table>

relative importance of the mapping between instance features and label information and $\beta$ controls the relative importance of the regularization terms. The first term in the loss function uses all the instances to model the low semantic space and the second and third terms use only the annotated instates to model the semantic space and the mapping between features and label information. The final algorithm uses stochastic gradient descent learning [17], by updating the transformation matrices at each iteration with a mini-batch of instances with their corresponding features and label representation that are randomly sampled from the training set, due to the fact that samples in a minibatch are discarded after the minibatch is processed, it is possible to scan large datasets without memory restrictions. The algorithm ends when a predefined maximum number of epochs is reached. Once the learning process is completed, the projection to the low-rank semantic representation can be performed by multiplying the original high-dimensional feature representation by the coding $W_v$ matrix.

$$h_i = W_v x_i$$  \hspace{1cm} (4-20)

4.2.4 Experiments and results

In this section, we evaluate our algorithm in comparison with several widely-used datasets for dimensionality reduction, manifold learning and classification tasks (the details of each dataset are shown in Table 4-4). We evaluate the performance of our algorithm by calculating classification accuracy in each one of these datasets. We compare our method with other linear supervised, semi-supervised and unsupervised dimensionality reduction methods. These methods include SVM (Support Vector Machines) with a linear kernel [31], LDA [39], SRDA (spectral regression discriminant analysis) [21], SDA [20] and PCA [63]. For determining the parameters of each method, we perform an exploration by using 5-fold cross-validation. For our method, we need to determine five parameters, including, the learning rate, the mini-batch size and the $\alpha$, $\beta$ and $\delta$ parameters present in the cost function. For all algorithms, except for the supervised, i.e, SVM, LDA and SRDA, we use the projected training set to construct a nearest neighborhood classifier (1NN) for evaluating the classification accuracy of the projected test set, in a similar setup as in [157]. In this evaluation, we
explore the performance for different percentage of randomly selected annotated instances in training set. Table 4-5 reports the average accuracies for 10 runs in each configuration in the four datasets using the low-scale partitions (see Table 4-4). As we can see, the STWOMF presents competitive results in comparison with all other algorithms when the dimensionality of the semantic representation coincides with the number of classes (r=C). Furthermore, when the dimensionality increases (r=C+10), STWOMF over performs the other algorithms (in our experiments, a further increase of the dimensionality did not contribute to improve the performance of the algorithm).

An evaluation with the two largest datasets using different sizes of training set was performed in order to verify the capability of the proposed method to deal with large-scale collections. Figure 4-1 presents the average classification accuracies and times for different sizes of the training set (the reported results are the average of 10 runs for each configuration). The STWOMF is compared against the SDA which is another semi-supervised method that also uses the unlabeled data to estimate the manifold structure of the data. For all training sizes only 30% of instances are annotated, so we can see that both methods are able to learn from labeled and unlabeled instances and both can improve their performance as more training instances are available. However, STWOMF presents two advantages: first, unlike SDA, in STWOMF we can increase the dimensionality of the semantic space resulting in an improvement in the performance. For instance, in the MNIST dataset, the STWOMF using 17 latent factor (STWOMF-r17) presents a gain in accuracy of about 6 points over the same STWOMF using only 7 latent factor (STWOMF-r7) and the SDA; and second, STWOMF presents a little increase in the time required for training as more training instances are used, leading to a speedup of about 3.5x-7x over SDA in MNIST and about 8x in CovType. The main reason for the short time used in training phase by STWOMF is that, thanks to its online formulation for large datasets, a few number of epochs are required until the algorithm converges (convergence in all algorithms is verified by means of a minimum threshold required to improve the reconstruction error in each epoch). In fact, for both datasets MNIST and CovType only two epochs are required to achieve convergence.

### 4.2.5 Conclusions and future Work

This chapter presented an approach for dimensionality reduction that takes advantage of annotated data to model a semantic low-space representation that preserves the separability of the original classes. Furthermore, this method has the ability to exploit unlabeled instances for modeling the manifold structure of the data and use it to improve its performance in classification and multi-label annotation.

The experimental evaluation shows that the proposed method presents competitive results in terms of classification accuracy in comparison with several unsupervised, semi-supervised and supervised linear dimensionality reduction methods.

An important characteristic of this method is that, unlike classical matrix factorization...
Table 4.5: Classification accuracy for different percentages of annotated instances in training set using low-scale partitions. Reported results are the average of 10 runs for each configuration (r = number of latent factors, C = number of classes in the dataset).

<table>
<thead>
<tr>
<th>METHOD</th>
<th>COVTYPE</th>
<th></th>
<th>MNIST</th>
<th></th>
<th>LETTERS</th>
<th></th>
<th>USPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCA</td>
<td>0.963±0.007 0</td>
<td>0.869±0.039 0</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
</tr>
<tr>
<td>STWOMF</td>
<td>0.963±0.007 0</td>
<td>0.869±0.039 0</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
</tr>
<tr>
<td>SDA</td>
<td>0.869±0.039 0</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
</tr>
<tr>
<td>LDA</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRDA</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCA</td>
<td>0.963±0.007 0</td>
<td>0.869±0.039 0</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
</tr>
<tr>
<td>STWOMF</td>
<td>0.963±0.007 0</td>
<td>0.869±0.039 0</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
</tr>
<tr>
<td>SDA</td>
<td>0.869±0.039 0</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
</tr>
<tr>
<td>LDA</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRDA</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCA</td>
<td>0.963±0.007 0</td>
<td>0.869±0.039 0</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
</tr>
<tr>
<td>STWOMF</td>
<td>0.963±0.007 0</td>
<td>0.869±0.039 0</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
</tr>
<tr>
<td>SDA</td>
<td>0.869±0.039 0</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
</tr>
<tr>
<td>LDA</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRDA</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCA</td>
<td>0.963±0.007 0</td>
<td>0.869±0.039 0</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
</tr>
<tr>
<td>STWOMF</td>
<td>0.963±0.007 0</td>
<td>0.869±0.039 0</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
</tr>
<tr>
<td>SDA</td>
<td>0.869±0.039 0</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
</tr>
<tr>
<td>LDA</td>
<td>0.916±0.012 0</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td>0.960±0.003 0</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRDA</td>
<td>0.922±0.001 0</td>
<td>0.963±0.007 0</td>
<td>0.916±0.012 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.2 Semi-supervised dimensionality reduction via multimodal matrix factorization

Figure 4-1: Average classification accuracy (top) and average required time for training (bottom) in MNIST (left) and CovType (right) datasets using different number of training instances. For all training sizes only 30% of instances are annotated.
methods, such as the method proposed by Otalora-Montenegro et al. [94] based on OMMF, the semantic representation is not constructed explicitly but a direct mapping to the semantic space is learned. This allows its formulation as an online learning algorithm, that makes it capable to deal with large collections of data by achieving a significant reduction in memory requirements and computational load. Another difference with classical matrix factorization methods, is that the transformation from the semantic representation to the label space is learned directly in training phase, making the annotation process very simple, requiring a simple multiplication by a transformation matrix.

A major limitation of this method as well as the other multi-label latent space embedding methods is that it is a linear model which imposes significant restrictions that limit its flexibility. Therefore, it would be interesting to explore non-linear alternatives which allow modeling more complex relationships that could improve the performance in annotation task.
5 Transductive non-linear semantic embedding for multi-class classification

In chapter 3, a semantic embedding method based on matrix factorization was presented. This method has interesting characteristics that allow it to deal with large-scale datasets and take extra information for a second auxiliary information modality in supervised and semi-supervised setups. Unfortunately, this is a linear method that imposes significant restrictions that limit its ability to accurately modeling the true nature of the data. Therefore, in this chapter, a non-linear alternative has been explored, this alternative is based on kernel methods that have been shown to be very effective to solve non-linear problems such as classification, annotation, and clustering among others.

The work presented in this chapter has been submitted to Pattern Recognition Letters.

5.1 Introduction

The exponential growth of multimedia data has attracted considerable attention in the research communities in data analysis [28], since this kind of data collections has become a valuable source of knowledge, by presenting several potential applications across many diverse areas. Supervised classification algorithms are one of the most popular machine learning techniques, however they require every training instance to be labeled. A proper annotation of a whole dataset is an arduous process, and for large-scale real-world collections is infeasible to ensure a reliable annotation for each instance. So, in many cases, we are in a situation where we have a big quantity of potential data for training our algorithms, but only a small fraction of properly annotated instances can be used. Even so, non-annotated evidence would present valuable information about the manifold structure of the data that should be exploited in some way.

This chapter presents a semi-supervised dimensionality reduction method based on matrix factorization that can be used in training datasets that are not fully annotated by using the information from annotated instances to preserve the separability between elements from different classes, but also by taking advantage of the non-annotated elements to estimate the intrinsic manifold structure of the data. In order to model the inherent non-linearity of the data, a kernel-based matrix factorization solution is proposed. Kernel extensions of classical matrix factorization methods such as kernel matrix factorization (KMF) [151], kernel PCA
(KPCA) [111] and kernel SVD (KSVD) [114], have demonstrated their capability extracting non-linear patterns, which is translated into a better performance compared to their linear counterparts. The drawback of such kernel methods is their high computational and space cost, required to compute the kernel matrix (Gram matrix) that is quadratic in terms of the number of examples. This leads to the impossibility of directly using these methods when the number of samples is large. The proposed semi-supervised online kernel matrix factorization algorithm (SS-OKMF) is able to handle the matrix factorization problem in a kernel-induced feature space in large-scale data sets, under a reasonable amount of time and storage resources.

SS-OKMF addresses the memory problem imposing a budget restriction [118, 95], this is, restricting the number of samples needed to represent the basis of the feature space. With respect to the computation time, SS-OKMF uses a stochastic gradient descent (SGD) strategy for optimizing its cost function by guaranteeing a faster convergence. The main contributions of this work are as follows:

1. It proposes a non-linear kernel based method that constructs a semantic embedded space modeled by the original feature representation and the class representation. This semantic representation can be used for automatic classification in two ways: 1) by finding a direct mapping to the label representation or 2) by exploiting the geometry of the semantic space and looking for the nearest annotated instances.

2. The proposed method is capable of taking advantage of unlabeled instances under a transductive setup in classification task, by using the annotated instances to maximize the discrimination between classes, but also, the non-annotated instances to better estimate the intrinsic manifold structure of the data.

3. In order to make the proposed method feasible for a large-scale setup, two strategies have been used in its formulation: first, a learning in a budget restriction is proposed to preserve low memory requirements, and second, an online learning strategy is used to reduce the computation time and keep low computational requirements.

4. Extensive experiments were carried out and the results showed that the approach presents a competitive performance in classification tasks under transductive as well as inductive learning setups while keeping a low computational cost.

5.2 Related work

In most of the cases, kernel matrix factorization methods extend the classical linear matrix factorization methods with kernels in order to achieve a factorization able to extract non-linear patterns. The work of Zhang et al. [151] is one of the first attempts that present a kernel based matrix factorization, in this case, by extending the original non-negative matrix
5.2 Related work

 factorization (NMF). This approach uses the kernel trick [55] to extend linear methods to work in a high-dimensional space, called feature space, without calculating an explicit mapping to that space. Some works [139, 95] have been based on the Convex-NMF [37], which presents an immediate and natural generalization to kernels. In a similar way, kernelized versions for other classical matrix factorization algorithms like principal component analysis (KPCA) [111], and singular value decomposition (KSVD) [114] have been proposed.

In a similar way, kernels have been employed in other dimensionality reduction strategies that are supported in non based matrix factorization learning approaches but attempt to model the manifold structure of the data. For instance, Wang et al. [136] proposed a supervised learning method for dimensionality reduction called kernel maximum margin projection (KMMP), which is a kernel generalization of the maximum margin projection (MMP) that discovers the local manifold structure by maximizing the margin between positive and negative examples at each local neighborhood. Ham et al. [51] presented an interesting result showing that three well known manifold learning algorithms: Isomap, graph Laplacian eigenmaps, and LLE, can be interpreted as KPCA with the adequate kernel matrix construction for each method.

All methods presented above are proposed for an unsupervised approach, however, some alternatives also have been proposed for supervised learning. For instance, several approaches based on kernelized versions of canonical correlation analysis (KCCA) have been evaluated [122, 60]. And more recently, Gonen et al. [44] proposed the kernelized Bayesian matrix factorization method that is able to integrate multiple modality sources by coupling matrix factorization with multiple kernel learning, and Zitnik et al. [162] presented a data fusion method based on constrained matrix factorization to reveal hidden associations.

Lately, more recent works that perform dimensionality reduction with some non-linear setup go for a semi-supervised learning approach (SSL) due to the fact that in many cases obtaining unlabeled instances for training is easy but a proper annotation for these instances can carry challenge and expensive processes. Most of semi-supervised strategies are based on label propagation, and process the training data as a graph. For instance, Kim et al. [67], proposed the minimax label propagation (MMLP) that propagates labels through only few important paths called “minimax paths”. Where the main idea of minimax paths emerge as a minimal set of paths lying in high-density regions. A more recent work, the label prediction via DGL (LPDGL) [45] propose the deformed graph laplacian (DGL) which is an extension of the classical Graph Laplacian [92] that has been widely exploited in traditional SSL algorithms, by defining a novel smoothness term. This strategy is linear but has a non-linear extension based on non-linear kernels [57]. We used the results reported in these two last works.

Kernel-based methods are a good alternative for modeling non-linearities, unfortunately, this kind of methods present and important drawback related with its high computational complexity. For instance, the kernelized version of CCA (KCCA) used by Yuncho et al. [122] presents a cubic computational complexity in the number of images in the dataset, which
makes it infeasible for large scale problems.
In this chapter we propose a kernel based method that minimize the computational and memory complexities by imposing a budget restriction, this restriction along with an online formulation based on stochastic gradient descent to reduce the training time and to keep low computational requirements in large-scale problems. A remarkable difference of the proposed method with respect to the matrix factorization method presented by [95] is the semi-supervised formulation that takes advantage of annotated instances to maximize the discrimination between classes and at the same time estimates the intrinsic manifold structure of the unlabeled data for multi-class classification. In addition, the new algorithm is able to predict not only a rich semantic space but also the direct class annotation for new instances. In contrast with other methods, discussed above, the proposed method addresses three different challenges simultaneously: non-linear modeling (using kernel methods); semisupervised dimensionality reduction and manifold learning (through semisupervised matrix factorization); and efficient learning and scalability (through on-line learning and learning in a budget).

5.3 Semi-supervised online kernel matrix factorization

Matrix factorization
For a collection represented by a matrix \( X \in \mathbb{R}^{n \times l} \), we can find a low level semantic representation by factorizing this original matrix as a product of two smaller matrices: \( F \in \mathbb{R}^{n \times r} \) that is known as the basis matrix and \( H \in \mathbb{R}^{r \times l} \) that is known as the encoding matrix, with the restriction of \( r \ll n \) [74]:

\[
X \approx FH
\]  

(5-1)

This projection performs a linear transformation from an original high-dimensional input representation to a low-dimensional semantic representation.

In this chapter we use kernel methods to generalize the basis matrix \( F \) to model non-linear mappings by translating the factorization problem to the features space induced by a kernel function (a kernel function \( k : x \times x \to \mathbb{R} \) induces a mapping \( \Phi : \mathcal{X} \to \mathcal{F} \) from the input space, \( \chi = \mathbb{R}^n \), which is the original representation represented by \( X \), to a feature space, \( \mathcal{F} = \mathbb{R}^p \), were samples may be linearly separable). This modification would be directly translated to: \( \Phi(X) \approx F_{\phi}H \). Unfortunately, the calculation of this factorization is infeasible due to \( F_{\phi} \) depends explicitly on the mapping function \( \phi(\cdot) \) (this implies a mapping to a very highly-dimensional space or even to an infinite-dimensional space). Therefore, instead of calculate directly \( F_{\phi} \), we impose the restriction that the column vectors of \( F_{\phi} \) lie within the space of \( \Phi(X) \), this is, \( F_{\phi} \) is composed by linear combinations of the \( X \) points in the feature space (\( F_{\phi} = \Phi(X)W_x \)).
5.3 Semi-supervised online kernel matrix factorization

\[ \Phi(X) \approx \Phi(X) W_x H \] (5-2)

This modification avoids the necessity of evaluating the data in the feature space, by requiring only the kernel value between the data points \( (K = \phi^T(X) \phi(X)) \), additionally, only a reduced number \( b \ll l \) of representative points are used to construct the basis matrix (we construct a budget kernel matrix \( B \in \mathbb{R}^{b \times l} \) instead of the full kernel matrix \( X \in \mathbb{R}^{l \times l} \)) [95]:

\[ \Phi(X) \approx \Phi(B) W_x H \] (5-3)

This mitigates the high computational cost of constructing the huge Gram Matrix (Kernel Matrix).

### 5.3.1 Semi-supervised and transductive formulation

The encoding matrix \( H \) presents the original data points in an \( r \) dimensional semantic space that models the manifold structure of the data. This space presents interesting properties that help to grouping similar data points using only its original feature representation, but additionally, in this work we want to take advantage of also the annotated instances that can help to preserve the separability between different classes. For this purpose, we want the construction of the matrix \( H \) to be influenced also by the class representation.

But in this work we are interested in the case when only a reduced number of \( k \) annotated instances is available (i.e., \( k \ll l \)). Therefore, we can construct a class representation \( Y \in \mathbb{R}^{m \times k} \), which is a One hot representation of the associated classes for the \( k \) annotated instances (the class of each data point is represented for an \( m \) dimensional vector, where \( m \) is total number of classes in the collection, and a 1 in the \( j \)-th position \( (1 \leq j \leq m) \) defines the membership of the instance in the \( j \)-th class). This leads to a second factorization problem defined by \( Y \simeq W_y H_k \), where \( W_y \in \mathbb{R}^{m \times r} \) is a weight matrix (basis matrix) for the class reconstruction and \( H_k \in \mathbb{R}^{r \times k} \) is a subset of the latent representation (as previously
described in Equation 5-3) composed only by the annotated instances. Therefore, the final goal is to find a common latent representation modeled by both, the input feature representation and the class representation of the annotated instances (See Figure 5-1). To this end, the previous conditions are putting together and the general problem is defined as an optimization problem by minimizing the following loss function:

\[
\min_{W_x, W_y, h} J(W_x, W_y, H) = \frac{\alpha}{2} \| \Phi(X) - \Phi(B) W_x H \|_F^2 + \frac{\beta}{2} \| Y - W_y H \|_F^2 + \\
\frac{\lambda_1}{2} \| W_x \|_F^2 + \frac{\lambda_2}{2} \| W_y \|_F^2 + \frac{\lambda_3}{2} \| H \|_F^2
\]  

(5-4)

where \(\alpha\) controls the relative importance of reconstructing the feature representation, \(\beta\) controls the relative importance of reconstructing the class representation, and \(\lambda_1, \lambda_2, \lambda_3\) control the relative importance of the regularization terms that penalizes big values for the weight matrices and avoid overfitting. This implies that in the learning process \(W_x\) and \(H\) are affected by both labeled and unlabeled instances, but \(W_y\) is updated only by the labeled ones.

### 5.3.2 Online formulation

The proposed method can be formulated as an online learning method by using stochastic gradient descent (SGD). The idea of online learning using stochastic approximations is to compute the new solution for each unknown in the problem using a single data sample at a time. Then, we can scan large data sets without memory restrictions. Thus, the loss function can be rewritten as follows:

\[
\min_{W_x, W_y, h_i} J_i(W_x, W_y, h_i) = \frac{\alpha}{2} \| \Phi(x_i) - \Phi(B) W_x h_i \|_F^2 + \frac{\beta}{2} \| y_i - W_y h_i \|_2^2 + \\
\frac{\lambda_1}{2} \| W_x \|_2^2 + \frac{\lambda_2}{2} \| W_y \|_2^2 + \frac{\lambda_3}{2} \| h_i \|_2^2
\]  

(5-5)

And the updating rule is reformulated in such a way that it only depends on the i-th sample. So, for the weight matrices the classic stochastic gradient updating rule is used: \(W_x^{i+1} = W_x^i - \gamma_x g_x^i\) and \(W_y^{i+1} = W_y^i - \gamma_y g_y^i\), where \(W^i\) is the value of \(W\) at the i-th iteration, \(\gamma\) is the learning rate, which controls how large of a step to take in the direction of the negative gradient, and \(g_x\) and \(g_y\) are the gradients of the loss function for \(W_x\) and \(W_y\) respectively, as follows:

\[
g_x^i = \frac{\partial J_i(W_x, W_y, h_i)}{\partial W_x} = \alpha k(B, x^i) (h^i)^T + \alpha k(B, B) W_x h_i (h_i)^T + \lambda_1 W_x
\]

(5-5)

\[
g_y^i = \frac{\partial J_i(W_x, W_y, h_i)}{\partial W_y} = \beta y_i (h_i)^T - \beta W_y h_i (h_i)^T + \lambda_2 W_y
\]

(5-6)
In Equation 5-5 we replaced $\Phi (B)^T \Phi (B)$ by the matrix $k (B, B) \in \mathbb{R}^{b \times b}$ defined as $k (B, B) = \left\{ k (b^i, b^j) \right\}_{i,j}$, where $b^j \in \mathbb{R}^{n}$ corresponds to the $j$-th column of $B$. In a similar way we can replace $\Phi (B)^T \Phi (x_i)$ by $k (B, x_i) \in \mathbb{R}^{b \times 1}$. This replacing leads to an important result, that is, that we can avoid computing the explicit mapping of the data into the feature space induced by the kernel function $k$ and use instead the kernel trick (compute the kernel in the input space directly). Finally, the updating rule for $h^i$ is a closed formula resulting from calculating the partial derivative of the loss function with respect $h^i$ and equal it to zero.

$$h^i = (\alpha W_x^T k (B, x^i) + \beta W_y^T y^i)^{-1} (\alpha W_x^T k (B, B) W_x + \beta W_y^T W_y + \lambda_3 I) \tag{5-7}$$

### 5.3.3 SS-OKMF algorithm

With the definition of the updating rules for all the model parameters, the general training algorithm for SS-OKMF (Algorithm 2), can be described as follows: starts by a random initialization of the weight matrices and each iteration consists in: first, calculating the latent representation from an observed pair of features and class representation randomly sampled and the weight matrices from the previous iteration, and second, updating the weight matrices by using the previously defined updating rules. For each iteration, a learning rate is calculated. In order to ensure convergence, it is defined as a decreasing function [16] that depends on the number of iterations and an initial step size $\gamma_0$. The final output of the algorithm are the weight matrices $W_x$ and $W_y$ that model the projection from the semantic space to the feature space and label space respectively. This algorithm can be generalized by using mini-batches (several grouped samples) at each iteration instead of using only one. This allows a faster convergence rate and also implies a better numerical stability [34].

### 5.3.4 Transductive learning

We can perform the training of our algorithm under a transductive setup in order to influence the final semantic representation with the feature representation of the target elements that we want to predict their class (unlabeled test instances), in this case the training $X$ matrix is an extended set composed with the staking of the feature representation of training set and test set ($X = [[X, X^u]]$). This helps to find a more suitable semantic space for the new elements.

### 5.3.5 Prediction

Once we have found the weight matrices by using Algorithm 2, we can use them to predict the corresponding semantic space for new unlabeled instances ($x^u$). This is done by using Equation 5-7 to calculate the corresponding semantic representation with $\beta = 0$ (without class information):
Algorithm 2: Semi-supervised Online Kernel Matrix Factorization

1. **input** $\gamma^0$: initial step size, $N$: number of iterations, $X$: feature representation of training dataset, $Y$: class representation of training dataset, $r$: dimension of latent space

2. Initialization of the weight matrices:

$$W^0_x = \text{random}_x\text{matrix}(b, r); \quad W^0_y = \text{random}_x\text{matrix}(n, r)$$

3. for $i = 1$ to $N$ do

4. New observation:

$$x^i, y^i \leftarrow \text{sample}_x\text{without}_x\text{replacement}(X, Y)$$

5. Compute gradients:

$$g^i_x = \frac{\partial J_i(W_x, W_y, h^i)}{\partial W_x}; \quad g^i_y = \frac{\partial J_i(W_x, W_y, h^i)}{\partial W_y}$$

6. Calculate learning rate:

$$\gamma^i = \frac{\gamma^0}{1 + \gamma^0 \lambda^i}$$

7. Update transformation matrices:

$$W^{(i+1)}_x = W^i_x - \gamma^i g^i_x; \quad W^{(i+1)}_y = W^i_y - \gamma^i g^i_y$$

8. Update semantic representation:

$$h^i = f(W_x, W_y, x^i, y^i)$$

9. end

10. for

11. return $W^{N+1}_x, W^{N+1}_y$

$$h^p = \alpha W^T_x k(B, x^u) (\alpha W^T_x k(B, B) W_x + \lambda_3 I)^{-1} \quad (5-8)$$

This semantic space can be used to predict the class for new unlabeled instances by projecting onto this space both unannotated and annotated data and perform a $k$-nearest-neighbor (KNN) classification. Furthermore, a direct classification can be performed by back-projecting this semantic representation to the class representation with $W_y$:

$$y^p = W_y h^p \quad (5-9)$$

where $y^p$ can be interpreted as a vector that defines a score or probability for each possible class. So we can assign the unlabeled instance to the class associated to the dimension with the maximum value in the predicted vector.

### 5.4 Experiments and results

#### 5.4.1 Experimental setup

We evaluate the performance of the SSOKMF method on different classification tasks under two main setups, semi-supervised classification and transductive classification, over several benchmark datasets. We use six real benchmark datasets by following the experimental
setup proposed by Chapelle et al. [26] with the aim of comparing our results with the results reported in [26],[67] and [45].

The list of datasets is composed of USPS, BCI, G241C, G241N, Digit1, and COIL2. All these datasets present data of different nature, G241C and G241N are artificial datasets were the data points are generated with a cluster assumption, the former is constructed by drawing points from two unit-variance isotropic Gaussians and the later was constructed to present a potentially misleading cluster structure. Digit1 is another artificial dataset that was constructed by artificially generating images of the digit '1' with different transformations including translations, rotations and the addition of artificial noise. USPS is a subset derived from the original US Postal (USPS) handwritten digit dataset by choosing only two classes (digits 2 and 5), BCI is a dataset of EEG signals recorded from a single person while he imagined movements with his right and left hands and COIL is a subset of the Columbia object image library that is a set of color images of 100 different objects taken from different angles at a resolution of 128 × 128 pixels. Table 5-3 presents the main characteristics for each data set. For further information about these datasets please refer to [26].

For all experiments, we obtained the parameters that reach the best performance for our method by using five-fold cross validation for a linear kernel as well as for an RBF kernel. For our method, we need to determine six hyperparameters, including, the learning rate ($\gamma^i$), the mini-batch size and the $\alpha$, $\beta$, $\lambda_1$, $\lambda_2$ and $\lambda_3$ hyperparameters present in the cost function. Additionally, in order to evaluate the behavior of our algorithm under a large-scale setup, we use the classical MNIST database. MNIST is a subset of a larger set of black and white (bilevel) images available from NIST composed by 60,000 training samples and 10,000 testing samples\(^1\) that belong to 10 classes, i.e., digits 0–9. The digits are size-normalized and centered in a fixed-size image. Each image consists of $p=28 \times 28=784$ gray-scale pixel intensities taken as the feature dimension. Additionally, we normalize the data to have the mean equal of zero and a standard deviation of one.

5.4.2 Transductive classification

We performed experiments to compare our method in a transductive setup by evaluating the error rate (i.e., the percentage of miss-classifications over all the dataset) of SS-OKMF against the results reported in [26], [67] and [45] in the six standard data sets described previously. For each dataset, there are two different setups: $l = 10$ and $l = 100$, where $l$ is the number of labeled instances, and the remaining instances compose the test set. Reported error rates are the mean values after 12 independent runs. In each run, the labeled and unlabeled examples are the same for all methods and the partitions are the standard (See detailed information about these data sets in [26]). We compare our method with some state-of-the-art algorithms. Two supervised: 1NN [25] and SVM[134], two unsupervised: MVU+1NN[140] and LEM+1NN[107], and fifteen semi-supervised: QC+CMN[161], Discrete

\(^1\)http://yann.lecun.com/exdb/mnist/
Reg[159], TSVM[61], SGT[62], Cluster-Kernel[27], Data Dep. Reg.[124], CHM[19], Self[49], LP[161], PVM[153], AGR[85], MMLP[67], LPDGL(Linear)[45] and LPDGL (Non-linear)[45]. Results are shown in table 5-1.

The last column presents the average performance rank in terms of error rate for each method over all datasets. For a single dataset, we assign rank 1 for the method with the lowest average test error on that dataset, then rank 2 for the method with the second lowest test error, and so on (the smaller the ranking score, the better the method has performed). As we can see, the best methods are semi-supervised ones, proving that we can get improvements in the performance by using only the structural information from unlabeled instances. Although our method does not have the best performance in all datasets, we have the majority of best results in comparison with the rest of the methods, demonstrating consistency in the results.

In the setup of $l=10$, our method achieved the best average rank and in the setup of $l=100$, we obtain in average the third place. This implies that our method is a good alternative in cases when the number of annotated instances is very limited.

In a similar way as the LPDGL method, we can define linear as well non-linear versions (in this evaluation the non-linear modeling is based on a radial basis function kernel), presenting in most of the cases a significant improve in the non-linear versions over the linear ones, especially in the $l=100$ partition. Even so, in some datasets, the difference is not significant or even the non-linear version performs worst. This could implicate that for these datasets a radial basis function kernel is not the best option and maybe another kind of function kernel could be more appropriate to model the singularities of the intrinsic structure of its data.

For each dataset, we explore the size of the budget and in most of the cases we obtain the same or even better results with a budget size smaller than 100% of the training instances, in fact, in most of the cases only the 50% or less of the total of instances were required to achieve the best performance. This result is very significant by taking into account that for this evaluation the dimensions of the datasets are quite small, which implies that only a small budget is required to correctly model this kind of problems, even though for the selection of the elements of the budget it has not been followed an approach more elaborate than a simple random sampling.

**Budget impact**

The application of a budget leads to our method to significantly reduce the algorithm complexity and to keep low computational requirements for training in large-scale collections. Two important aspects related with the budget constraint are the size of the budget (number of training instances contained in the budget) and the selection method to construct the budget. This aspects were explored with the aim of preserving or even improving the performance of the method, while, in the same time substantial reductions in memory consumption and computational time are achieved. The budget size selection was performed by exploring budget sizes from 5% up to 100% of all samples in a validation partition and
### Table 5-1: Error rate for 6 benchmark datasets in the transductive setup for different methods in the state of the art for $l = 10$ labeled samples. The three best results for each dataset are marked in red/bold (1), blue/italic (2) and green/underline (3), respectively. (Avg. Rank: Average of all the rankings obtained in each dataset)

<table>
<thead>
<tr>
<th>Datasets</th>
<th>USPS</th>
<th>BCI</th>
<th>g241c</th>
<th>g241d</th>
<th>Digit1</th>
<th>COIL</th>
<th>Avg. Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labeled Samples: 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1NN</td>
<td>19.82</td>
<td>48.74</td>
<td>44.05</td>
<td>43.22</td>
<td>23.47</td>
<td>65.91</td>
<td>13.33</td>
</tr>
<tr>
<td>SVM</td>
<td>20.03</td>
<td>49.85</td>
<td>47.32</td>
<td>46.66</td>
<td>30.6</td>
<td>68.36</td>
<td>17.67</td>
</tr>
<tr>
<td>MVU+1NN</td>
<td>14.88</td>
<td>50.24</td>
<td>48.68</td>
<td>47.28</td>
<td>11.92</td>
<td>62.72</td>
<td>14.33</td>
</tr>
<tr>
<td>LEM+1NN</td>
<td>19.14</td>
<td>49.94</td>
<td>47.47</td>
<td>45.34</td>
<td>12.04</td>
<td>65.96</td>
<td>14.67</td>
</tr>
<tr>
<td>QC+CMN</td>
<td>13,61</td>
<td>50.36</td>
<td>39.96</td>
<td>46.55</td>
<td>9.8</td>
<td>59.63</td>
<td>10.17</td>
</tr>
<tr>
<td>Discrete Reg</td>
<td>16.07</td>
<td>49.51</td>
<td>49.59</td>
<td>49.05</td>
<td>12.64</td>
<td>63.38</td>
<td>14.5</td>
</tr>
<tr>
<td>TSVM</td>
<td>25.2</td>
<td>49.15</td>
<td>24.71</td>
<td>50.08</td>
<td>17.77</td>
<td>67.5</td>
<td>15.33</td>
</tr>
<tr>
<td>SGT</td>
<td>25.36</td>
<td>49.59</td>
<td>22.76</td>
<td>18.64</td>
<td>8.92</td>
<td>-</td>
<td>10.83</td>
</tr>
<tr>
<td>Cluster-Kernel</td>
<td>19.41</td>
<td>48.31</td>
<td>48.28</td>
<td>42.05</td>
<td>18.73</td>
<td>67.32</td>
<td>13.33</td>
</tr>
<tr>
<td>Data_Dep,Reg</td>
<td>17.96</td>
<td>50.21</td>
<td>41.25</td>
<td>45.89</td>
<td>12.49</td>
<td>63.65</td>
<td>13.33</td>
</tr>
<tr>
<td>LDS</td>
<td>17.57</td>
<td>49.27</td>
<td>28.85</td>
<td>50.63</td>
<td>15.63</td>
<td>61.9</td>
<td>12.17</td>
</tr>
<tr>
<td>Laplacian RLS</td>
<td>18.99</td>
<td>48.97</td>
<td>43.95</td>
<td>45.68</td>
<td>5.44</td>
<td>54.54</td>
<td>9.5</td>
</tr>
<tr>
<td>CHM</td>
<td>20.53</td>
<td>46.9</td>
<td>39.03</td>
<td>43.01</td>
<td>14.86</td>
<td>-</td>
<td>11.66</td>
</tr>
<tr>
<td>Self</td>
<td>20.13</td>
<td>50.81</td>
<td>40.67</td>
<td>49.84</td>
<td>11.38</td>
<td>46.51</td>
<td>13.83</td>
</tr>
<tr>
<td>LP</td>
<td>10.55</td>
<td>48.38</td>
<td>43.61</td>
<td>43.63</td>
<td>9.61</td>
<td>35.13</td>
<td>5.83</td>
</tr>
<tr>
<td>PVM</td>
<td>15.36</td>
<td>49.1</td>
<td>39.25</td>
<td>37.72</td>
<td>17.08</td>
<td>39.77</td>
<td>8.16</td>
</tr>
<tr>
<td>AGR</td>
<td>27.02</td>
<td>49.89</td>
<td>42.09</td>
<td>45.54</td>
<td>11.45</td>
<td>39.43</td>
<td>13</td>
</tr>
<tr>
<td>MMLLP</td>
<td>14.44</td>
<td>49.57</td>
<td>44.35</td>
<td>44.7</td>
<td>11.16</td>
<td>35.93</td>
<td>8.83</td>
</tr>
<tr>
<td>LPDGL(Linear)</td>
<td>19.77</td>
<td>43.5</td>
<td>44.15</td>
<td>45.11</td>
<td>38.11</td>
<td>73.21</td>
<td>14.17</td>
</tr>
<tr>
<td>LPDGL</td>
<td>17.88</td>
<td>48.17</td>
<td>42.73</td>
<td>42.01</td>
<td>5.37</td>
<td>61.69</td>
<td>6.83</td>
</tr>
<tr>
<td>SSOKMF</td>
<td>21.07</td>
<td>42.97</td>
<td>15.66</td>
<td>19.04</td>
<td>11.20</td>
<td>38.82</td>
<td>6.16</td>
</tr>
<tr>
<td>SSOKMF-RBF</td>
<td>20.93</td>
<td>42.58</td>
<td>12.79</td>
<td>28.82</td>
<td>10.97</td>
<td>37.48</td>
<td>5.33</td>
</tr>
</tbody>
</table>
Table 5-2: Error rate for 6 benchmark datasets in the transductive setup for different methods in the state of the art for \( l = 100 \) labeled samples. The three best results for each dataset are marked in red/bold (1), blue/italic (2) and green/underline (3), respectively. (Avg. Rank: Average of all the rankings obtained in each dataset)

<table>
<thead>
<tr>
<th>Datasets</th>
<th>USPS</th>
<th>BCI</th>
<th>g241c</th>
<th>g241d</th>
<th>Digit1</th>
<th>COIL</th>
<th>Avg. Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labeled Samples: 100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1NN</td>
<td>7.64</td>
<td>44.83</td>
<td>40.28</td>
<td>37.49</td>
<td>6.12</td>
<td>23.27</td>
<td>12.5</td>
</tr>
<tr>
<td>SVM</td>
<td>9.75</td>
<td>34.31</td>
<td>23.11</td>
<td>24.64</td>
<td>5.53</td>
<td>22.93</td>
<td>9.83</td>
</tr>
<tr>
<td>MVU+1NN</td>
<td>6.09</td>
<td>47.42</td>
<td>44.05</td>
<td>43.21</td>
<td>3.99</td>
<td>32.27</td>
<td>13.16</td>
</tr>
<tr>
<td>LEM+1NN</td>
<td>6.09</td>
<td>44.64</td>
<td>42.14</td>
<td>39.43</td>
<td>2.52</td>
<td>36.49</td>
<td>10.83</td>
</tr>
<tr>
<td>QC+CMN</td>
<td>6.36</td>
<td>46.22</td>
<td>22.05</td>
<td>28.2</td>
<td>3.15</td>
<td>10.03</td>
<td>8.67</td>
</tr>
<tr>
<td>Discrete Reg</td>
<td><strong>4.68</strong> (1)</td>
<td>47.67</td>
<td>43.65</td>
<td>41.65</td>
<td>2.77</td>
<td><strong>9.61</strong> (2)</td>
<td>9.5</td>
</tr>
<tr>
<td>TSVN</td>
<td>9.77</td>
<td>33.25</td>
<td>18.46</td>
<td>22.42</td>
<td>6.15</td>
<td>25.8</td>
<td>9.33</td>
</tr>
<tr>
<td>SGT</td>
<td>6.8</td>
<td>45.03</td>
<td>17.41</td>
<td>9.11</td>
<td><strong>2.61</strong> (3)</td>
<td>-</td>
<td>8.33</td>
</tr>
<tr>
<td>Cluster-Kernel</td>
<td>9.68</td>
<td>35.17</td>
<td><strong>13.49</strong> (3)</td>
<td><strong>4.95</strong> (1)</td>
<td>3.79</td>
<td>21.99</td>
<td>7</td>
</tr>
<tr>
<td>Data_Dep,Reg</td>
<td>5.1</td>
<td>47.47</td>
<td>20.31</td>
<td>32.82</td>
<td><strong>2.44</strong> (2)</td>
<td>11.46</td>
<td>7.67</td>
</tr>
<tr>
<td>LDS</td>
<td><strong>4.96</strong> (3)</td>
<td>43.97</td>
<td>18.04</td>
<td>23.74</td>
<td>3.46</td>
<td>13.72</td>
<td>6.33</td>
</tr>
<tr>
<td>Laplacian RLS</td>
<td><strong>4.68</strong> (1)</td>
<td><strong>31.36</strong> (3)</td>
<td>24.36</td>
<td>26.46</td>
<td>2.92</td>
<td>11.92</td>
<td>6.16</td>
</tr>
<tr>
<td>CHM</td>
<td>7.65</td>
<td>36.03</td>
<td>24.82</td>
<td>25.67</td>
<td>3.79</td>
<td>-</td>
<td>11.16</td>
</tr>
<tr>
<td>LPDGL(Linear)</td>
<td><strong>13.44</strong></td>
<td><strong>24.9</strong> (1)</td>
<td>34.04</td>
<td>33.62</td>
<td>10.19</td>
<td>70.64</td>
<td>12.5</td>
</tr>
<tr>
<td>LPDGL</td>
<td>5.09</td>
<td>34.52</td>
<td>21.54</td>
<td>23.9</td>
<td><strong>2.23</strong> (1)</td>
<td><strong>7.27</strong> (3)</td>
<td>4.67</td>
</tr>
<tr>
<td>SSOKMF</td>
<td>19.85</td>
<td>31.94</td>
<td><strong>12.85</strong> (1)</td>
<td><strong>8.80</strong> (2)</td>
<td>7.80</td>
<td>17.27</td>
<td>8</td>
</tr>
<tr>
<td>SSOKMF-RBF</td>
<td>12.62</td>
<td><strong>30.58</strong> (2)</td>
<td><strong>12.88</strong> (2)</td>
<td>22.91</td>
<td>5.97</td>
<td>15.53</td>
<td>7.33</td>
</tr>
</tbody>
</table>

Table 5-3: Main properties of transductive datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Classes</th>
<th>Dimension</th>
<th>Instances</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>g241c</td>
<td>2</td>
<td>241</td>
<td>1500</td>
<td>Artificial</td>
</tr>
<tr>
<td>g241n</td>
<td>2</td>
<td>241</td>
<td>1500</td>
<td>Artificial</td>
</tr>
<tr>
<td>Digit1</td>
<td>2</td>
<td>241</td>
<td>1500</td>
<td>Artificial</td>
</tr>
<tr>
<td>USPS</td>
<td>2</td>
<td>241</td>
<td>1500</td>
<td>Imbalanced</td>
</tr>
<tr>
<td>Coil2</td>
<td>2</td>
<td>241</td>
<td>1500</td>
<td></td>
</tr>
<tr>
<td>BCI</td>
<td>2</td>
<td>117</td>
<td>400</td>
<td>EEG signals</td>
</tr>
</tbody>
</table>
by selecting the budget size of highest performance in each one of the datasets, and for the construction of the budget itself, two methods were explored:

1. Random selection: On each one of the datasets, we apply a random selection of \( b \) (the budget size) instances to compose the budget matrix.

2. K-means: Instead of selecting instances, we constructed prototypes by applying the K-means method on each dataset and defining \( b \) as the number of centroids. The final founded centroids are used to construct the budget matrix.

After evaluated both alternatives, a significant difference was not found, however, using K-means requires additional computational requirements. Thus the use of K-means was discarded and in the consequent evaluations, random selection will be the only alternative.

In order to observe the impact of the budget size, we evaluated the performance of the method in terms of classification accuracy and training time when increasing the number of samples to construct the budget. For this evaluation, we use the MNIST dataset which is considerably larger compared to the datasets used in the previous evaluations. The evaluation was performed by using a budget size that increases from 1000 to 10000 samples for a classification task. At each result, we reported the average (accuracy/time) of running two experiments. In all the results, we use 60000 samples to train the method and 10000 to evaluate its performance (accuracy/time).

Figure 5-2 shows the results of the SS-OKMF method in terms of classification accuracy in MNIST. It is quite evident that the accuracy improves when more samples are included in the construction of the budget, but also, the training time and the required memory increase considerably when the size of the budget becomes higher. However, we can see that the contribution in the performance improvement is becoming less significant as the size of the budget increases, and the maximum accuracy is achieved with a budget size composed of about 11000 samples, after this amount of samples the accuracy remains relatively constant. Therefore, only the 18% of the training samples are required to construct a suitable budget, which implies a significant reduction in the amount of required memory.

The effect of unlabeled instances

With the aim of evaluating the quality of the semantic space and the prediction of the labels generated by the proposed model a comparison, with several linear and non-linear supervised, unsupervised and semi-supervised dimensionality reduction method has been performed. Among the supervised methods we evaluated the classical support vector machines (SVM) [18], from the group of unsupervised methods we evaluated the principal component analysis (PCA) [9], its kernelized counterpart (KPCA)[111] and the local linear embedding (LLE) [107], and finally, for semi-supervised methods we evaluated the STWOMF [12] and the SS-OKMF. In the case of the kernelized methods (SVM, PCA and SS-OKMF) we evaluated
the behavior with linear and a radial basis functions kernels in order to observe the impact of a non-linear modeling (SVM-RB, KPCA-RB and SS-OKMF-RB).

The evaluation is as follows: first, we observe the performance of each algorithm by using a typical supervised setup, in this case, we only use 10 annotated instances to train each algorithm, and later them, are evaluated in a test partition composed of 500 unlabeled images by predicting the annotations using a 1-KNN in the semantic space (i.e., we annotated the testing images with the class of the closest image from the annotated images (from training) in the learned semantic space by using Euclidean distance). Second, we observe the impact of a transductive setup by using the unlabeled test partition, (i.e. we use the feature representation of the test images to model the semantic representation without taking into account the class information). Finally, we observe the impact of new unlabeled images in the training dataset, in this case we, combine the classical unsupervised setup with the transductive setup. In the case of unsupervised and semi-supervised methods the semantic space is constructed by using all the training images, but in the case of semi-supervised method the class information of the percentage of the annotated images is also taken into account, and in the case of supervised methods, the annotation model is only constructed with the percentage of annotated images.

Figure 5-3 shows the average accuracy classification for all the previous methods in the G241C dataset, the first value in the x axis (from left to right) is the performance of each algorithm under a supervised setup, in this case, as is expected, the supervised methods
Figure 5-3: Classification accuracy in G241C dataset for different training configurations (10$l$: supervised with 10 labeled samples, 10$l$ − transd: transductive and 10$l$ − transd: transductive with extra $n$ unlabeled samples)

present the best performances; the second value is the performance under a transductive setup, in this case, the semi-supervised and unsupervised methods increased significantly their performance, surpassing the previous performance obtained by supervised methods, the results also show that among the methods based on a semantic representation, the non-linear methods present a better accuracy. Furthermore, by taking into account that all methods use the same classification strategy, this increase in the performance implies that non-linear methods can model a more accurate semantic representation in comparison with linear methods. This more accurate representation is that which can maximize the discrimination between classes by grouping the elements that belong to the same class and separating those of different classes; the following reported values (10$l$ + $n$) present the same transductive setup, with $n$ additional unlabeled images. In Figure 5-3, we can see that for all unsupervised and semi-supervised algorithms, the increment of unlabeled samples helps in the final performance. Since SS-OKMF can predict directly the class of the instances, this result is also reported (SS-OKMF-DP). This configuration presents the best performance under all the semi-supervised setups.
5.5 Conclusions

We have presented a novel semi-supervised kernel matrix factorization method that uses a learning-in-a-budget strategy to tackle the memory issue and computation time. The main advantage of using a budget in our method, SS-OKMF, is that it allows us to save memory, since it is not necessary to store the complete kernel matrix, but a smaller kernel matrix proportional to the size of the budget reducing significantly the algorithm complexity and keeping low computational requirements in large-scale problems while the final performance is not affected. SS-OKMF is able to take advantage of annotated data to model a semantic low-dimensional space that preserves the separability of the original classes, and additionally, has the ability to exploit unlabeled instances for modeling the manifold structure of the data and use it to improve its performance in classification.

The experimental evaluation shows that the proposed method presents competitive results in terms of error rate in the transductive setup as well as classification accuracy in the inductive approach, in comparison with several unsupervised, semi-supervised and supervised dimensionality reduction methods. In a considerable percentage of the evaluated datasets our method shows a significant improvement over several existing semi-supervised-learning methods.

One of the drawbacks of the SS-OKMF is the big number of hyper-parameters for fine-tuning that difficult the training step. So, an important future task is to perform an analysis of the sensibility and general behavior of the algorithm respect to each hyper-parameter, in order to try to define the most suitable configuration rank according to the dataset properties. This would reduce the complexity of the hyper-parameter exploration step. Additionally, we plan to perform an evaluation of other kinds of kernels that may further improve the performance of the method by taking into account the particularities of each dataset.
6 Semi-supervised online kernel semantic embedding for multi-label annotation

Chapter 4 presented a kernel-based semantic embedding method that was shown to be effective for modeling a semantic space that can be learned in a semi-supervised fashion, by using annotated instances to maximize the discrimination between classes, but also, the unlabeled instances to estimate the intrinsic manifold structure of the data. This chapter presents a new method that solves two drawbacks of the previous method: first, the method only learns to map from the semantic space to the original space, so after the learning process, the calculation of an inverse matrix is required to project the data; second, in training phase, at each iteration the encoding matrix have to be calculated by solving an equation system, requiring a mixed updating strategy that does not allows the pure definition of an online-learning strategy based on stochastic gradient descent and it could lead to a possible numerical instability. Therefore, in this chapter, inspired in the linear matrix factorization methods proposed in chapter 4, we propose a new method that allows learning the semantic projection and back-projection from and to the semantic space at the same time, this also allows a formulation in an entirely end-to-end architecture that can be optimized with stochastic gradient descent.

This work has been published in Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, 22th Iberoamerican Congress, CIARP 2017. This publication has won the IARP-CIARP best paper award 1. Additionally, an extended version was submitted to the Journal of Pattern Recognition Letters.

6.1 Introduction

The automatic multi-label annotation problem presents several applications in areas as diverse as text and music categorization and classification, semantic labeling of images and videos, medical diagnosis, and functional genomics, among others [131]. Several methods transform the problem of multi-label learning to a conventional classification problem (i.e, a set of binary classification problems solved independently). Unfortunately this kind of approaches present two principal drawbacks, first, usually they do not scale well when the

1https://www.ciarp2017.org/
number of labels and/or instances increase, and second, these approaches do not have into account the possible strong correlations between the labels. Another important issue is that these approaches require a significant amount of labeled data to achieve a reasonable generalization performance. In multi-label learning, this issue is more evident than in single-class classification since manually assigning multiple labels is more time-demanding than assigning unique global labels. A possible strategy to deal with a large number of labels and the lack of annotated instances is to find a compact representation of them by using, for instance, a dimensionality reduction method. This approach is followed by multi-label latent space embedding methods, which have recently shown competitive results. There are several strategies to construct the latent semantic space, and most of them proposed supervised and semi-supervised extensions. The important thing about this kind of methods, is that this compact semantic representation also can be modeled by the discriminative structure of not only labeled but also unlabeled data. In this section, we present a method for multi-label annotation based on semantic embedding that finds a common semantic space based on the kernel feature representation of an instance and its corresponding labels that model a mapping between the feature representation and the annotation labels. The proposed method has three important characteristics: 1) the method is formulated as a semi-supervised learning algorithm that learns to construct a common semantic representation not only from labeled instances but also from unlabeled ones, 2) despite being based on kernels, the method scales well to deal with large datasets thanks to a budget restriction which allows tackling one of the main problems of kernel-based methods, that is the scalability in terms of number of training instances, and 3) the method is formulated as an on-line learning algorithm, based on stochastic gradient descent, which allows it to deal with large collections of data, achieving a significant reduction in memory requirements and computational load. Additionally, the latter characteristic allows the efficient implementation of the method in dataflow GPU frameworks such as Theano and TensorFlow, which are used for efficient training and simulation of deep neural networks.

6.2 Multi-label annotation based on semantic embedding methods

The existing methods for multi-label classification problems can be grouped into two main categories [131]: 1) problem transformation methods, which transform the multi-label classification problem into several single-label classification or regression problems and 2) algorithm adaptation methods, which extend specific learning algorithm in order to handle multi-label data directly. In the group of algorithm adaptation methods, we can find several adaptations to classical discriminative methods. For instance, Andrews et al. [5] proposed two extensions for Support Vector Machine (SVM) for multi-instance learning methods the miSVM and MISVM. miSVM treats instance labels as unobserved variables and maximizes
the margin on instances. MISVM, in contrast, define a new concept of bag margin maximization that maximizes the difference between individual patterns. Unfortunately, these methods present two main problems: first, Kernel-based methods usually do not scale well due to the high computational complexity caused by the kernel matrix that grows quadratically with the number of training instances, and second, these methods cannot be extended for a semi-supervised learning and require ground-truth labels for all training documents. Topic models can overcome the second problem by modeling a compact semantic representa-
tion that can be modeled only by the discriminative structure of the data. Classical semantic embedding models are usually unsupervised, but several extensions to add supervision have been proposed, for instance, several matrix factorization based methods have been extended to improve the semantic representation by taking advantage of label information [11, 76]. Under the same approach, several probabilistic topic models have been extended. For instance, many works have extended the classical Latent Dirichlet Allocation (LDA) [15] to add supervised and semi-supervised information, such as Semi-supervised LDA [90], Maximum Entropy Discrimination Latent Dirichlet Allocation (MedLDA) [160], Partially Labeled LDA (PLLDA) and more recently the Semi-supervised Multi-label Topic Model (MLTM) [121]. Probabilistic topic models like LDA have the advantage of incorporating prior knowledge to guide the topic modeling process to improve both the quality of the resulting topics and of the topic labeling, but unfortunately are very computational demanding, making them prohibited for large scale problems.

In this section we propose the Semi-supervised Online Kernel Semantic Embedding (SS-OKSE) method that is formulated for large scale problems by tackling two main issues: first, the method presents a budget restriction that reduces the computational complexity caused by the kernel matrix, and second, the proposed method is formulated as an on-line learning algorithm which allows it to deal with large collections of data.

### 6.3 Semi-supervised online kernel matrix factorization for multi-label annotation

Based on the matrix formulation described in Equation 5-3, we can find an encoding matrix \( H \) that represents the original data points in an \( r \) dimensional semantic space. In this work, we propose not to construct an explicit representation in the semantic space but learn a mapping from the feature representation to this semantic space, and again, using only the \( b \) representative points to model the restricted kernel feature space:

\[
H = W_x \phi(B)^T \phi(X) \tag{6-1}
\]

\[
H = W_x K(B, X) \tag{6-2}
\]

In this manner, in a similar way that the liner two-way matrix factorization formulation (please refer to Section 4.1.2), the model learns two transformations what allows to project the original data representation to the lower-dimensional semantic space and at the same time to reconstruct from this semantic representation the original data in the feature space.

\[
\phi(X) \approx \phi(B) \tilde{W}_x W_x K(B, X) \tag{6-3}
\]
Additionally to the original feature representation, we want the semantic representation to also lie the label representation $Y \in \mathbb{R}^{m \times k}$, where $m$ is the total number of possible labels and $k$ a reduced number of annotated instances (i.e., $k \ll l$), as follows:

$$Y \approx \sigma \left( \tilde{W}_y H \right) = \sigma \left( \tilde{W}_y W_x K (B, X) \right)$$

(6-4)

where $W_y \in \mathbb{R}^{m \times r}$ is another transformation matrix that projects from the semantic representation to the label space, and finally, an additional non-linear function $\sigma$ is used to add more flexibility to the model. Putting all these restriction together, the final model can be represented as is shown in Figure 6-1a.

### 6.3.1 Loss function

The final loss function to be minimized forces the feature reconstruction by defining a squared minimum error and learns the binary label reconstruction by imposing a binary cross entropy function:

$$\min_{W_x, \tilde{W}_x, W_y} J^i \left( W_x, \tilde{W}_x, \tilde{W}_y \right) = \frac{\alpha}{2} \left\| \Phi \left( x^i \right) - \Phi \left( B \right) \tilde{W}_x W_x K \left( B, x^i \right) \right\|^2_F +$$

$$+ \frac{\beta}{k} \sum_{i=0}^{k} \log \left( 1 + \exp \left( -y^i \cdot \tilde{W}_y W_x K \left( B, x^i \right) \right) \right) +$$

$$+ \frac{\lambda_1}{2} \left\| W_x \right\|^2_F + \frac{\lambda_2}{2} \left\| \tilde{W}_x \right\|^2_F + \frac{\lambda_3}{2} \left\| \tilde{W}_y \right\|^2_F$$

(6-5)

where $\alpha$ and $\beta$ control the relative importance of reconstructing the feature and label representation, respectively, and $\lambda_1, 2, 3$ control the relative importance of the regularization terms, which penalize large values for the Frobenius norm of the transformation matrices. Unfortunately, solve this problem directly is infeasible due to the function $\phi(\cdot)$ performs a mapping to a highly-dimensional space or even to an infinite-dimensional space, but, the first term of the loss function can be rewritten in terms of kernel matrices and employ the kernel trick [58].

$$\frac{\alpha}{2} \left\| \Phi \left( x^i \right) - \Phi \left( B \right) \tilde{W}_x W_x K \left( B, x^i \right) \right\|^2_F = \frac{\alpha}{2} \left( K \left( x^i, x^i \right) - 2 K \left( x^i, B \right) \tilde{W}_x W_x K \left( B, x^i \right) + K \left( B, x^i \right)^T W_x^T \tilde{W}_x^T K \left( B, B \right) \tilde{W}_x W_x K \left( B, x^i \right) \right)$$

(6-6)

Finally, we can make a change of variables and redefine the first term of the loss function $\left( \frac{\alpha}{2} (1 - 2 (z^i)^T \tilde{z}^i + (z^i)^T K \left( B, B \right) \tilde{z}^i) \right)$ and the structure (Figure 6-1b), so that can be easily implemented in some deep learning framework.
6.3.2 Kernel learning

In the implemented model architecture (Figure 6-1b), we can see that the kernel matrix \( K(B, X) \) can be precomputed and be used directly as input and as a target \((z)\) to train the algorithm, this avoids the necessity of recalculating the kernel matrix at each epoch. This implies a considerable reduction in the computational requirements. But, on the other hand, preserving the calculation of the kernel function in the computational graph presents an interesting advantage. This is, all the parameters in the graph can be learned. Therefore, the hyper-parameter of certain kernel functions can be automatically determined in the training stage. This avoids the necessity of a costly hyper-parameter exploration. Even more, the adequate selection of this parameter value drastically affects the final performance. For this reason, obtaining this optimum value is essential. In order to learn this parameter, some modifications have to be made in the model implementation as shown in Figure 6-2. The main difference with the previous model is that the target value \((z)\) is not fixed and have to be updated at each iteration. Therefore, both \(\tilde{z}\) and \(z\) have to be recalculated before applying them to the loss function.

This model is clearly computationally more demanding due to the need to constantly compute the kernel function. Even so, this avoids the necessity of a costly exploration for fine tuning of this hyper-parameter.

---

**Figure 6-2:** Model implementation of SS-OKSE with kernel hyper-parameter learning.
6.3.3 Prediction

Once the parameters have been learned (coding and decoding matrices), we can use the model to predict the label representation \( \hat{y} \) from the feature representation \( x \) of a new unlabeled document by forward propagating it through the network.

6.3.4 Implementation details

The proposed method was implemented in the Keras [30] Framework, a high-level neural networks API written in Python and capable of running on top of either TensorFlow or Theano [127] libraries. The optimization is performed by stochastic gradient descent (SGD) with the RMSProp optimizer.

6.4 Experiments and Results

In this section, the proposed SS-OKSE algorithm will be evaluated in a multi-label annotation task under a semi-supervised setup. In order to compare our algorithm, we use the same experimental setup proposed by Soleimani et al. [121], where the performance of our method is compared against several supervised (PLLDA, miSVM and MISVM) and semi-supervised (ssLDA, and MLTM) methods in two different datasets of text documents:

**Delicious** This dataset is composed of tagged web pages from the social bookmarking site delicious.com [163]. We adopt the subset proposed in [121] where the top 20 common tags are used as a class labels, constructing a subset portioned in 8350 documents for training and 4000 documents for testing. The documents are represented in a bag-of-word representation composed by a codebook of 8500 unique words obtained after applying Porter stemming and stopword removal.

**Ohsumed** This collection contains medical abstracts from the MeSH categories of the year 1991 [163]. The specific task in this dataset is to categorizing 23 cardiovascular diseases categories. It is composed by 11122 training and 5388 test documents. Almost half of the documents have more than one label.

6.4.1 Experimental setup

For a fair comparison for all methods, where in some of them a suitable selection of the threshold is not trivial, the ROC AUC (Area Under the Curve) is used as the evaluation metric. Micro-ROC and Macro-ROC AUC are reported separately. In Micro-ROC, TPR and FPR are computed globally. In Macro-ROC, the ROC AUC is computed for each class across all documents and then the average is taken over all classes. While Micro-ROC may be dominated by the bigger classes, Macro-ROC gives equal weight to all classes and better
reveals performance on rare classes. For each dataset, a \((1 - p)\) fraction is randomly selected from the training documents and their labels are removed. Then, for semi-supervised models, both labeled and unlabeled documents are used for training. But, for the purely supervised methods, only the remaining labeled documents are used for training. The annotation experiment is performed for different label proportions \(p \in 0.01, 0.05, 0.1, 0.3, 0.6, 0.8, 0.9\) (five experiments are executed and the average is reported).

6.4.2 Determining the hyperparameters

Our model has 7 hyperparameters \((\alpha, \beta, \lambda_1, \lambda_2, \lambda_3, b, r)\). To properly determine the values of these hyperparameters, we randomly extract 20\% of instances from the training set to validate the performance under a random (uniform) exploration in 30 different hyperparameter configurations trained with the remained 80\% of training instances. The best configuration was chosen to be evaluated with the test partition. (this strategy have shown similar results than grid search while requires much fewer computation resources [13]).

6.4.3 Multi-label annotation performance

Figure 6-3a presents the performance in ROC AUC in the Delicious dataset for different proportions of labeled documents. As we can see, the proposed SS-OKSE using a linear kernel presents competitive results against the other semi-supervised algorithms, and using a histogram intersection (hi) kernel presents the best performance showing that a suitable kernel can help to model a good semantic space where non-labeled instances can be exploited. This becomes evident by achieving a performance very close to the maximum possible just with only 10\% percent of labeled instances. Another important result is that this performance is obtained only with a budget composed by 500 instances randomly selected, this is only about the 6\% of the training instances. Figure 6-3b presents the same previous experiment for the Delicious dataset. In Micro-ROC our method presents competitive results but the results in Macro-ROC suggest that the classes with the highest number of instances are dominating the learning. Still, it is important to emphasize that this result is obtained by using a budget of 2500 instances randomly selected (22\% of the training instances).

6.4.4 Multi-label Annotation Performance using kernel learning

In this subsection, we evaluate the kernel learning variation of the proposed model. Due to the nature of the data features of both dataset, the exponential \(\chi^2\) (Chi-squared) kernel is proposed, which is commonly used on histograms (bags of words) representations. This kernel function is defined as follows:
6.4 Experiments and Results

Figure 6-3: Performance comparison in multi-label annotation. (hi: Histogram intersection kernel)

\[ k(x, y) = \exp \left( -\gamma \sum_i \frac{(x^i - y^i)^2}{x^i + y^i} \right) \]  

(6-7)

So, in this case, the free parameter to learn is the \( \gamma \) value, which is a scaling parameter. Due to the model variation that learns the kernel hyper-parameter proposed in subsection 6.3.2 is computationally more demanding, it is proposed to use it only in the validation partition to define the hyper-parameter value for each dataset. This is done by training the model in a supervised fashion. Once the parameter is found, this value is fixed and used in the following semi-supervised experiments. Figure 6-4 shows the learning progression of the \( \gamma \) parameter at each epoch, along with the epochs we can see that the gamma value achieve convergence. In the Delicious dataset, the obtained value is \( \gamma = 0.52 \), and in the same manner, for Ohsumed dataset value obtained is \( \gamma = 1.1 \).

Once we have found the optimal \( \gamma \) values, we perform the same semi-supervised experiments of the subsection 6.4.3, for the \( \chi^2 \) kernel. These results are reported in Figure 6-5. As we
62 6 Semi-supervised online kernel semantic embedding for multi-label annotation

**Figure 6-4**: Gamma ($\gamma$) learning progression for $\chi^2$ (Chi-squared) kernel in Delicious dataset

**Table 6-1**: Dataset information a data partition for each dataset. N: number of unique words, m: number of classes, l: number of instances, cardinality: average number of labels per instance.

<table>
<thead>
<tr>
<th>Name</th>
<th>n</th>
<th>m</th>
<th>l</th>
<th>Cardinality</th>
<th>l</th>
<th>Cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delicious</td>
<td>8520</td>
<td>20</td>
<td>8251</td>
<td>2.89</td>
<td>3983</td>
<td>2.91</td>
</tr>
<tr>
<td>Ohsumed</td>
<td>13117</td>
<td>23</td>
<td>11122</td>
<td>1.65</td>
<td>5388</td>
<td>1.64</td>
</tr>
</tbody>
</table>

can see in Figure 6-5a, in Delicious dataset, the proposed method using the $\chi^2$ kernel with the learned $\gamma$ achieve a considerable increase in both Micro and Macro ROC measures, in comparison with the version with the histogram intersection kernel, This shows that was found a suitable value for $\gamma$. In Ohsumed dataset an increase in the performance is also obtained, but this increase is presented only when the percentage of annotated instances is high (between 50% and 90% of labeled instances), This may suggest that the $\gamma$ learned in fully supervised setup is not the most suitable when there are a high number or unlabeled instances.

### 6.4.5 Time-consuming and complexity

The main advantage of the SS-OKSE method is its highly scalable formulation. Table 6-2 shows the comparison between the proposed method and the state-of-the-art topic model method (we use the implementation supplied by the authors [121]). In both datasets SS-OKSE obtains remarkable speedups in comparison with MLTM under the same conditions (same computer with CPU device on a single core), but also thanks to its implementation, SS-OKSE is capable of running transparently in GPU devices achieving even more dramatic speedups, showing the ability of the proposed method to work on large scale datasets.
We have presented the SS-OKSE, a novel semi-supervised kernel semantic embedding method that uses a budget restriction to tackle the memory issue and computation time associated to kernel methods. The main advantage of using a budget in our method is that it allows us to save memory, since it is not necessary to store the complete kernel matrix, but a significantly smaller matrix defined by a budget keeping low computational requirements in large-scale problems. SS-OKSE is able to take advantage of annotated data to model a semantic low-dimensional space that preserves the separability of the original classes, and additionally, has the ability to exploit unlabeled instances for modeling the manifold structure of the data and use it to improve its performance in multi-label annotation tasks. The results confirm that the ability of the proposed method for modeling non-linearities can over-improve the performance in the multi-label annotation task.
Table 6-2: Training time comparison (in minutes). For CPU device, only one core is using. Experiment running in a Intel(R) Xeon(R) CPU E5-2640 v3 @ 2.60GHz with 16 cores, 64 GB RAM and an Nvidia Titan X.

<table>
<thead>
<tr>
<th>Delicious</th>
<th></th>
<th></th>
<th>Ohsumed</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>CPU</td>
<td>Speedup</td>
<td>GPU</td>
<td>Speedup</td>
<td>CPU</td>
</tr>
<tr>
<td>641.3</td>
<td>43.5</td>
<td>22.1</td>
<td>5.1</td>
<td>186.9</td>
<td>1026</td>
</tr>
</tbody>
</table>
7 Online kernel semantic embedding for cross-modal retrieval

All the methods proposed so far model a semantic representation supported by a second modality that presents a structured and clean representation, such as defined classes or a restricted set of labels. In this section, we evaluated the richness of a common semantic representation that is modeled by two unstructured data modalities. For this purpose, The last matrix factorization strategy is extended to modeling simultaneously a factorization problem for each modality in a symmetric way while a semantic alignment strategy enforces that both factorizations share the same semantic representation. This strategy is evaluated under the challenging task of cross-modal retrieval.

This work will be published in journal to be defined.

7.1 Introduction

Cross-modal retrieval is an information retrieval task where users express their information need throughout a query in one modality (or media type) and get the related result set in another modality. The most common instance of the cross-modal retrieval task corresponds to the interaction of two modalities, usually, text and images. For instance, a query can be expressed as a text fragment and the expected results would be related images, this is especially useful to answer common queries made by users searching for products in an online shop. Also, cross-modal retrieval is commonly used as a proxy task to image understanding by allowing to assign short text descriptions similar to that a human would do, this task can be simplified and better understood as the ability of the system to pairing an image with a descriptive text.

The main challenge of cross-media retrieval is the media gap issue [96], which means that because of the different nature of each modality the representations are inconsistent and lie in different feature spaces, so in the retrieval process the similarity between queries and documents cannot be directly measured, such as in other information retrieval tasks is commonly done. A typical strategy to address this problem is projecting queries and documents from the different modalities to a common space, where similarity measures can be performed.

In this section, we present an On-line Multimodal Kernel Semantic Embedding method for Cross-modal retrieval (MKSE-CM). The main idea of this model is to construct a common
representation by solving a kernel matrix factorization problem for each modality. Each factorization problem defines a semantic space that is forced to be aligned so that objects in one modality semantically related to objects in the other have a similar semantic representation. The proposed method has five important characteristics:

1. The construction of the common semantic representation is based on kernel methods allowing to model complex non-linear relations between the different modalities.

2. Despite being based on kernels, the method scales well to deal with large datasets thanks to a budget restriction which allows tackling one of the main problems of kernel-based methods, that is the scalability in terms of number of training instances.

3. This method can take advantage of supervised information such as semantic label data that allows the construction of a richer common semantic representation.

4. The method is formulated as an on-line learning algorithm, based on stochastic gradient descent, which allows it to deal with large collections of data, achieving a significant reduction in memory requirements and computational load.

5. Additionally, the latter characteristic allows a straightforward efficient implementation of the method in dataflow GPU frameworks such as Theano and TensorFlow, which are used for efficient training and simulation of deep neural networks.

The method was evaluated in a publicly available cross-modal retrieval dataset extracted from Wikipedia, which is widely used to evaluate cross-modal retrieval system performance. The dataset has standard partitions for training and test that facilitate the comparison and replication of results [104, 32].

### 7.2 Related work

Most of the works in cross-modal retrieval try to address the problem by modeling a common space where the different modalities can be projected. There are several strategies that model this common space by using statistical correlation analysis methods. Canonical Correlation Analysis (CCA)[29] and Partial Least Squares (PLS)[54, 52] methods are some of the most common approaches to solve cross-modal retrieval problem based on this approach. The aim of this methods is to learn a linear mapping from the original space to a joint space, where the correlation between modalities is maximized.

Kernelized extension of CCA (KCCA) method was proposed to learn mapping non-linear relationships[125], improving retrieval performance and learning complex relationships such as the importance of the objects in a scene. Despite the effectiveness shown by these methods, they require keeping in memory all the dataset covariance matrix. In many implementations,
these methods employ Singular Value Decomposition (SVD), whose computational cost limits their ability to scale on large datasets. Rasiwasia et al. [104] and Costa et al. [32] proposed three methods: (i) Correlation Matching (CM) that is based on CCA and KCCA family methods to maximize the correlation in the projections of different modalities; (ii) Semantic Matching (SM) that uses a classifier and label information to map the multimodal data to a new space modeled by semantic concepts in the label space; (iii) and an ensemble method called Semantic Correlation Matching (SCM) that combines CM and SM methods mapping from CCA space to a label space. This ensemble method shows a significant improvement and a better semantic space can be modeled. Also, several Matrix Factorization (MF) methods have been extended to learn from multimodal data [47] and applied to image indexing and multimodal retrieval. Online multimodal matrix factorization algorithm was proposed to overcome the scalability limitations and show good performance on the unimodal image retrieval task [22].

In the area of neural networks, autoencoders and cross autoencoders are deep learning methods that show the ability to learn compact representations of the original data, embedding them into a new latent space. Feng et al. [41, 40] proposed an autoencoder and cross-autoencoder models where the correlation between hidden layers are maximized. In the same area of neural networks, some deep variations of canonical correlation analysis have been proposed [112, 3], showing that the use of several layers helps to better model the complex non-linear relationships between the different modalities.

Some recent works proposed supervised alternatives [112, 98] to model a common space that rely on rich semantic annotation, such as tags, keywords or captions that usually appear in the multimedia content on the web (for instance, news archives, Wikipedia pages and blog posts, among others), this kind of methods present the best performance, showing that with the use of semantic information a richer common semantic representation can be constructed. The method presented uses as main strategy the construction of a common semantic representation based on kernel methods allowing to model complex non-linear relations between the different modalities without the necessity of defining deep and complex architectures; but, also avoiding the high computational complexities presented by methods based on KCCA: first, the construction of the joint space is not based on the maximization of the correlation between the modalities, which requires a costly calculation and storage of the covariance matrix, and second, despite being based on kernels, the method scales well to deal with large datasets thanks to a budget restriction which allows tackling one of the main problems of kernel-based methods, that is the calculation and storage of the kernel matrices. Also, the proposed method can take advantage of supervised information such as semantic label data to construct a richer common semantic representation. Finally, the proposed method is formulated as an on-line learning algorithm that can be implemented in dataflow GPU frameworks such as Theano [127] or TensorFlow [1] which are used for efficient training and simulation of deep neural networks, allowing a scalable architecture that can deal with large collections of data.
7.3 Method

The proposed method is composed of four main modules: 1) The \textit{kernel semantic embedding} that models the semantic representation for each modality; 2) The \textit{Semantic alignment} strategy that performs the matching between the different semantic projections from each modality; 3) \textit{Label prediction and alignment} that aligns the common semantic representation according with the supervised information, and provides an alternative semantic concept space based on the annotation labels; and 4) the \textit{cross-modal reconstruction} strategy that learns directly to project to the representation of the opposite modality. The details of this main modules are described in the following subsection.

\textbf{Kernel semantic embedding}

This first module refers to the kernelized latent embedding introduced in section 6.3 without taking into account the labeled information. Figure 7-1 shows the module architecture. In the left (7-1a), it is presented the conceptual model that learns the projection to the semantic space and simultaneously the back projection to the original feature space induced by the kernel, and in the right 7-1b, the actual implementation of OKSE so that can be easily implemented in some deep learning framework.
7.3 Method

7.3.1 Semantic alignment

The previous module independently generates a semantic representation for each modality. Particularly, in this work, we address two kinds of data modalities: visual and textual, that will be represented by \( X_v \in \mathbb{R}^{n_v \times l} \) and \( X_t \in \mathbb{R}^{n_t \times l} \) respectively. For each modality, there is an associated matrix \( H_v, H_t \in \mathbb{R}^{r \times l} \) that correspond to the semantic representation of the original data. In order to force the alignment of these semantic representations, we expect for a particular sample \( x^i \in X \) that the semantic representations of both modalities, \( h^i_v \) and \( h^i_t \), are very close. We can enforce this alignment with the following constraint:

\[
\text{sim}(h^i_v, h^i_t) \approx 1 \quad (7-1)
\]

where, \( h^i_v = W_v K(B_v, x^i_v) \), \( h^i_t = W_t K(B_t, x^i_t) \) and \( \text{sim}(\cdot, \cdot) \) is some similarity function with a maximum value of 1.

7.3.2 Label prediction and alignment

Along with the goal of finding a common semantic representation for both modalities, in this work, we propose to take advantage of extra rich semantic concepts that usually appear in the multimedia content in form of tags or keywords. This extra information can help to model a more discriminative space for elements that belong to different semantic concepts. For this purpose, we propose to align the common semantic representation with the semantic concepts by learning to predict the labels from the semantic space. If we represent the labels associated with a specific instance by an \( m \)-dimensional vector in one-hot representation we can define the following projection functions:

\[
\bar{y}^i_t = f(W_y^t h^i_t) \\
\bar{y}^i_v = f(W_y^v h^i_v) \quad (7-2)
\]

where \( W_y^t \in \mathbb{R}^{m \times r} \) and \( W_y^v \in \mathbb{R}^{m \times r} \) are transformation matrices that, along with \( f \) project from the semantic representation of each modality to the corresponding label representations. The \( f \) function can add an extra non-linearity according with the most suitable output for a determined label representation. Additionally, this label representation can be used as an alternative semantic concept space (\( Y \)) to perform the retrieval process.

7.3.3 Learning with negative samples

The label information can be also used to extend the previously defined semantic alignment. Therefore, we can maximize the similarity between the semantic projections of different modalities from the same sample, and also minimize the similarity for projections from
different modalities that belong to examples that do not share any label. This helps to model a more discriminative space for elements that belong to different semantic concepts. For this purpose, new negative examples are created by randomly generating image-text pairs that belong to different instances that share no label, in the same proportion of the original samples. Along with this, we can generalize the semantic alignment constraint as follows:

$$\text{sim}(h_t^i, h_v^j) \approx y_t^i \cdot y_v^j$$ (7-3)

In this manner, if the example $x^i$ does not share any label with the example $x^j$, the similarity between semantic projections will tend to be zero.

### 7.3.4 Cross-modal reconstruction

Another strategy to enrich the semantic representation is forcing to reconstruct directly from each semantic representation the opposite modality. This is, learning for a target representation $f$, another basis matrix $F_{\Phi_f} \in \mathbb{R}^{n \times r}$ where the column vectors will lie within the space of $\Phi(B_f)$, as follows:

$$\Phi(x_v^i) \approx F_{\Phi_f} h_v^i \approx \Phi(B_v) \tilde{W}_v h_v^i$$

$$\Phi(x_t^i) \approx F_{\Phi_f} h_t^i \approx \Phi(B_t) \tilde{W}_t h_t^i$$ (7-4)

Finally, this reconstruction is carry out by defining the following loss functions:

$$\min_{W_t, \tilde{W}_t} J_{\phi_{vt}}^i(W_t, \tilde{W}_t) = \frac{\alpha}{2} \left\| \Phi(x_v^i) - \Phi(B_v) \tilde{W}_v W_t K(B_t, x_t^i) \right\|^2_F$$ (7-5)

$$\min_{W_v, \tilde{W}_v} J_{\phi_{tv}}^i(W_v, \tilde{W}_v) = \frac{\alpha}{2} \left\| \Phi(x_t^i) - \Phi(B_t) \tilde{W}_t W_v K(B_v, x_v^i) \right\|^2_F$$ (7-6)

This restrictions also help to enforce the alignment between the semantic representation of both modalities.

### 7.3.5 Loss function

For the final model, we pull all the previous considerations together and define for an $i$–th instance the following loss function:

$$\min_W J^i(W) = \alpha_1 J_{\phi_t}^i + \alpha_2 J_{\phi_v}^i + \alpha_3 \sigma(y_t^i, \tilde{y}_t^i) + \alpha_4 \sigma(y_v^i, \tilde{y}_v^i) + \alpha_5 (\text{sim}(h_t^i, h_v^i) - y_t^i \cdot y_v^i)^2 + \alpha_6 J_{\phi_{vt}}^i + \alpha_7 J_{\phi_{tv}}^i + \lambda_{1.7} \|W\|_2^2$$
Figure 7-2: Proposed model: On-line Multimodal Kernel Semantic Embedding for Cross-modal retrieval (MKSE-CM)

where $J_{\phi t}^i$ and $J_{\phi v}^i$ are the objective functions for feature reconstruction for textual and visual modalities respectively (from Equation 6-6), $J_{\phi vt}^i$ and $J_{\phi tv}^i$ are the objective functions for cross-modal reconstruction from textual to visual and visual to textual respectively. $W = (W_v, W_t, \tilde{W}_v, \tilde{W}_t, W_{yt}, W_{yt}, W_{tv})$ are all the parameters to be learned, $\alpha_1,...,7$ controls the relative importance of each reconstruction error and $\lambda_1,...,5$ control the relative importance of the regularization terms that penalizes big values for the weight matrices and avoid overfitting. $\sigma(.)$ defines the error function for the label reconstruction. In this work, we are going to assume that all instances in the dataset bellow to disjoint categories, so we are going to use categorical cross-entropy as error function. Also, for efficiency reasons, the cosine similarity will be used as similarity metric for the semantic space ($\text{sim}$). Finally, the full architecture of the proposed model is shown in Figure 7-2.

7.3.6 On-line multimodal kernel semantic embedding algorithm for cross-modal retrieval (MKSE-CM)

The final algorithm consists of the following steps:

1. Construct the budget subsets $B_v$ and $B_t$ for both modalities by uniformly randomly selecting instances from the training set.

2. Extend the training set by adding synthetic negative examples constructed by visual and text representations from different instances that share no semantic labels.
3. Learn the projections from each modality to the common semantic representations $H$ and $Y$.

Once the training process has finished, we can project each modality by propagating the input data through the model (Figure 7-2) and obtain the required reconstruction:

$$x_t \in \mathbb{R}^{n_t \times l} \xrightarrow{M_{th}} h_t \in \mathbb{R}^{r \times l} \xrightarrow{M_{hty}} \tilde{y}_t \in \mathbb{R}^{m \times l}$$

(7-8)

$$x_v \in \mathbb{R}^{n_v \times l} \xrightarrow{M_{vh}} h_v \in \mathbb{R}^{r \times l} \xrightarrow{M_{hvy}} \tilde{y}_v \in \mathbb{R}^{m \times l}$$

(7-9)

**Algorithm 3:** On-line Multimodal Kernel Semantic Embedding algorithm for Cross-modal retrieval (MKSE-CM)

1. **MKSE-CM** ($S = \langle X_v, X_t, Y \rangle$);

   **Inputs**: $X_v$: visual features representation,
   $X_t$: textual features representation,
   $Y$, label representation

   **Outputs**: $M_{th}: X_v \rightarrow H_t$;
   $M_{vh}: X_t \rightarrow H_v$;
   $M_{hty}: H_t \rightarrow Y_t$;
   $M_{hvy}: H_v \rightarrow Y_v$;

   **Hyperparameters**: $b$, budget size, $r$, semantic space dimension

   /* Budget construction */
   2. $B = \langle B_v, B_t \rangle \subset \langle X_v, X_t \rangle$ : $|B| = b \ll n$

   /* Negative samples construction */
   3. $S^- = \langle X_v^-, X_t^-, Y_v^-, Y_t^- \rangle \subset \{ \langle x_v^i, x_t^i, y_v^i, y_t^i \rangle \in S \mid y_t^i \neq y_v^i \}$

   /* Combining positive and negative samples */
   4. $T = \langle X_v \cup X_v^-, X_t \cup X_t^-, Y \cup Y_v^-, Y \cup Y_t^- \rangle$

   /* learn mapping functions to the semantic spaces $H$ and $Y */
   5. $M_{th}, M_{vh}, M_{hty}, M_{hvy} = MKSE(T, B, r)$;
   6. return $M_{th}, M_{vh}, M_{hty}, M_{hvy}$;

### 7.3.7 Cross-modal retrieval process

With all the mapping functions learned in the trained process, the data from both modalities (textual and visual) can be projected into two different common semantic spaces: the intermediate semantic space ($H$) and the semantic concept space modeled by the label annotations ($Y$). The cross-modal retrieval process can be performed by projecting a query from one modality and the target elements from the other modality and retrieve the closest
Table 7-1: Different distance metrics used in the retrieval process

<table>
<thead>
<tr>
<th>Distance metric</th>
<th>(d(a, b))</th>
</tr>
</thead>
<tbody>
<tr>
<td>(L_1)</td>
<td>(\sum_i</td>
</tr>
<tr>
<td>(L_2)</td>
<td>(\sum_i (a_i - b_i)^2)</td>
</tr>
<tr>
<td>(NC)</td>
<td>(1 - \frac{a^T b}{|a||b|})</td>
</tr>
<tr>
<td>(NC_C)</td>
<td>(1 - \frac{(a-\mu_a)^T (b-\mu_b)}{|a-\mu_a||b-\mu_b|})</td>
</tr>
</tbody>
</table>

elements according to a defined distance metric. In this work we explored four different distance metrics: \(L_1, L_2\), normalized correlation (\(NC\)) and centered normalized correlation (\(NC_C\)) (see Table 7-1)[32].

7.3.8 Implementation details

The proposed method was implemented in the Keras [30] framework, a high-level neural networks API written in Python and capable of running on top of either TensorFlow [1] or Theano [127] libraries. The optimization is performed by stochastic gradient descent (SGD) with the Adam optimizer [68].

7.4 Experimental evaluation

7.4.1 Experimental setup

7.4.2 Dataset

The dataset used to evaluate the performance of our method is a dataset extracted from Wikipedia that consists of 2866 documents of text-image pairs that belong to ten classes. Images are described using SIFT descriptors with a 128-word codebook, and text information is described using the result of Latent Dirichlet Allocation (LDA) algorithm with ten latent topics. Additionally, this dataset is annotated with 10 semantic categories (such as “History” and “Biology”). The dataset was split into two partitions: a test set with 693 documents and a train set with 2173 documents as described in [104]. In addition to the original partition, we perform a random partition for validation using 20% of the train set, this is 1738 documents for train and 435 for validation and hyper-parameter exploration. Additionally, both modality representations were \(L_1\) normalized.

7.4.3 Performance measurement

In the following experiments, two different tasks will be evaluated: Image to text (Img2Txt) that refers to define the query with an image and expect as results a ranked list of related
texts and its counterpart Text to images (Txt2Img) where the query is the text document and the expecting results are images. For both tasks, we are going to employ a metric widely used in the information retrieval literature: mAP (Mean Average Precision). mAP has the advantage of summarizing in a single number the precision at different recall levels within a ranked results, and has good properties that make it a reliable measurement [89].

7.4.4 Hyper-parameter exploration

Hyper-parameter tuning is done by cross-validation over 30 randomly generated configurations. this strategy have shown similar results than grid search while requiring much fewer computation resources [13]). The final results are reported for the best configuration applied to the test partition.

7.4.5 Kernel selection

Evidently an adequate selection of the kernel function is an important factor in the final performance of the proposed method, this depends also on the kind of representation of each modality. For the following experiments several kernel functions were evaluated: linear, cosine, radial based function (RBF), Histogram intersection (HI) and Chi squared ($\chi^2$). Some of these kernels such as RBF and $\chi^2$ have its own parameters that also require a proper exploration.

7.4.6 Baseline methods

To assess the effectiveness of the proposed MKSE-CM, in the comparison are included methods within two main categories: first, methods that learn a common space and second methods based on cross-media similarity learning [96].

In the first category, there are several methods based on Canonical Correlation Analysis CCA [29], that learns a common space where the correlation between the modalities is maximized, such as, CCA+SMN (CCA + Semantic Correlation Matching) [105], that first learns the correlation between different modalities by CCA, then the abstraction is achieved by representing texts and images in a more general space, using the available labeled annotations, ml-CCA (Multi-label CCA) [103] based on CCA includes label information to represent semantic concepts, mv-CCA (Multi-View CCA) [46], this method learns a common semantic space using three views image, text, and tags that can be used in supervised or unsupervised train mode. CCA has been extended to model more complex relationships by using deep learning approaches, such as DCMIT [146] that is an unsupervised deep learning method based on DCCA (Deep Canonical Correlation Analysis). In this category, we also can find methods that construct a common space based on other strategies: BITR (Bilateral Image-Text Retrieval) [135] uses a structural SVM (Support Vector Machine) to learn relationships between images and texts; CFA [77] adopts a criterion to minimize the Frobenius norm
Table 7-2: MAP scores by using different distance metrics in the retrieval process for image to text (Img2Txt) and text to image (Txt2Img) tasks

<table>
<thead>
<tr>
<th>Space</th>
<th>Distance metric</th>
<th>Img2Txt</th>
<th>Txt2Img</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$L_1$</td>
<td>0.137±1.16e-03</td>
<td>0.246±1.18e-03</td>
<td>0.192±5.14e-04</td>
</tr>
<tr>
<td></td>
<td>$L_2$</td>
<td>0.137±2.50e-04</td>
<td>0.237±7.76e-04</td>
<td>0.187±3.91e-04</td>
</tr>
<tr>
<td></td>
<td>$NC$</td>
<td>0.297±1.04e-03</td>
<td>0.249±6.47e-04</td>
<td>0.278±7.27e-04</td>
</tr>
<tr>
<td></td>
<td>$NC_C$</td>
<td>0.297±2.6e-03</td>
<td>0.249±8.9e-04</td>
<td>0.273±1.2e-03</td>
</tr>
</tbody>
</table>

between pairwise data in the transformed domain; LGCFL (Local Group based Consistent Feature Learning) [64] learns a common space based on semantic category labels. Finally, the JRL (Joint Representation Learning) [150] explores jointly the correlation and semantic information in a unified optimization framework by using labeled and unlabeled data.

In the second category, we can group methods such as CMCP (Cross-Modality Correlation Propagation) [147] that is a semi-supervised graph-based cross-media similarity measurement learning method that learn similarities and dissimilarities between training samples; the HSNN (heterogeneous similarity measure with nearest neighbors) [148] that is a supervised cross-media similarity measurement learning method that finds a heterogeneous similarity by computing the probability for two media objects belonging to the same semantic category. This probability is calculating by analyzing the homogeneous nearest neighbors of each media object; and finally, the JGRHML (Joint graph regularized heterogeneous metric learning) [149] that integrates the structure of different modalities into a joint graph regularization to learn a high-level heterogeneous semantic metric through label propagation.

7.4.7 Results and discussion

7.4.8 Distance metric evaluation

As discussed in the Subsection 7.3.7, four different distance metrics: $L_1$, $L_2$, $NC$, $NC_C$, where used for ranking the results in both the $H$ and $Y$ spaces. Table 7-2 presents the mAP scores obtained by using each distance measures in $H$ and $Y$ spaces (The reported values are the result of running each configuration five times). The results show that $NC_C$ exhibits in average the best performance in $Y$ with a significant difference in the task of image to text (Img2Txt) (see Figure 7-2). Since $NC_C$ had the best average performance in nearly all experiments, this measure was adopted as distance metric for all the following experiments.
Table 7-3: MAP scores for cross-retrieval over the semantic H space for image to text (Img2Txt) and text to image (Txt2Img) tasks by using different combinations of the objectives (SA: Semantic alignment; DR: Direct Reconstruction; CR: Cross-reconstruction; LR: Label Reconstruction).

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Img2Txt</th>
<th>Txt2Img</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA</td>
<td>0.232±1.5e-03</td>
<td>0.197±8.9e-04</td>
<td>0.214±8.6e-04</td>
</tr>
<tr>
<td>SA + CR</td>
<td>0.233±1.3e-03</td>
<td>0.197±7.8e-04</td>
<td>0.215±6.7e-04</td>
</tr>
<tr>
<td>SA + DR</td>
<td>0.249±5.5e-03</td>
<td>0.209±1.7e-03</td>
<td>0.229±3.0e-03</td>
</tr>
<tr>
<td>SA + DR + CR</td>
<td>0.249±5.2e-03</td>
<td>0.211±1.7e-03</td>
<td>0.230±1.8e-03</td>
</tr>
</tbody>
</table>

7.4.9 The contribution of each objective function

The proposed method is based on the composition of several objective functions that help to model the two proposed semantic spaces. In this section, we evaluate the contribution of each objective in the enrichment of the semantic representations and in the final performance. Table 7-3 presents the average mAP performance in cross-modal retrieval using the semantic space (H) for different model configurations.

Table 7-3 shows some of the possibles combinations of the objectives in H space, trained with positive and negative samples. The results show that each additional objective helps to over improve the performance in average. Even so, in H space the best performance is not achieved when all objectives are used together. Table 7-4 presents the retrieval results by searching in Y space. Evidently, in this case, the reconstruction of the label representation is mandatory and the addition of a strategy that helps to model a mapping between the different modalities is also required. Both, semantic alignment and cross-reconstruction, serve this purpose, but together, they complement each other even more, by further improving the performance. The results confirm that the space generated for the semantic annotations is more suitable to perform the retrieval. Again, although Semantic alignment (SA) and Label Reconstruction (LR) present the greatest contribution the overall performance, the best value is achieved when all objectives are used in the lost function.

7.4.10 Kernel evaluation

A suitable selection of the kernel function is crucial for the correct modeling of the common semantic space, which will be reflected in the final performance. Table 7-5 shows the results for several kernel selections. It is interesting to see that, in this case, the RBF kernel, which is considered the de facto standard due to its homogeneous behavior and its very good approximation capabilities, presents in this dataset a poor performance, this is due to RBF Kernels are useful for smooth functions, whereas histogram style features such as bag of features based representations require a kernel that can handle discrete features. This is the case of HI and $\chi^2$ kernels that present the best results.
7.4 Experimental evaluation

Table 7-4: MAP scores for cross-retrieval over the semantic label space (Y) for different combinations of the objectives (SA: Semantic alignment; DR: Direct Reconstruction; CR: Cross-reconstruction; LR: Label Reconstruction).

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Img2Txt</th>
<th>Txt2Img</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR</td>
<td>0.298±1.2e-03</td>
<td>0.260±4.7e-04</td>
<td>0.279±8.5e-04</td>
</tr>
<tr>
<td>LR + SA</td>
<td>0.326±4.1e-04</td>
<td>0.261±5.8e-04</td>
<td>0.293±1.8e-04</td>
</tr>
<tr>
<td>LR + SA + CR</td>
<td>0.327±4.0e-04</td>
<td>0.262±3.4e-04</td>
<td>0.294±2.4e-04</td>
</tr>
<tr>
<td>LR + SA + DR</td>
<td>0.328±4.6e-04</td>
<td>0.261±4.9e-04</td>
<td>0.295±4.0e-04</td>
</tr>
<tr>
<td>LR + DR + SA + CR</td>
<td><strong>0.330±1.4e-04</strong></td>
<td><strong>0.263±1.6e-05</strong></td>
<td><strong>0.296±7.4e-05</strong></td>
</tr>
</tbody>
</table>

Table 7-5: Comparison in mAP scores of MKSE-CM with different kernel functions.

<table>
<thead>
<tr>
<th>Method</th>
<th>( k(a, b) )</th>
<th>Img2Txt</th>
<th>Txt2Img</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>linear</td>
<td>( a^Ty )</td>
<td>0.227</td>
<td>0.203</td>
<td>0.215</td>
</tr>
<tr>
<td>cosine</td>
<td>( \frac{a^Tb}{|a||b|} )</td>
<td>0.279</td>
<td>0.268</td>
<td>0.274</td>
</tr>
<tr>
<td>RBF</td>
<td>( \exp(-\gamma |a - b|^2) )</td>
<td>0.264</td>
<td>0.180</td>
<td>0.222</td>
</tr>
<tr>
<td>HI</td>
<td>( \sum_i \min(a_i, b_i) )</td>
<td>0.322</td>
<td>0.242</td>
<td>0.282</td>
</tr>
<tr>
<td>( \chi^2 )</td>
<td>( \exp(-\gamma \sum_i \frac{(a_i-b_i)^2}{(a_i+b_i)^2}) )</td>
<td><strong>0.330</strong></td>
<td><strong>0.263</strong></td>
<td><strong>0.296</strong></td>
</tr>
</tbody>
</table>

7.4.11 Budget impact

The definition of a budget restriction leads to our method to reduce significantly the algorithm complexity and to keep low computational requirements for training in large-scale collections. In order to observe the impact of the budget size, we evaluated the performance of the proposed MKSE-CM in terms of mAP across different budget sizes. At each budget configuration, we reported the average mAP for both tasks: Image to text and text to image, after running ten different experiments, also the standard deviation is reported. For each run, the model is trained with a different randomly selected budget, this is for the purpose of observing the sensibility of the method to different budget sets constructed with different random samples.

From Figure 7-3, it is quite evident that the mAP improves when more samples are included in the construction of the budget. However, we can see that the contribution in improving performance is becoming less significant as the size of the budget increases, and the maximum mAP is achieved with a budget size composed of about 1500 samples, after this amount of samples, the mAP remains relatively constant. Therefore, only the 69% of the training samples are required to construct a suitable budget, which implies a significant reduction in the amount of required memory. This result not only implies less memory but also less training time. In Figure 7-3, we can also see that the variation in performance is quite not significant to the random samples used to construct the budget, showing that this basic
strategy for selecting the budget elements is fairly suitable, taking into account that other more elaborated strategies would imply greater computational expenses.

### 7.4.12 Comparison with the state-of-the-art methods

Finally, in Table 7-6, the proposed method is compared against all the baseline methods described in subsection 7.4.6 that use as input the basic descriptors proposed in [104] (i.e., Images representation based SIFT features and text representation with LDA). The results show that the proposed MKSE-CM achieves in average the state-of-the-art when the $\chi^2$ kernel is used for both modalities. Furthermore, it is important to highlight that this result is obtained by using a budget composed of 1500 instances randomly selected (this is about the 69% of the training instances).

### 7.4.13 Evaluation with high quality descriptors

The classical benchmark in cross-modal retrieval over the Wikipedia dataset proposed a set of basic descriptors for each modality that reduce the efficacy of the proposed methods and limits the maximum performance that could be achieved. This is due to the loss of information caused by the use of poor descriptors. For this reason, the most recent works have begun to use more sophisticated sets of characteristics [98], [112], this allows them to present remarkable outperforms. In this section, we propose to employ a richer feature set for each modality that would allow to construct a better common semantic space.
Table 7-6: Comparison in mAP scores of MKSE-CM against baseline methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>Img2Txt</th>
<th>Txt2Img</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>BITR</td>
<td>0.222</td>
<td>0.171</td>
<td>0.196</td>
</tr>
<tr>
<td>CCA</td>
<td>0.249</td>
<td>0.196</td>
<td>0.222</td>
</tr>
<tr>
<td>CCA+SMN</td>
<td>0.277</td>
<td>0.226</td>
<td>0.251</td>
</tr>
<tr>
<td>CFA</td>
<td>0.246</td>
<td>0.195</td>
<td>0.220</td>
</tr>
<tr>
<td>CMCP</td>
<td>0.326</td>
<td>0.251</td>
<td>0.288</td>
</tr>
<tr>
<td>DCMIT</td>
<td>0.277</td>
<td>0.250</td>
<td>0.263</td>
</tr>
<tr>
<td>HSNN</td>
<td>0.321</td>
<td>0.251</td>
<td>0.286</td>
</tr>
<tr>
<td>JGRHML</td>
<td>0.329</td>
<td>0.256</td>
<td>0.292</td>
</tr>
<tr>
<td>JRL</td>
<td>0.339</td>
<td>0.250</td>
<td>0.294</td>
</tr>
<tr>
<td>LGCFL</td>
<td>0.274</td>
<td>0.224</td>
<td>0.249</td>
</tr>
<tr>
<td>ml-CCA</td>
<td>0.269</td>
<td>0.211</td>
<td>0.240</td>
</tr>
<tr>
<td>mv-CCA</td>
<td>0.271</td>
<td>0.209</td>
<td>0.240</td>
</tr>
<tr>
<td>MKSE-CM (χ²)</td>
<td>0.330</td>
<td>0.263</td>
<td>0.296</td>
</tr>
</tbody>
</table>

7.4.14 Image representation

The images are described by propagating them into the pretrained VGG model (16 layers) [117], and taking the activations of the penultimate layer as feature vector given an initial 4,096-dimensional vector representation. In order to reduce the computational requirements, a dimensionality reduction via PCA was performed by taking the largest 512 components (this ensure to preserve the 99% of the variance while only an eighth of the original dimensionality is required).

7.4.15 Text representation

The text information is described by using the pretrained Word2Vec model trained on Google News data composed of 1000 billion words [91], for each instance, the final text document representation is the average of all the contained words projected into the 300-D vector representation given by the model.

Table 7-7 presents the retrieval results obtained by the most recent proposed methods, all these methods proposed several variations for the modalities representation. In S²UPG the training instances are modified by using a patch strategy that extracts the same kind of descriptors to parts or patches, of the text or image instances giving some spatial information. For SCM [98] the same descriptors are used, but, the codebook sizes were highly extended: for images is used a 4096-D SIFT descriptor and for text a 200-DLDA. In DCCA-PHS [112], which is a deep version of CCA it is proposed for the images a combination of three descriptors, composed for a 1000-D pyramid histogram of dense SIFT, a 512-D Gist, and a 784-D MPEG-7 descriptor, and for the text a high dimensional 3000-D bag of high-frequency
words.
The results show that the combination of the proposed rich descriptors and the semantic kernel embedding method achieve the state-of-the-art in cross-modal retrieval in this dataset. Moreover, the results show that with the use of an appropriate kernel function, the proposed MKSE-CM achieves a significant improvement without the necessity of use complex deep learning architectures. In this case, Unlike the original histogram based features, kernel functions such as HI and $\chi^2$ are not appropriated due that these kernels cannot handle negative values, and for this kind of smooth representation the most appropriate function is the (Gaussian) radial basis function kernel (RBF). Moreover, another important result is that this performance is obtained with a budget composed of only 500 instances randomly selected, this is only about the 23% of the training samples, showing that starting from a richer feature set a richer semantic space can be modeled with a very small budget, making the model even more simple, but keeping the strength of kernel methods, allowing a complex non-linear modeling, which is evidenced by the increase in the average performance of about 5 percentage points from the linear to the RBF version of the algorithm.

### 7.5 Conclusion and future work

In this section, we presented the MKSE-CM, a novel cross-modal retrieval method based on kernel matrix factorization that constructs a common semantic representation to perform the retrieval process. This semantic representation is enriched by several constraints based on the reconstruction of the original features representation and the prediction of semantic concepts that ensure a suitable space to perform the searching based on classical distance measures.

One of the remarkable aspects of the proposed method is its scalable architecture, which despite being based on kernel methods scales well to deal with large datasets, thanks to its

<table>
<thead>
<tr>
<th>Method</th>
<th>Representation</th>
<th>Img2Txt</th>
<th>Txt2Img</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCM [98]</td>
<td>BoW(4098)</td>
<td>0.362</td>
<td>0.273</td>
<td>0.318</td>
</tr>
<tr>
<td>S²UPG [97]</td>
<td>BoW+PGR</td>
<td>0.377</td>
<td>0.286</td>
<td>0.331</td>
</tr>
<tr>
<td>DCCA-PHS [112]</td>
<td>PBoW+Gist+MPEG-7</td>
<td>0.341</td>
<td>0.379</td>
<td>0.360</td>
</tr>
<tr>
<td>3view-DCCA [112]</td>
<td>PBoW+Gist+MPEG-7</td>
<td>0.364</td>
<td>0.441</td>
<td>0.403</td>
</tr>
<tr>
<td>MKSE-CM (linear)</td>
<td>VGG</td>
<td>0.451</td>
<td>0.413</td>
<td>0.432</td>
</tr>
<tr>
<td>MKSE-CM (cosine)</td>
<td>VGG</td>
<td>0.467</td>
<td>0.431</td>
<td>0.449</td>
</tr>
<tr>
<td>MKSE-CM (RBF)</td>
<td>VGG</td>
<td>0.493</td>
<td>0.467</td>
<td>0.480</td>
</tr>
</tbody>
</table>
learning-in-budget strategy and its online learning formulation, that allows a straightforward efficient implementation in common dataflow GPU frameworks. Experiments over the Wikipedia dataset show that the proposed method achieves the state-of-the-art performances in mAP when the original low-quality features are used with any alteration and can take advantage of the extra information of high-quality descriptors obtaining remarkable outperforms in comparison with other complex state-of-art latent space embedding methods in the task of cross-modal retrieval.
8 Conclusions

8.1 Multimodal representations

As part of this thesis, several algorithms were proposed, focused on solving different data modeling and analysis tasks such as dimensionality reduction, multi-class classification, multi-label annotation and cross-modal retrieval, over different machine learning strategies: supervised, unsupervised, semi-supervised and transductive. In all of these scenarios, we could observe that the complementary information extracted from a second modality helps to construct a richer model in the learning process. No matter the nature of the second modality, this presents a complementary source of knowledge that can be harnessed. Structured data can be used to overperform the results as we shown in the supervised setup, but also, incomplete information can help to better modeling the problem as was shown in the semi-supervised and transductive experiments, and even, unstructured data can be modeled as the cross-modal retrieval experiments confirmed. There are several alternatives to model multimodal information, but in this thesis, we opt for semantic-based strategies, this is because methods based on this kind of strategies can capture the true essence of multimodal data collaboration as all the features from different modalities are combined in a unified representation. This is what has allowed addressing a broad range of data analysis tasks. Challenging tasks such as semi-supervised annotation or cross-modal retrieval helped to show how it is possible to take advantage of the complementary information of each modality to construct a new richer representation even if this information is incomplete or noisy. The richness of this new space has been proved by means of a complete experimentation by evaluating the proposed algorithms against several baseline methods for each task. And most importantly, for each of these tasks, the proposed algorithms have achieved competitive results, and in several cases, these have surpassed the state-of-the-art.

8.2 Learning multimodal non-linear relationships

Linear methods present several restrictions that do not allow them to model complex relations properly, and give a poor approximation to the problem. Multi-modal information presents data from different sources that are represented with different strategies in different representation spaces. Due to this complexity, a linear method will present a very poor
modeling of the problem that is reflected in a poor performance in a specific task. In this thesis, a kernel-based strategy was used as a way to tackle this problem. This implies some advantages but also some challenges. Among the advantages, we can highlight that kernel-based methods are mathematically and statistically well-founded, they usually present accurate and robust results in many tasks. The kernel based matrix factorization methods proposed in this thesis are a natural non-linear extension of latent semantic methods. Another important characteristic is that through the kernel representation, we can take advantage of extra domain information. For instance, according to the original feature representations, we can define the kernel functions. For instance, this is shown when we are dealing with histogram-like representations such as bag of words or bag visual of features, were kernels such as histogram intersection or $\chi^2$ produce better results. Finally, throughout all the experimentation it has been corroborated how the kernelized version of the proposed methods can generate a richer representation compared to their linear counterparts by achieving better results in different data analysis tasks.

On the other hand, most of the challenges in using kernel-based semantic embedding methods are related to scalability issues. These scalability issues are related in one way with the classical learning strategies that do not allow an online optimization, and in another way with the high computational demand of kernel representations.

8.3 Large-scale multimodal kernel semantic embedding learning

8.3.1 Scalable semantic space construction

One of the main strategies used in this thesis to achieve scalability is the reformulation of the matrix factorization problem from the classical optimization based on multiplicative rules to additive rules that allows performing the optimization based on stochastic gradient descent. This implies a reduction in the memory requirements due to the full training data is not required in full to perform the actualization of the parameters at each iteration. This is also possible due to the reformulation of the factorization that does not requires the explicit calculation of the encoding matrix $H$. This implicit calculation of the semantic space was inspired in the original two-way linear formulation, which in turn was inspired in the idea of autoencoders in neural networks.

8.3.2 Scalability of kernel-based methods

The main characteristic of kernel-based methods is that they require the calculation of all pairwise similarity values between the elements in the training dataset. This implies quadratic space and time complexities to compute a kernel matrix. This leads to two computational problems: first, the amount of required memory to represent the kernel matrix,
becomes enormous and intractable, and, second, the processing time of the kernel-based algorithms grows faster, making them infeasible for real-life applications. In order to tackle this problem, a learning-in-a-budget strategy was proposed. In this strategy the calculation of all pairwise similarities is not required, instead of that, for every single element in the dataset, its similarity is calculated with just a subset of \( b \) significant elements from training. In this thesis two main strategies to construct this subset were evaluated:

1. Random selection: to construct the sub set, a simple random selection of \( b \) (the budget size restriction) instances is applied to compose the budget set.

2. K-means: Instead of selecting instances, a set of prototypes is constructed by applying the K-means method where the number of clusters is set to \( b \) and the final founded centroids are used as new instances to construct the budget set.

An interesting funding is that for a large enough \( b \) values, a significant difference in performance is not found between both alternatives, so a suitable budget can be constructed just by the simple random selection, avoiding the additional computational burden of performing K-means.

The most important finding about the budget strategy is that the maximum performance in the proposed algorithm in all the evaluated tasks is achieved with a reduced \( b \) value that is significantly lower than the total number of elements in the training set. Furthermore, the experimental results show that the bigger the training set the smaller is the proportion between the \( b \) value and the total number of elements in the training set. This implies that no matter the size of the dataset, thanks to the online learning strategy and the budget restriction the memory requirements are kept low. Additionally, our proposed method is able to improve the task performance of baseline methods that do require the complete construction of the kernel matrix.

### 8.4 Future research directions

#### 8.4.1 Regarding the budget construction

There is an open question, that is, it is possible to reduce the size of the budget by the selection or construction of representative instances following a more robust strategy? A reduced budget implies not only less memory but also less time in the training process.

#### 8.4.2 Combination with deep learning models

One of the more remarkable characteristics of the most elaborated methods developed in this thesis is their straightforward implementation over end-to-end architectures. This allows to use and incorporate neural network components and strategies in a simple way, such as
classical loss functions, activations functions, and update rules, among others. It would be interesting to see how this method can be integrated with more complex deep learning architectures.

### 8.4.3 hyper-parameter exploration and selection

In the last section of this thesis, we showed how taking advantage of the fully gradient descent formulation of the proposed method we are able to learn the kernel parameters. This is very useful, by taking into account that the adequate selection of this value is crucial in the final performance. In the same way, there is another group of hyper-parameters that present a big impact in the final performance, in some cases the exploration of these hyper-parameters is a demanding task. Therefore, it is important to study and understand the behavior of each one these hyper-parameters to try to predict the most suitable values according to the characteristics of the specific task to solve and the domain, instead of using a demanding brute force grid exploration.
Bibliography

[1] Abadi, Martín; Agarwal, Ashish; Barham, Paul; Brevdo, Eugene; Chen, Zhifeng; Citro, Craig; Corrado, Greg S.; Davis, Andy; Dean, Jeffrey; Devin, Matthieu [u. a.]: Tensorflow: Large-scale machine learning on heterogeneous distributed systems. In: arXiv preprint arXiv:1603.04467 (2016)


[14] Bergstra, James; Breuleux, Olivier; Bastien, Frédéric; Lamblin, Pascal; Pascanu, Razvan; Desjardins, Guillaume; Turian, Joseph; Warde-Farley, David; Bengio, Yoshua: Theano: a CPU and GPU Math Expression Compiler. In: Proceedings of the Python for Scientific Computing Conference (SciPy), 2010. – Oral Presentation


[34] Cotter, Andrew; Shamir, Ohad; Srebro, Nathan; Sridharan, Karthik: Better Mini-Batch Algorithms via Accelerated Gradient Methods. In: CoRR abs/1106.4574 (2011)


[41] Feng, Fangxiang; Wang, Xiaojie; Li, Ruifan: Cross-modal Retrieval with Correspondence Autoencoder. In: Acm Mm (2014), S. 7–16. – ISBN 9781450330633


[45] GONG, Chunying ; LIU, Tiegen ; TAO, Dacheng ; FU, Keyuan ; Tu, Enmei ; YANG, Jian: Deformed Graph Laplacian for Semisupervised Learning. In: IEEE Transactions on Neural Networks and Learning Systems (2015)


[48] GOODFELLOW, Ian J. ; WARDE-FARLEY, David ; LAMBLIN, Pascal ; DUMOULIN, Vincent ; MIRZA, Mehdi ; PASCANU, Razvan ; BERGstra, James ; BASTIEN, Frédéric ; BENGIO, Yoshua: Pylearn2: a machine learning research library. In: CoRR abs/1308.4214 (2013)


[63] Jolliffe, Ian: Principal component analysis. Wiley Online Library, 2002


[77] Li, Dongge ; Dimitrova, Nevenka ; Li, Mingkun ; Sethi, Ishwar K.: Multimedia content processing through cross-modal association. In: *Proceedings of the eleventh ACM international conference on Multimedia* ACM, 2003, S. 604–611

[78] Li, Mu ; Bi, Wei ; Kwok, J T. ; Lu, Bao-Liang: Large-Scale Nyström Kernel Matrix Approximation Using Randomized SVD. In: *Neural Networks and Learning Systems, IEEE Transactions on* 26 (2015), Nr. 1, S. 152–164. – ISSN 2162–237X


[81] Lin, Wanxia ; Lu, Tong ; Su, Feng: A Novel Multi-modal Integration and Propagation Model for Cross-Media Information Retrieval. In: Schoeffmann, Klaus (Hrsg.) ;


[86] LIU, Y ; CHEN, Y ; ZHU, L: Tensor-based video categorization via sparse representation. In: Advances in Information Sciences and Service Sciences 3 (2011), Nr. 8, S. 8–15

[87] LU, Zhiyun ; MAY, Avner ; LIU, Kuan ; GARAKANI, Alireza B. ; GUO, Dong ; BELLET, Aurélien ; FAN, Linxi ; COLLINS, Michael ; KINGSBURY, Brian ; PICHENY, Michael ; SHA, Fei: How to Scale Up Kernel Methods to Be As Good As Deep Neural Nets. (2014), November


[104] Rasiwasia, Nikhil; Costa Pereira, Jose; Coviello, Emanuele; Doyle, Gabriel: A new approach to cross-modal multimedia retrieval. In: Mm (2010), S. 251–260. ISBN 9781605589336


[112] Shao, Jie; Wang, Leiquan; Zhao, Zhicheng; Cai, Anni [u. a.]: Deep canonical correlation analysis with progressive and hypergraph learning for cross-modal retrieval. In: Neurocomputing 214 (2016), S. 618–628


[120] SMEULDERS, A W M. ; WORRING, M ; SANTINI, S ; GUPTA, A ; JAIN, R: Content-based image retrieval at the end of the early years. In: IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (2000), Nr. 12, S. 1349–1380. – ISSN 01628828


[125] TAN, Shoubiao ; HU, Lingyu ; WANG-XU, Anqi ; TANG, Jun ; JIA, Zhaohong: Kernelized cross-modal hashing for multimedia retrieval. In: WCICA, 2016 12th World Congress on IEEE, 2016, S. 1224–1228


[129] Tsai, Min-Hsuan; Wang, Jinjun; Zhang, Tong; Gong, Yihong; Huang, Thomas S.: Learning semantic embedding at a large scale. In: ICIP, 2011, S. 2497–2500


[136] Wang, Can; Zhao, Jun; He, Xiaofei; Chen, Chun; Bu, Jiajun: Image retrieval using nonlinear manifold embedding. In: Neurocomputing 72 (2009), Nr. 16, S. 3922–3929

[137] Wang, Hua; Huang, Heng; Ding, Chris: Simultaneous Clustering of Multi-type Relational Data via Symmetric Nonnegative Matrix Tri-factorization. In: Proceedings

[139] Wang, Lijun; Rege, Manjeet; Dong, Ming; Ding, Yongsheng: Low-rank kernel matrix factorization for large-scale evolutionary clustering. In: Knowledge and Data Engineering, IEEE Transactions on 24 (2012), Nr. 6, S. 1036–1050


[149] Zhai, Xiaohua; Peng, Yuxin; Xiao, Jianguo: Heterogeneous Metric Learning with Joint Graph Regularization for Cross-Media Retrieval. In: AAAI, 2013


[154] Zhang, Lijun; Jin, Rong; Chen, Chun; Bu, Jiajun; He, Xiaofei: Efficient Online Learning for Large-Scale Sparse Kernel Logistic Regression. In: Hoffmann, Jörg (Hrsg.); Selman, Bart (Hrsg.): AAAI, AAAI Press, 2012


[156] Zhang, Min-Ling; Zhou, Zhi-Hua: Multilabel Neural Networks with Applications to Functional Genomics and Text Categorization. In: IEEE Transactions on Knowledge and Data Engineering 18 (2006), Nr. 10, S. 1338–1351. – ISSN 1041–4347

[157] Zhao, Mingbo; Zhang, Zhao; Chow, Tommy W.; Li, Bing: A general soft label based Linear Discriminant Analysis for semi-supervised dimensionality reduction. In: Neural Networks 55 (2014), S. 83–97

[158] Zhao, P; Hoi, SCH: Bduol: Double updating online learning on a fixed budget. In: Machine Learning and Knowledge Discovery in Databases (2012)
[159] ZHOU, Dengyong ; BOUSQUET, Olivier ; LAL, Thomas N. ; WESTON, Jason ; SCHÖLKOPF, Bernhard: Learning with local and global consistency. In: Advances in neural information processing systems 16 (2004), Nr. 16, S. 321–328


