Interactive learning of words and objects for a humanoid robot

Yuxin Chen

NNT : 2017SACLY003

Thèse de doctorat
de
l'Université Paris-Saclay
préparée à
ENSTA-ParisTech

Ecole doctorale n°573
Approches Interdisciplinaires, Fondements, Applications et Innovation (INTERFACES)
Spécialité de doctorat : Informatique

par

M. Yuxin CHEN

Apprentissage interactif de mots et d'objets pour un robot humanoïde

Thèse présentée et soutenue à Palaiseau, le 27 Février 2017.

Composition du Jury :

M.	Chen Yu	Professeur Indiana University Bloomington	(Rapporteur)
M.	Mohamed Chetouani	Professeur Université Pierre et Marie Curie	(Rapporteur)
M.	Alexandre Pitti	Maitre de conférence ÉNSEA - Université de Cergy Pontoise	(Examinateur)
Mme.	Adriana Tapus	Professeur ÉNSTA ParisTech - Université Paris Saclay	(Présidente)
M.	David Filliat	Professeur ÉNSTA ParisTech - Université Paris Saclay	(Directeur de thèse)
M.	Jean-Baptiste Bordes	Docteur École Polytechnique - Université Paris Saclay	(Encadrant de thèse)

Acknowledgement

It has been long time that I wish I could have the chance to study in a country where elegance, romance, challenges and wonders have been rooted for centuries…

First of all, I really appreciate the chance that opens a new door for me, under the aegis of CSC scholarship that supports also a lot of other excellent Chinese youth to see and embrace the whole world. Yet, without the appreciation of David Filliat for what I had temporarily achieved during my master period which sharply formed the burning motif of studying in depth the potential intelligence of algorithms, I wouldn't have been here for another fully engaged life of three years… So this very first thanks should be given to my *supervisor of thesis* !

Life always continues like a mingled pair, either of *pain* and *gain*, *frustration* and *aspiration*, *conservation* and *exploration* or *degradation* and *rejuvenation*… It still reminds to me the warning from David - "three years in a project which is not of your interest would be a dangerous thing" and my response - "I would be willing to find my own interest once a project is presented to me" that have deeply engraved in my memory of the very first skype interview between us. If the truth is that I was too young at that age, the fact would be I am still young for my endless pursuing and breeding-up the intuition, not only rendering me the consciousness and courage to do better in everyday affaires but also giving birth to the inspiration and interest necessary for the academic endeavors.

Undoubtedly, at least from my own view point, the current intelligence limit of a robot reflects the level of cognition and the range of wisdom that its developer(s) would possess. Being aware of this, I would rather conduct a kind of living in which both life and academy are integrated as well as interactive so as to achieve a timely refreshed mindset. And more interestingly, similarities in terms of mechanism can be found by comparing some certain topics, inspired by the power of algorithms and exhibited by wisdom of humans, such as the field of *information fusion* during master study and the *concept learning* of my Ph.D. In a larger sense, the area in which I am engaged, the *developmental robotics*, is just interdisciplinary, thus the new wonder and challenge stand and I am looking forward to making my next difference.

Very luckily yet unexpectedly, Jean-Baptiste Bordes joined my research in the midway as the *encadrant de thèse*, giving me brand new suggestions about doing comparison studies between NMF and LDA and therefore opening door to a broader horizon which greatly enriched my Ph.D project. Here, special thanks are for Jean-Baptiste and his patience as well as devotion to help me arrive at where I am today.

Finally, I am grateful to my families, relatives, friends for all the lasting love and care as well as support…Because life is neither said to be perfect nor predetermined, and success is always cumulated, so is accomplishment through devotion. Therefore, I have nothing more to rely on but the appreciation of everything good, however tiny it might be, from YOU !

俯仰穷天地，西楼望断东。
义生原素理，象化应约综。
龙见三才贯，犀灵一点通。
衣宽憔悴敛，灯火夜阑空。

March 1st, 2017

Contents

Introduction - A developmental perspective on robotic intelligence

The ability to continuously adapt to their environment and to learn to recognize new objects will be fundamental for future service robots if they eventually operate in everyday environments and in interaction with people. Research in computer vision for object detection made tremendous progress in the past decade thanks to the apparition of efficient visual features and the improvement of machine learning models. However, performances are still limited for autonomous robot operation and moreover heavily rely on the availability of good training data that are by themselves difficult to obtain. In contrast, two year old children have an impressive ability to learn to recognize new objects and at the same time to learn the object names during interaction with adults and without precise supervision. We propose in this PhD thesis to develop object and name learning approaches inspired by the children capabilities following the developmental robotics approach. The goal will be to build computational models which make it possible to match objects to words for a humanoid robot using human interaction similar to the interaction taking place between children and parents.

With this objective, we will build on several researches recently conducted regarding multi-modal object discovery using sound and images and we compared two topic models in the framework of cross-situational learning to tackle ambiguities in the supervision. We also studied how active learning strategies could improve performances and compare them, in terms of behaviors, with human learning.

Robot learning in social context

Among the scientific breakthrough during the past two centuries, the invention of computer, followed by the prosperity of computer science, has not only made huge progress within its own domain but also revolutionize interdisciplinary subjects with other fields, such as biology, psychology and cognitive science. Taking direct advantage of ever-changing technology and algorithm progress, the traditional generation of robotics has successfully been applied to factory manufacturing, automatics and controlling systems. It exhibits superior performances in well-predefined, targeted and repetitive tasks, executing precisely the orders programmed by humans.

However, despite the computational and logical power it possesses, computer seems weak at acquiring or even approximating the human intelligence which we humans take for granted, for instance in terms of graphical pattern recognition, CAPTCHA Program has been launched since 1996 to successfully prevent computer's controlling the access of web service [1]. Therefore, if robotics really intends to evolve beyond the industrial level, from pure mechanical equipment

[1] http://en.wikipedia.org/wiki/CAPTCHA#Origin_and_inventorship

to life-like human companions or partners in the domain of personalized service, it has to come closer to how human intelligence works, and in particular, to be endowed with some basic mechanisms (existing during the growth of infants' first several years after birth) to learn new skills.

Supported by mathematics or inspired by biological phenomena, some intelligent algorithms that serve as the brain of robot have been proposed to simulate the biological intelligence, from both the structural (e.g. Neural Networks [78], Genetic Algorithm [46], Ant Colony Optimization [53], Physarum Algorithm [205]) and functional (e.g. D-S theory of evidence [49], Non-negative Matrix Factorization [112, 113]) perspectives. These methods alone could serve as effective tools and achieve satisfactory results in some particular tasks. Yet due to their use as tools, it is impossible to expect any such algorithm alone to go beyond what has been initially envisioned by the robot developer.

Therefore, social learning mechanisms are studied to exploit the human-robot interaction as a means to upgrade the "wisdom" of robot, since this kind of social-guided learning could help the robot take full advantage of human while acquiring new skills. Some biologically inspired social learning mechanisms include emulation, mimicking, imitation, stimulus enhancement [189, 28], and currently, one of the issues for this domain still not satisfactorily solved is to decide what is the trade-off between the amount of guidance (social) and exploration (self), because social interaction usually requires a high level of human involvement.

As for guidance-based methods, [76, 27] for example, propose methods of learning by demonstration, and [158] describes how the robot imitates demonstrated motor actions for learning. Besides, [110] is an example of how natural language works for guidance based learning. Generally speaking, if the social interaction is highly dependent on guidance, no matter what the differences among each methods [188], the human partner should be at the expert level, knowing not only the way of communicating with robot but the clear picture of how robot would perform in the task as well.

The exploration based method, on the contrary, requires much simpler signal from human participants to adjust the learning procedures. [187, 91] seek to control the reward signal for the reinforcement learning, and the role of human advice is discussed in [121]. Similar works of the method in [121] also include the control over agent's action during training [172]. While the laborious step-by-step human efforts are reduced at a large scale, the contribution of participants' involvement are much limited and the human tutors should know exactly the tricks to convey the message effectively to the robot for improving learning.

For improving the social learning for robotics, the ultimate goal is simple to state: the robot should be able to learn from a human without special expertise, through progress little by little, and discover features and traits by itself from seemingly ordinary information source. In this framework, a human as tutor or a robot as learner (not excluding its possible role of tutor as well), are embedded in a social environment, structured like a network where every participant should be aware about its optimal goal (eg. of whether becoming a tutor or learner at the moment) and how to make an optimized use of current resources in an adaptive learning manner. As one of pioneering works following this route, the Never Ending Learning

[132] has initiated the study of paradigm that is supposed to help robot handle many different types of knowledge from diverse experiences with the learning duration over many years, by reasoning over its beliefs.

Without going this far in this research direction, we will focus our work on scenarios where we take advantage of simple and natural supervision by a human teacher in order to learn words and their corresponding definitions in the visual domain. We will therefore perform experiments using only data recorded from naive human teachers that do not have special knowledge of the underlying algorithm used for learning.

Word and object learning

Word and object learning, on top of which effective communication could be established, is one of the bases of cognition in a social context. The objective of word and object learning can be defined as object or feature labeling by using corresponding words, or as word-referent matching. The open issue lies in the situation that in ordinary teaching situations, if no constraint is imposed, there could exist multiple mappings from word to referent, known as "indeterminacy of reference" problem, proposed by [145].

Regardless of the stage of life, whether childhood or adulthood, there are respectively specific strategies and mechanisms which appear active and primarily applied to decrease or even eradicate the issue of referential indeterminacy during different ages. And recent studies which aim at decreasing the uncertainty of potential referents involve two main directions: cross-situational [144, 3] and social learning [8, 9]. Cross-situational learning, as an unsupervised learning method, means that the learner does not receive feedback on its performance but merely relies on finding and analyzing common factors between different ambiguous situations, however, the supervised social interactive learning requires that the learner receive feedback during interaction [30]. And both methods indicate that word-object learning could not be accomplished at one stroke: on one hand, cross-situational learning provides multiple scenes, which are relating to one specific word, the learner is supposed to make use of the the intersection of meanings among those scenes to get the possible referents; on the other hand, interactive learning also has the request that the teacher conduct sufficient times of guidance over potential mapping for the learner.

Despite the mechanisms and strategies already proposed, the problem of ambiguity still sets up obstacles for the word's referent learning. There are two main types of ambiguities that the models and experiments described in the remaining of this thesis will tackle:

1) *Referential ambiguity*: Imaging a simple situation in which an object has features in different modalities, for example shape and color, and a minimal description of this object contains two keywords. The referential ambiguity exists in the two possible correspondences between the words and the features. This ambiguity is even more present in cases in which multiple objects are presented to the learner

while only one general descriptive sentence is given by the teacher: the mapping of a set of keywords (among many) to its corresponding object (among many) is also undefined;

2) *Linguistic ambiguity*: The sentences for the description of an object or a set of objects might also contain not only keywords but other grammatical words, mood words, or speaking errors as well, so the algorithm should try to distinguish keywords from other words considered as noise in this context.

A developmental robotics perspective

The goal of developmental robotics is to develop robots that are able to learn incrementally a large variety of tasks by taking inspiration on the way human children develop and learn. It studies ways of transferring the mechanism of how humans (especially infants) gradually acquire knowledge and cognitive abilities to the applications of robotics. For early age children, it is an amazing show of intelligence and curiosity to see them observing the surroundings, recognizing their own bodies, exploring and interacting with the physical world. On one hand, children can discover an object by manipulation and exploration, analyzing its size, shape, texture, color, etc.; while on the other hand, they show an ingenious language learning ability [107, 106] from birth to 3 or 4 years old. And the progress made by these efforts will be far more enhanced if effective interaction with adults is introduced. A more detailed analysis about the different stages of mental developments of infants can be found in the Piaget's theory of cognitive development [143].

In order to develop a robot with the same types of skills as human infants, six principles from developmental psychology [173] indicate possible directions for further breakthrough: 1)multi-modal learning, 2)incremental development, 3)learning through physical interaction, 4)exploratory behavior, 5)social interaction, 6)symbolic communication.

Alan Turing [191] and other pioneers of cybernetics first proposed the idea that led to developmental robotics. Among these, there is the idea of letting the robot's learning be "autonomous throughout its lifelong mental development" as proposed in [198] and developed by follow-up practices including the aforementioned work of Never Ending Learning [132]. A key observation that we will use in this thesis is the fact that, when facing the vast space of learning, curiosity often guides the infants' exploration and sets constraints for the learning processes. Therefore artificial curiosity [138, 33] was developed for simulating the intrinsic motivation of infants for seeking new information. As an attempt for a comprehensive application of all above theories, the combinatorial learning with social guidance was studied in [33], capable of autonomous new object detection and tracking, sensorimotor coordination, intrinsic motivation and hierarchical learning, which are investigated on the iCub humanoid robot platform.

In our thesis, we will study how curiosity, and more generally active learning, can be used to improve the learning performance of an agent learning word-referent associations.

Objectives

The objective of this Ph.D research, on a general manner, is to establish a framework of interactive learning system by means of computational modeling which is applicable for a humanoid robot, primarily tackling the word-object learning issue by focusing on the two basic examples: the accurate associations between color features and color adjectives along with shape features and nouns, as foundations for more complicated cognitive researches in future for developmental robotics. While other word categories, such as adjectives for size, spatial indication (left, right..) or verbs could be studied, we focus on these simple concepts in order to be able to study in depth the algorithms and the active learning behaviors.

Our concern and interest are concentrating on the inspiration from very young infants, who are at dawn of acquiring external knowledge and have little knowledge about linguistics, especially in terms of syntax and grammar. In this case, other than resorting to ontological tools such as WordNet, our efforts are engaged in the true early phase of cognitive learning, so early that not only feature knowledge appears unknown before but linguistic ability of a learner is supposed to be at its very basic level. What's more, in order to concentrate on the disambiguation potentiality of proposed computational models, we likewise put aside the interaction techniques to decrease the level of ambiguities, such as joint attention or precise pointing which on the contrary lead to easier data and less challenging environment to learn.

Besides, we found that most of the word-object learning studies highlighted the evaluation for effectiveness by resorting to how well they perform in explicit tests. However, what exactly the results are, which the learning stage manages to achieve, is rarely tackled. Therefore, apart from using traditional task performances for evaluation, this thesis is also going to discuss the form of the factorized components as the acquired knowledge with visualized demonstrations as well as their evolution.

In addition, as emphasized previously, since the adaptive learning ability will be at the crux of the future trend of robotics, we instantiate this topic by studying the applications of active learning strategies based on intrinsic motivations to incremental learning processes as an endeavor to promote the learning speed as well as quality, compared to those by just using passive data gathering.

In fact, more challenges stand in the way of building a complete modeling of word-object learning, and the most significant of them are listed in the following:

1. *Feature extraction and presentation*

   The information acquisition for experimental data normally concerns multimodalities: computer vision, audio feature and linguistic symbols. Normally, raw information should be processed into the space of features which are compatible with the algorithms, effective in characterizing, taking less storage and being invariant to the changing conditions before further analyses. In this thesis, vision and audio raw data are dealt with, especially for vision feature and we use two descriptors: a simple one (ie. pixel) and a more complex one (ie. HOG);

2. *Learning algorithms*

   In general, the algorithms applied should prove not only effective and efficient in processing feature data but robust to noise, disturbances and uncertainties as well. In the case of this thesis, we expect the proposed algorithms, apart from the basic objective of discovering feature-word associations, to be compatible with an adaptive environment where special learning strategies are applied, capable of dealing with ambiguities and, in contrast to more traditional supervise learning, less dependent of good and precise training data;

3. *Human-robot interaction*

   Since the research of this thesis is from the inspiration of parent-infant interactions, it would naturally becomes one of the objectives to see if some characteristics of interaction behaviors derived from human-robot communications are in accordance with those from humans. And evidently, this behavioral study is highly relevant to the rules and protocols that we deliberately design as the platform. In our thesis, in order to exploit attentively from the aspect of learning algorithm, in particular related to behavioral studies, we make use of simulations as substitution of real robot-human participations (although applied at the very beginning) to get experimental data more efficiently, on which analysis could be conducted at the statistical level to arrive at more convincing conclusions.

Contributions

This thesis is mainly focused towards complete computational models for word-object learning, in particular in terms of word-feature associations by means of basic interactions between a human teacher and a humanoid robot as a learner, and special concern has been made about the two central issues in the domain of machine learning and developmental robotics: *cross-situational learning* and *active learning* and their applications in a series of systematically designed simulation experiments. The highlights of contributions of the thesis are listed as follows:

- Significant modifications have been made on NMF so as to solve its related two open issues: the determination of number of components as well as the stability of decomposed results that could be regard as simple concepts in the form of "one modality-one word" and adapt the demands of our experiments;
- Based on the idea of topic model, two computational models by applying Non Negative Matrix Factorization and Latent Dirichlet Association respectively have been built, capable of fulfilling tasks of cross-situational learning in which two sort of ambiguities exist: *referential* and *linguistic*;
- Two active learning strategies have been proposed, which are compatible to the proposed learning models, in an effort to benefit the incremental learnings;

- Series of experiments simulating word-object learning in different scenarios haven been conducted, including batch learning and incremental learning, learning with ambiguous/unambiguous or noisy/noise-free data and active learning vs. random learning. The analyses of these experiments are not only for the validation for our proposed algorithms but also contributing to the fundamental studies about cognitive thinking behaviors via the comparison of performances between humans and learning models that really control the actions of humanoid robots;
- In terms of experimental settings, a complete image processing pipeline concerning object segmentation and object feature computation, along with audio to text conversion has been applied in our research.

The majority of the works of the thesis was presented in the following publications:

- Yuxin Chen, Jean-Baptiste Bordes, and David Filliat. "An experimental comparison between NMF and LDA for active cross-situational object-word learning." In: *ICDL EPIROB 2016*. 2016
- Yuxin Chen and David Filliat. "Cross-situational noun and adjective learning in an interactive scenario." In: *2015 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*. IEEE. 2015, pp. 129–134

as well as contributions to workshops as:

- Yuxin Chen, Jean-Baptiste Borde, and David Filliat. "A comparative study on active learning behavior in word-meaning association acquisition between human and learning machine." In: *ICDL-EpiRob 2016 Workshop on Language Learning*. IEEE. 2016
- Fabio Pardo, Yuxin Chen, and David Filliat. "Effects of robot feedback on teacher during word-referent learning." In: *ICDL-EpiRob 2015 Workshop on Mechanisms of Learning in Social Contexts*. IEEE. 2015

additionally, a journal publication is currently under submission.

Overview of the manuscript

In the remainder of this thesis, related works of word-object learning will be presented by in Chapter 1. We then present our own models and experimental settings mainly from aspects of the multimodal signal processing for feature extraction and presentation, the modeling of the the learning behavior by applying and improving topic model algorithms and the introduction of available active learning strategies in Chapter 2.

The experiments starts from Chapter 3, in which batch learning scenarios are established in order to validate our modifications on Non negative Matrix Factorization (NMF), especially

involving the issue of auto-estimation of the NMF dictionary size (ie. number of decomposed factors) and the quality of learned decomposed factors by resorting to the ground truth data. We then focus on incremental scenarios in Chapter 4 where ground truth data are replaced by real interactive testing tasks as the evaluation criteria and our proposed models will be challenged from the simplest experiment to more and more complicated ones by increasing ambiguities and noise as well as adding active learning strategies. In addition, data using a more complex vision feature are implemented to further validate our models and comparative studies between performances of our models and those of humans are also demonstrated.

Finally, Chapter 5 discusses several aspects of our proposed models while chapter 6 gives a short summary of our work as a conclusions before discussing potential future works.

Chapter 1

State of the art

Contents

1.1 Overview of word-referent learning problem

The objective of word-referent learning is to achieve an accurate association between a set of real world elements (for example an object identity or an object feature such as color) and another set of discrete word symbols that are used for communication. This problem, in a larger perspective, can be stated as language grounding [72] or symbol grounding [81, 178, 109, 202].

As described in the "talking head" experiment [179], the goal is to establish the various relations described in a semiotic square picturing the bidirectional connection between an external referent and an utterance through an agent perception and its internal meaning representation (Figure 1.1):

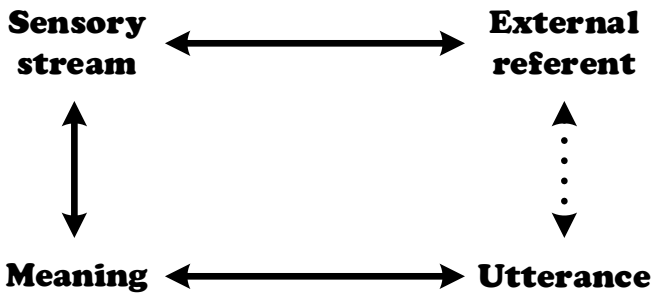


Figure 1.1: Semiotic square linking an uterance and an external reference.

- The **External referent** is an entity in the real world such as an object which will be the subject of the **Utterance**.
- The **Utterance** is the set of words pronounced by an agent which can be complete natural sentences, comprising nouns, verbs, adjectives, articles, etc. Obviously, some of the words are related to the description of the referent while others are related to the grammar or indicate other concepts.
- The **Sensory stream** is the perception of the referent by the agent and depends on the modalities of information source. In most cases, vision is applied to characterize the referent in terms of features (like shape, size, color).
- The **Meaning** is a set of internal discrete symbols representing features from the sensory stream. These symbols have associated **Utterance** that make it possible to express them.

In the frame of the semiotic square described above, language grounding is engaged in the correct mapping between the **Utterance** and the properties of the **External referent** through the mediation of the **Sensory stream** and the **Meaning**. While all parts of these mappings could be learned, we will focus on learning the mapping between the **Sensory stream** and discrete **Meaning** and between **Meaning** and **Utterance**, assuming a fixed mapping between the **External referent** and the **Sensory stream** (but considering noise). In the remainder of this document, we will most of the time refer to this sub-problem as "word-referent learning".

Many researches have been conducted in the field of word-referent learning, with the goal of grounding various contents, including personal pronouns [127, 71], colors, shapes [148], set of multiple properties [126], locations [177] and spatial relations [146, 149, 42, 170, 204, 186, 184, 185]. In our work, we focus on the relatively simple yet primary task of learning words related to object identities as well as features and concentrate on studying various algorithmic aspects that influence the performance of a system performing this task with data obtained by interaction with a human teacher.

Applying word-referent learning in a human-robot interactive scenarios leads to potentially both theoretical and pragmatic problems. We give a short overview of these problems here before detailing them in the following sections:

- **Perception problems**, which can be divided into two parts:

  (a) **speech perception problem**, where individual words are to be properly segmented from natural fluent sentences before they are endowed with real world meanings. The simplest way to carry out the communication is that both human and robot utter only keywords (ie. feature related words, for instances, nouns and adjectives describing attributes), however, in realistic parent-child or human-robot interactions, we seldomly observe that the speaker will just use keywords, but will resort to more complex sentences comprising keywords (nouns and adjectives) plus grammatical words (eg. articles, conjunctions, pronouns, interjections, etc.). In human-robot interaction scenarios, the speech recognition may also frequently produce errors so that the speech modal data seem *noisy* because those grammatical words and spoken errors will not be matched by any feature of real world objects. Several solutions exists that can be categorized as *grammar based* and *statistics based* and will be detailed later;

  (b) **object perception problem**, where objects are to be firstly segmented from the background, and then different feature descriptors (eg. appearance, shape, color etc.) are computed to characterize the objects as the basis for recognition algorithms. Although there are also recent experiments in which more than visual features are used [5, 136], still visual perception plays a predominant role and is still an active research area that is confronted to problems such as varying environmental conditions that produce strong noise in object segmentation and feature extraction.

- **Choice of the word-referent learning algorithms** in order to learn the correct associations between words and features. Various algorithms have been developed in the domain of machine learning, which can be presented in two approaches, either based on *statistics* or on *matrix decomposition*. Not only are the principles different, but they have their own respective pre-processing necessities and preferred working conditions which lead to different integration problems in an interactive learning scenario.

- **Choice of the learning strategy**, which defines the interaction protocol between the learner and the teacher. On one hand, strategies should be devised to limit the ambiguity on the referred object on which to learn, so that the correct word-referent associations can be formed. To solve this problem, there are strategies either directly removing the ambiguity at the moment of perception by using non-verbal communication techniques such as eye gaze coordination [64, 25, 24, 134, 2, 79, 80] and shared attention [1, 157, 156, 48, 59] or, on a more global scale, by decreasing the effects of ambiguity via cross-situational learning (learning through statistics on several situations). On the other hand, humans do not simply learn passively, but have the capability to choose the objects/words they want to learn, especially when then learning behavior is active for a long period of time. Therefore, besides simply learning on a database, it is interesting to perform incremental learning and active learning by choosing the examples so as to learn faster.

- **Choice of the evaluation criteria**, which up to now can be characterized into the

following categories:

(a) **accuracy measure**, also known as the correctness percentage of successful performances in tasks such as categorizing objects [5], recognizing associations across modalities [125, 123], testing linguistic operations (eg. speech segmentation, its correspondence to a real word) [150] and matching word-object pairs [150, 201];

(b) **direct comparison with ground truth raw data**, for example in [136], image retrieval performance is quantified by the root mean square (RMS) errors in comparison with original image patches;

(c) **phenomenal or behavioral observation and analysis**. For instance, in the Talking Heads experiment [178], the frequency of different words expressing a single meaning and the category variance are plotted as the learning goes on. In [174], the statistics are recorded of real human behaviors concerning fixation time to the target against distracter objects, given different conditions and cases;

(d) **other customized criteria** that could be represented by works in Naming Games [164] where the traditional success rate (ie. accuracy rate) is not working considering that the common shared vocabulary among agents might converge to only a subset of all available meanings and other measures like the mean time needed for convergence as well as the average number of meanings per agent does not fit the demand either. As a solution, which is based on information theory [166], the number of possible configurations according to the currently known words and meanings is defined in bits by applying a logarithm and able to represent a coherent distance to global convergence.

This thesis focuses on cross-situational word-referent learning with active learning strategies. We will aim at implementing these models on a humanoid robot in interactive scenarios by applying and improving related learning algorithms and visual processing techniques, which will be detailed in the later sections.

1.2 Existing models of word-referent learning

Regarding the current approaches to word-referent learning, [40] proposed to categorize the symbol grounding problems as *physical symbol grounding* (firstly defined in [196]) and *social symbol grounding* (originally proposed by [29]). While the *physical symbol grounding* means the grounding of symbols to real world objects by a physical agent (eg. robot) interacting in the real world, the *social symbol grounding* refers to the collective negotiation for the selection of shared symbols (words) and their grounded meanings in (potentially large) populations of agents.

Our work is mainly focussed on the domain of *physical symbol grounding*. Existing models in this category mainly differ on the following aspects:

1) the scenarios (*experimental settings*) they are working on. For example, is it a learning machine system or a real robot which takes part in the learning? Does the learning takes place in an interactive scenario or just database learning?

2) the *data types* which are used, concerning vision (local descriptors or global descriptors), speech (raw sound or symbolic words) and even other modalities;

3) the *learning algorithms* that are applied not only for the objective of word-referent learning but also for classification or identification.

Yu [201] presents a multimodal learning system that can ground spoken names of objects in their physical referents and categorize those objects simultaneously from vocal and vision input. For the audio part, a *natural language processing* module processes raw audio data using lexical and grammatical analysis on the utterance consisting of several spoken words (ie. keywords as nouns) so as to convert the continuous wave pattern into a series of recognized words by considering phonetic likelihoods and grammars. For *visual processing*, a head-mounted camera is used to get visual features (including color, shape and texture description) that are extracted as perceptual representations. These feature sets are labeled with temporally co-occurring object name candidates to form many-to-many word-meaning pairs. For learning, the problem of multimodal clustering and correspondence is finally solved by the proposed *Generative Correspondence Model*.

Mangin [123] proposes an approach based on Non-negative Matrix Factorization for learning complex human movements applied to data recorded in a database. The learning system associates motions with sound and word labels. The motion part encodes the skeleton position and velocity acquired from a single human dancer through a Kinect device and the raw sound information is taken from the Acorns Caregiver dataset [4].

Araki [5] proposes a multimodal (vision, sound and haptic properties of object) approach implemented on a real robot, focusing on learning object concepts by using Latent Dirichlet Allocation (LDA). The multimodal data are acquired autonomously by a robot equipped with a 3D visual sensor, two arms and a small hand-held observation table that serves as the platform for capturing multi-view visual images of objects and complemented by a small amount of linguistic information from human users.

Noda [136] uses deep neural network to achieve the association of cross-modal information, including image, sound and motion trajectory. The memory retrieval, behaviour recognition and causality modeling experiments are tested on a small humanoid robot NAO, developed by Aldebaran Robotics [1].

The Talking Heads [178] is another model of language acquisition among a population of agents, which consists of a visual perception system, a symbolic communication channel, and an associative memory. In this experiment, a pair of agents are chosen randomly as "speaker" and "hearer" to accomplish series of guessing games on an open-ended set of geometric figures

[1] http://www.aldebaran-robotics.com/Downloads/Download-document/192-Datasheet-NAO-Humanoid.html

Model	Scenario	Data	Algorithm
Yu [201]	Learning Machine & Interactive	Color, Shape, Texture, Raw Audio	Generative Correspondence Model
Mangin [123]	Learning Machine & Database	Motion, Images, Raw Audio	Non-negative Matrix Factorization
Araki [5]	Real Robot & Interactive	Image texture, Raw audio, Haptic	Online Multimodal LDA
Noda [136]	Real Robot & Interactive	Image, Sound, Motion	Deep Neural Network
Talking Heads [178]	Learning Machine & Interactive	Shape, color, Symbolic Words	Categorization trees
CELL model [150]	Learning Machine & Database	Shape, Raw Audio	Clustering and mutual information

Table 1.1: Overview of the word-referent learning approaches.

pasted on a white board so that a shared lexicon as well as the perceptually grounded categorization of objects are self-organized within this population without human intervention or prior specification. The learning is based on the gradual construction of categorization trees that associate features and words.

The CELL model of Roy [150] (Cross-channel Early Lexical Learning) is a cross-situational model of word-referent learning from multimodal sensory input. It has been implemented in the experiment of grounding shape names acquired through a word acquisition model based on directly processing raw data from spontaneous infant-directed speech which are paired with video images of single objects. The main structure of CELL is composed of speech processing, computer vision, and machine learning algorithms together with the settings of STM (short-term memory) and LTM (long-term memory). STM serves as a buffer where pairs of recurrent co-occurring utterance-shape events (also known as audio-visual prototypes or AV-prototypes) are filtered and LTM further applies a recurrence filtering by first clustering the AV-prototypes from STM and then consolidating them based on a mutual information criterion [43] as the final lexical units.

The main aspects of the selected existing models are summarized in Table 1.1.

1.3 Perception for word-referent learning

Since word-referent learning entails the association of both *linguistic words* and (mainly) *vision features*, the experimental settings and computational models have to deal with the perception of speech and vision before applying algorithms for the association.

Before detailing the speech and visual processing approaches used in the word-referent learning context, let's first introduce a very generic model used in several systems: the Bag of Words (BoW) modelling approach. This approach has been used in many domains, ranging from text processing to computer vision and acoustic processing. BoW theory assumes that an object (possibly a document, an image or an audio stream, etc) is represented as an orderless collection of words, vision features or acoustic events whose set is defined as a dictionary. Therefore, a described object can be encoded by the frequencies of its words (features or events) associated with the dictionary, while discarding most sequential information and geometric relationships. Generally speaking, the BoW algorithm consists of the following steps: raw feature extraction, feature quantization into dictionaries (also called codebooks) of words (quantized features or events), and representation of data as the word frequencies for each document. The BoW approach has the advantage of providing a fixed size and sparse data representation that is suited as input for many learning algorithms.

1.3.1 Speech perception

Speech perception has been treated with two approaches: either by processing directly the raw audio speech data or by first converting voice to text before analysis. Besides this distinction, another distinction can be made on the way models deal with noisy speech data from a complete natural sentence comprising non-referent articles, conjunctions, pronouns, interjections, and even spoken errors, apart from key words such as nouns and adjectives. Existing solutions either require that the speaker only utter keywords or make use of grammar or statistics to process the noisy input.

1.3.1.1 Raw audio wave based approaches

Some models directly take the raw audio signal, not assuming a preliminary segmentation and filtering of the words.

Mangin et al. [123] proposed a BoW method based on histograms of acoustic co-occurrence (HAC) presented by Van Hamme in [194] and in [23, 56]: from the spoken utterances, the Mel-Frequency Cepstral coefficients (MFCC) is computed together with its first and second order time derivative; then MFCC and its derivatives are split at multiple time scales into small chunks, which will be vector quantized into a dictionary. Using this dictionary, each utterance is represented by a sequence of discrete acoustic events each corresponding to an occurrence of a chunk from a cluster; and the final representation of the utterance will be an histogram over the occurrences and the successive co-occurrences of pairs of the events above.

CELL model [150] directly deals with raw acoustic input which is first converted into a spectral representation using the Relative Spectral-Perceptual Linear Prediction (RASTA-PLP) algorithm [83] for the purpose of filtering out the nonspeech components of acoustic signals. This is done by using bandwidth filtering, because the variation frequency of non-speech signals always appear either faster or slower than that of speech signals. The filtered

signals are processed with an exponential transformation and scaling so as to be presented as a set of 12 RASTA-PLP coefficients estimated at a rate of 100 Hz followed by an analysis with a recurrent neural network (RNN) for computing a probability distribution over 40 phonemes, ranging from “aa”, “ae” to “z”. The RNN is trained off-line using the TIMIT database of phonetically transcribed American English speech recordings [66]. Finally as a result, input speech is represented as an array of probabilities for the 40 phonemes.

1.3.1.2 Text based approaches

Other approaches take already recognized words as inputs, thereby directly using symbolic information. There are many specialized softwares and online services to process raw audio wave data and convert them into recognized sentences. Example of these cloud processing services include Google speech-api[2], IBM watson [3] or Microsoft Bing Speech API [4].

In [201], “Dragon Naturally Speaking” software is used for speech recognition by considering phonetic likelihoods and grammars. Besides, Link Grammar Parser [171], which is capable of labeling words' syntactic categories, is used to extract from transcripts the nouns, among which the nouns belonging to physical objects or entities are selected by applying WordNet [129]. Using these pre-processing, the model is relying on a set of noiseless keywords.

In [5], a robot collects sentences uttered by a human participant as the multimodal perception of the observed object goes on. Each description of object appears as continous speech signals which then are converted into sequences of words by utilizing morphological analysis which makes it possible to extract real informative linguistic components such as nouns and adjectives that are then treated as a bag of words. Finally, a histogram whose length equals the number of selected informative words is generated as the word information of objects.

In the Talking Heads experiment [178], speech recognition is not used since the goal of the experiment is to show the construction of a shared vocabulary between agents, but not to align the meaning of human words to agent perception. Therefore all agents have Internet connections where they directly exchange symbolic information, without verbal communication, thus emulating a perfect word recognition system.

Note that the Bag of Words provides a very simple representation of a data, but overlooks the words ordering. Other language representations could be used, such as the "n-grams" which is very popular in the Natural Language processing domain since it can easily be learned through the frequency calculation of occurrences in a database. It has proved very efficient for speech recognition, since it catches the order of the words. However, even 3-grams and 4-grams require very large model since, for N words, N^3 and respectively N^4 parameters need to be learned, even with backoff methods which make it faster. Such models are therefore rarely used in the language grounding community.

[2] `https://github.com/gillesdemey/google-speech-v2`
[3] `https://github.com/watson-developer-cloud/speech-to-text-nodejs`
[4] `https://github.com/Microsoft/ProjectOxford-ClientSDK/tree/master/Speech`

1.3.2 Visual perception

Vision is the most frequently used modality for word-referent learning, but the raw stream of images from a camera carries a lot of redundant information which not only makes the data processing difficult but also submerges the useful information. Therefore, dimension reduction, feature extraction or object segmentation are common processing steps applied before learning word-referent associations.

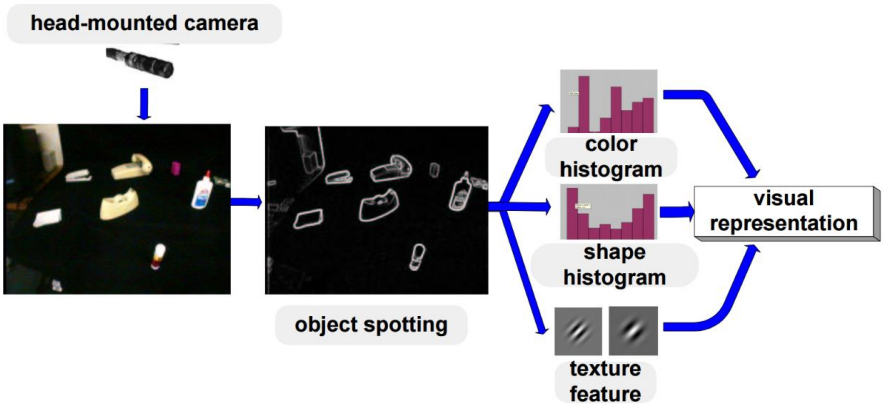


Figure 1.2: The diagram of visual processing in Yu [201].

The visual perception in the model of Yu [201] takes place on a simple uniform background, making object segmentation relatively easy. A head-mounted camera captures sequences of images continuously and a position sensor tracks the movement of the head. When the head is detected stable, a snapshot of the scene is taken together with the temporally co-occurring spoken utterances. Objects are extracted using color segmentation. Each object extracted from the scene is represented by histograms of features including color, shape and texture properties. By defining a discrete color space on the basis of color axes (e.g. red, green, blue) and counting the number of times each color unit from the color space occurs in the image array [181], a 64-dimensional color histogram is obtained. The shape feature histogram which is 48-dimensional in length is calculated on top of the multidimensional receptive field histograms capturing shape-related local characteristics (eg. neighborhood operators and local appearance hashing) [159]. As for the texture presentation of an object, assuming that local texture regions are spatially homogeneous, Gabor filters [62] with three scales and five orientations are applied to the segmented image and the mean and the standard deviation of the magnitude of the transform coefficients are used to form a 48-dimensional texture feature vector. In total, histograms of color, shape and texture form a 160-dimensional feature vectors, which will finally be transformed in a lower dimensional subspace of 15 dimensions in total by using Principle Component Analysis (PCA). The overall diagram is depicted in Figure 1.2.

Although Mangin's [125] grounding objective is mainly towards choreographies (dancer's movement), vision is also used to encode object appearance (see details in [120]): pictures are acquired through a Kinect camera, then objects are segmented from the background by using depth and motion detection and characterized by local features (SURF keypoints [12]) and by their colors in the HSV color space. These descriptors are quantized in dictionaries and objects are represented following the Bag of Words approach by an histogram of these

feature occurences.

The information perception in Araki [5] also depends on vector quantization (k-means algorithm) and feature extraction. At first, a small hand-held observation table is prepared for the robot when it takes a target object on the observation table to get non-occluded various views with necessary manipulations. Then in each image frame of the observed views, object is detected and segmented simply by applying the planar segmentation algorithms as described in [152, 153] which is followed by the feature description through the computation of 36-dimensional PCA-SIFT descriptors [101]. In fact, every image is characterized by from 300 to 400 feature vectors, and considering that there are ten image frames for one object, thus for every object there would be about 3000-4000 features, each of which is vector-quantized using a codebook (generated by k-means algorithm in advance) with 500 clusters. Finally, a histogram for the bag of features representation is formed.

In Noda [136], image sequences of the scene (an observation of the manipulation of objects by the robot) are used for the training of deep neural networks. This network is trained to reconstruct the images using a limited size in its inner layer (a model called an auto-encoder) and the corresponding feature vectors is taken from the inner layer. In this work, the full images are used without object segmentation as the objects represent a very large part of the field of view given the small size of the robot's hand and fingers.

The Talking Heads experiment [178, 179] works with simple colored geometric objects and deals with the segmentation of objects by color thresholding thanks to a simple background (ie. a white board). For each object, a set of features are computed: *area*, *hpos* (horizontal position), *vpos* (vertical position), *height*, *width*, *bb-area* (the area of the bounding box), *gray* (the average gray-scale value of the pixels in an object), *r, g, b* (the average redness, greenness, and blueness values in a segment), *edge-count* (the number of edges in a segment), *angle-count* (the number of angles in a segment) and *ratio* (the ratio between the area of the segment and the area of its bounding box).

In the CELL model [150], only object shapes are observed and used for the lexical acquisition. A 3-D observation of an object is carried out on top of a view-based approach in which a group of two-dimensional images of an object is captured from multiple viewpoints. Simplification which facilitates the vision perception includes an uniform background and the fact that referent objects are presented single and unoccluded to the system. Besides, all the visual data are generated off-line. As for the segmentation, a Gaussian modeling of the illumination-normalized background color is estimated so as to classify background as well as foreground pixels, and the large connected regions near the center of an image indicates the presence of an object. The shape of an object is represented by an histogram describing the shape of its boundary. Finally, the 3-D representation of an object in terms of shape is based on a set (noted as *view-set*) of two-dimensional histograms representing different views. In order to compare two objects, the distance between these object is computed as the sum of the four best matches (using χ^2 divergence) between histograms of two view-sets.

Object representation and recognition is actually a very active research area in computer vision. Besides the approaches presented above and used in word-referent learning models,

Figure 1.3: The Talking Heads experiment (from [178]).

more powerful representations exist and could be used in these models. We will however not try to give an overview of the existing feature presentation approaches, but in particular present the Histogram of Oriented Gradient (HOG) representation that we use in some of our experiments.

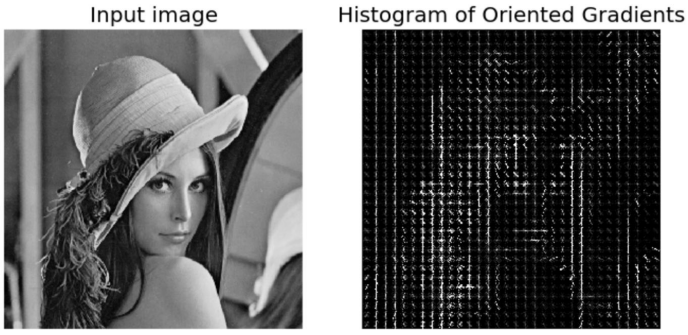


Figure 1.4: HOG image of *Lena*.

HOG, primarily proposed by Navneet Dalal and Bill Triggs [45], is based on the counting of occurrences of gradient orientations in localized portions of an image, thus less influenced by geometric and photometric transformations as well as the scaling of size. The image is divided into cells which are small connected regions of pixels, and within each cell a histogram of gradient directions concerning every pixel is compiled. Besides, the contrast-normalized strategy is applied across a block, normally containing several cells, to enhance the invariance to changes in illumination and shadowing by normalizing all cells within the block via several effective normalization methods in [45, 119]. HOG originally focuses on pedestrian detection in static images and then expands its application to the static detection of other objects and even the detection in videos. An example of HOG could be found on the processing of *Lena*, shown in Figure 1.4.

1.3.3 Other modalities

Apart from visual and speech perceptions, there are other modalities that contribute to the feature presentation of the observed objects, mainly including motion, sound and haptics.

1.3.3.1 Motion

Motion perception is highly related to the *grounding of verb*, referring to a physical action or a temporal flow of events. According to its complexity from low to high, the objectives of motion perception could be roughly categorized as

- *Motion profile based*, where the system characterizes motions only in terms of time-series data concerning relative-and-absolute positions, velocities, and accelerations of the participant objects and applies stochastic reasoning (often in the form of hidden Markov models) to classify the time-series data into event types;
- *Event logic based*, as a result of the analysis of the force-dynamic relations between the participant objects. Instead of motion trajectory oriented, the time-series data will take the form of the truth values of force-dynamic relations between the participant objects as a function of time. Finally, by applying logical reasoning in the form of event logic, the classification of events is executed.

Evidently, the *event logic based* motion perception, which is firstly proposed by Siskind [169] by adopting Talmy's theory of force dynamics [182], has advantages over the *Motion profile based* method on several aspects, such as it is less sensitive to a wider variance in motion profile (eg. the angle and velocity of the participant objects), not influenced by the presence of unrelated objects in the field of view and capable of handling complex events (eg. sequential, overlapping or even hierarchical events). However, on considering the word-referent learning scenarios of related works, we find that they most adopt the framework of the first motion perception category, mainly due to the fact that most actions in experiments remain simply at the motion level other than the event level.

For example, in Yu's model [200], a simplified model is build, where body movements (regarded as motion profiles) are mapped to action verbs, without inferences about causality function and force dynamics described in [169]. Main procedures consist of *action segmentation* and *motion featuring*. Based on the fact that actions mostly occur during fixations of either eyes or head, user-centric attention switches, indicated by gaze and head cues, are calculated and used for the segmentation of a continuous action stream into action units that are then categorized by Hidden Markov Models (HMMs). From the raw position and the rotation data of each action unit, the feature vectors consisting of the hand speed on the table plane, the speed in the vertical-axis, and the speed of rotation in the three dimensions are extracted for recognizing the types of motion rather than the accurate trajectory of the hand movement.

In the work of Mangin [125], motion is used as input in order to ground choreographies (dancer's movements). The data is acquired as 3D position of a set of skeleton points from a single human dancer through a Kinect device and the OpenNI software that enables the direct capture of the subject skeleton. By applying a simple geometrical model of human limbs, 12 angle values are calculated, representing the dancer's position at a specific time. Further more, angular velocities are computed out of 12 angular values for better descriptions of motion features. These data are clustered and represented as histograms of position/velocities for all angles.

In Noda [136], motion is represented by joint angle sequences of performing bell-ringing behaviors by robot's arms. For each joint angle data input, 10 degrees of freedom of the arms (from the shoulders to the wrists) are used. The joint angles of both arms (as well as image frames) are recorded at approximately 66 Hz, and a contiguous segment of 30 steps from the original time series is used as a single feature vector describing the recent motion.

1.3.3.2 Sound

The perception of sound is specified for those objects (such as a bell) having intrinsic auditory features under certain conditions, for example the manipulations performed by humans or robots.

In Araki's model [5], the sound information is used to characterize an object. When the hand of the robot holds and shakes the object, the raw audio stream is processed to a 13-dimensional MFCCs feature vectors before the vector quantization based on k-means algorithm constructs a histogram.

In the model of Noda [136], the sound of objects is also used. It is generated mainly because of the touch on the tip of a ring bell by the hand of robot. The pulse-code modulation (PCM) sound data is recorded by a single channel microphone mounted on the forehead of the robot, preprocessed by discrete Fourier transform (DFT[5]) that is then converted into a feature vector by a deep auto-encoder .

1.3.3.3 Haptics

Haptic information contributes to characterize physical properties of an object such as its materials. Araki [5] obtains the haptic information from a three-finger robotic hand with a tactile array sensor: when the robot' hand presses an object, the time series of data from 162 pressure sensors are collected and then transformed to 162 feature vectors, following the bag-of-features approach with 15 codewords.

[5] https://en.wikipedia.org/wiki/Discrete_Fourier_transform

1.3.4 Summary of data representation

The approaches for encoding perceptual information have been described in the previous section, and in terms of presenting data from a pragmatic point of view, the aforementioned Bag of Words and histogram approaches are the most popular due to their convenient ways of recording very different kinds of features while occupying limited data space.

In fact, although humans can process raw sensory information naturally and easily extract their intrinsic features for further analysis and reasoning, it is still a computationally expensive process for a learning machine or a real robot. Hence for any computational learning model, perceived data should be processed in a way that only critical features remains, keeping them prominent thus mathematically determinant, while limiting representation space, which is expected to accelerate their processing.

Therefore the majority of related works, by using feature descriptors, characterizes modal information on the basis of frequency presentation over a codebook of features, resulting in histogram presentation whose dimensionality has been massively reduced. We will therefore resort to similar approaches in our work, by designing data processing that results in data representation in the form of histograms.

1.4 Learning algorithms

In the domain of word-referent learning, supervised learning approaches could be used by giving samples of situations that correspond to each word to be learned. However, in realistic interactive scenario such as when a child learns the meaning of words from his parents, such detailed supervision is not available. On the contrary, as will be explained in more details in Section 1.5, several sources of ambiguities in the referent objects and in the utterances have to be dealt with. Considering the absence of ground truth in the above contexts, classical supervised classification methods appear invalid and unsupervised learning algorithms should be resorted to for the problems of word-referent associations.

Among various unsupervised-learning solutions, an effective approach for learning word-referent associations, is the topic model approach. The goal of topic modeling is the automatic discovery of the topics from a collection of documents. Topic models [19, 73, 75, 74, 88, 89] are based on the assumption that documents are mixtures of topics, where a topic is a probability distribution over words. As a generative model for documents, a specific topic model acts as a simple probabilistic procedure by which documents can be generated. In order to write a new document, one has to choose a set of topics from a topic distribution, followed by the choice of words from these topics, obeying a topic-word distribution, and this generative iteration continues until all words are chosen. As shown in Figure 1.5, if topics are well-defined by following their respective distributions over words (see the far left), then a document can be featured by a certain distribution over these topics (the proportion among topics as shown in the histogram on the far right). Thus, topics are not directly observable, but give a structured

presentation of seemingly unstructured data by bridging the connection between words and documents.

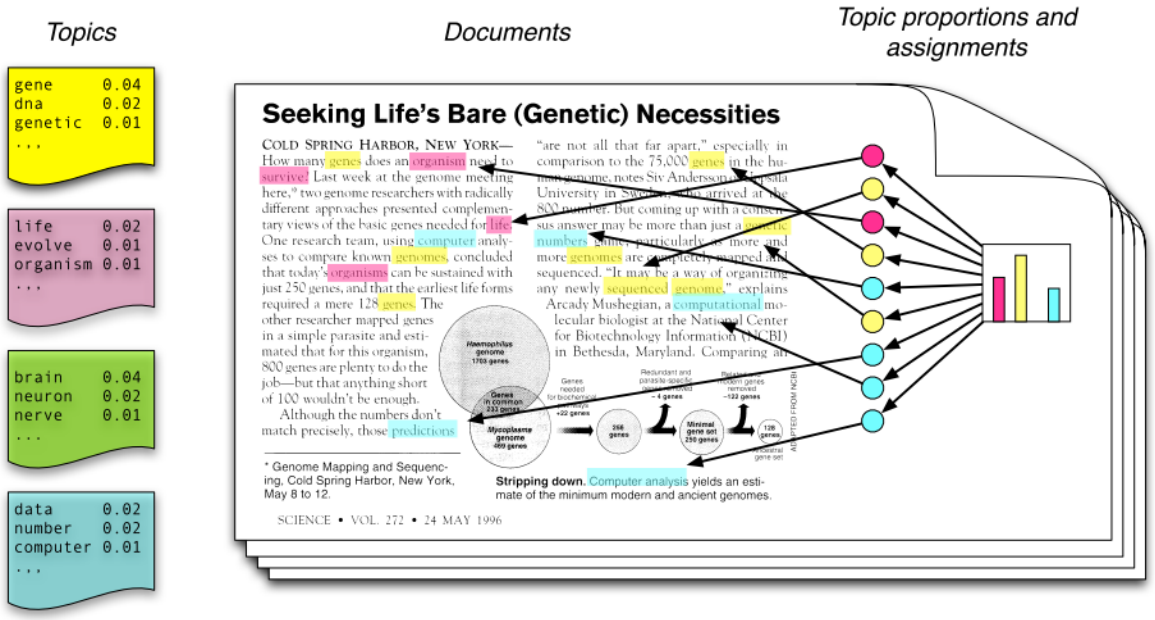


Figure 1.5: An illustration of a topic model (from [18]).

A particular feature of the topic models is that they assume individual words' exchangeability. Under this assumption, two sequences of words have the same density of probability if they contain the same words but in different orders. This is the standard assumption of the Bag of Words approach we proposed to use for data representation, thus making topic models applicable to our problem.

Topic models have several interesting properties. First of all, they are unsupervised approaches that let latent topics emerge from the analysis of the original data, and since no prior annotation or labeling is needed, they enable the automatic organization and summarization of many kinds of data such as genetic data, images, social networks, etc. Secondly, regarding the interpretability, the extracted topics can always find explanations either from a probabilistic point of view or in a specified vector feature space, and these explanations make it possible to devise good criterion for task like data comparison. Finally, latent topics can deal with more than literal meanings, thus to some extent simulating humans' comprehension from seemingly non-informative or misleading raw data while removing the noise from the original raw data, for instance, tackling a situation in which a certain meaning could be indicated by multiple words or a certain word corresponds to several real world meanings.

Nevertheless, we could not neglect its drawbacks that is mainly due to its word exchangeability assumption, rarely true in practice. By this assumption, the topic model also discards the temporal and/or spatial information when generating a topic and hence ignores the semantics of the context.

When boiling down to the applications of the word-meaning association learning, the topic model will generate topics that could be considered a *concept* having manifestation in

the different modalities. For example, the topic for "red" would relate a word label "red" in the linguistic modality and correspond to a spectrum histogram representing red in the visual modality. This kind of concept is called *multimodal concepts* in [125], where a learned concept is represented by an histogram, consisting of three parts describing vision, motion, and sound.

Topic models may be implemented by algorithms following different mathematical principles, the main algorithms can be categorized as *probabilistic* models, such as Probabilistic Latent Semantic Analysis (pLSA) or Latent Dirichlet Association (LDA) and *matrix decomposition* models, such as Principal Component Analysis (PCA) or Non-negative Matrix Factorization (NMF). In the remainder of this section, we will first give a brief introduction to these algorithms and present their applications in related works. Before going to these topic models, we will first review other algorithms that have been applied to the word-referent learning problem.

1.4.1 Non Topic model approaches

Besides the wide use of topic models in the applications of word-referent learning/symbol grounding, there have been also other non topic model approaches, based on, for example, the statistical analysis of the directly observable variables or the low dimensional feature learning from observations. Hence in this part, we briefly introduce a co-occurrence based approach and a neural network based method, each one of them being representative to the two mentioned categories respectively.

1.4.1.1 Co-occurrence based approach

Several algorithms working directly on co-occurrences between word and referent have been proposed. In the CELL model [150], as an example of this category, a measure of mutual information in terms of co-occurence frequency between clusters representing *words* and cluster representing object *shape* is used to find the most probable word-shape association.

As previously described, words are obtained through the conversion from the raw acoustic wave data and shapes are derived from camera images. And in order to represent a spoken utterance (of words) paired with a visual object shape, an *audio-visual event* or *AV-event* is defined as {speech segment, object}. Figure 1.6 shows the diagram of the CELL model:

1) The short term memory (STM) works as a first-in-first-out buffer containing input AV-events of a constant number (eg. limited to five), followed by the exhaustive searches of the short-term recurrence filter for AV-events that share similar visual contexts and also have recurrent/repeating legal speech segments (ie. containing at least one vowel). Any AV-event that passes this filtering is called a *AV-prototype*, to be stored in the long term memory (LTM);

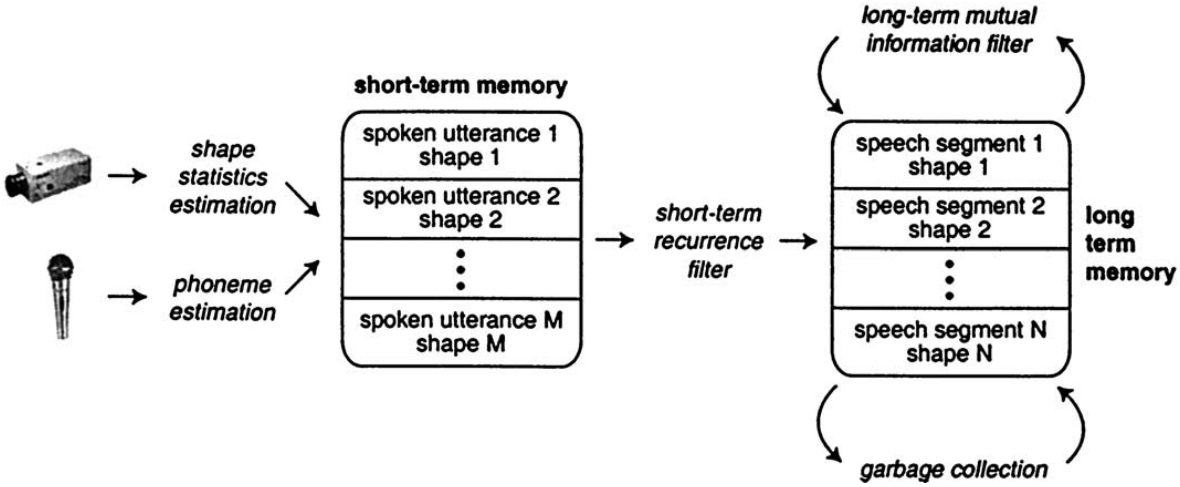


Figure 1.6: Diagram of the CELL model [150].

2) In LTM, however, the AV-prototypes contain noisy words (ie. words not referring to objects, like "the", "a/an", etc.). Therefore, *lexical items* are created by consolidating AV-prototypes based on a mutual information criterion. At first, the acoustic and visual presentation of a new prototype is stored in the acoustic and visual feature space with a radius taking into consideration pronunciation variance and visual noise. Then a mutual information function [43] is applied for both the new prototype and all n AV-prototypes in LTM as

$$I(A;V) = \sum_i \sum_j P(A=i, V=j) \log \left[\frac{P(A=i, V=j)}{P(A=i)P(V=j)} \right] \tag{1.1}$$

where $P(A=i) = \frac{|A=i|}{n}$ is calculated as the proportion of the number of AV-prototypes in LTM that shares approximately the same acoustic segment of the new prototype (ie. by falling within the circle of the acoustic space illustrated on the left of Figure 1.7) against the total number of n, then vice versa $P(V=j) = \frac{|V=j|}{n}$ is for the case of visual prototype and $P(A=i, V=j) = \frac{|A=i, V=j|}{n}$ is jointly for both acoustic and visual prototypes. Note that two radii – r_A, r_V are determined by a search procedure that maximizes $I(A;V)$;

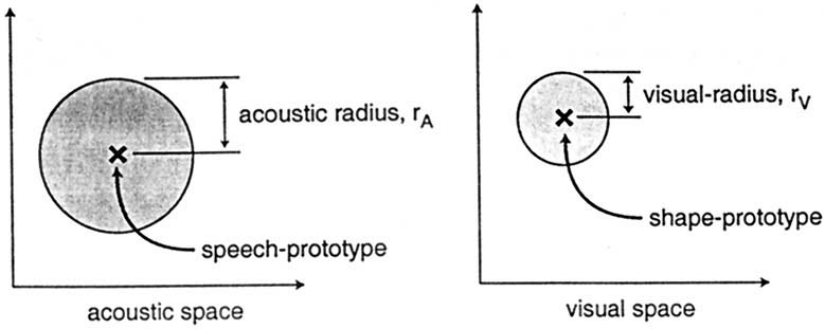


Figure 1.7: Illustration of the acoustic-prototype and shape-prototype marked as centers, with acoustic radius and visual radius as allowable deviations (from [150]).

3) Finally, a lexical item is formed, by following a "winner-takes-all" strategy, when $I(A;V)$

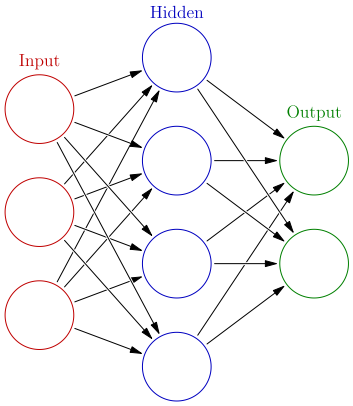


Figure 1.8: An illustration of a neural network with one hidden layer.

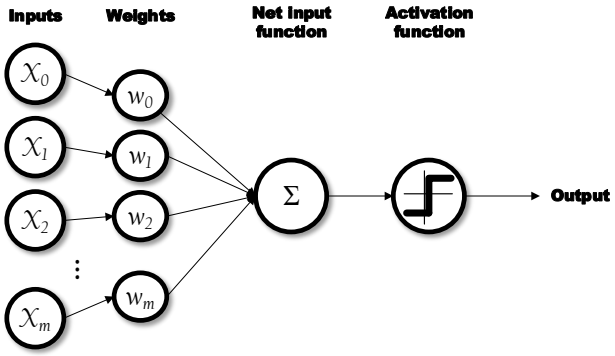


Figure 1.9: Diagram of data processing at a neuron node.

exceeds an empirically set threshold, meanwhile all other AV-prototypes in LTM that match this lexical item acoustically and visually are removed as the last procedure of garbage collection filtering.

As a summary, the CELL model, 1). is based on the fact that elements truly concerning either a referent or a word symbol co-occur frequently in contrast to the occurring behaviors of noisy parts; 2). the co-occurring phenomenon is proved effective in a statistical manner to validate real word-referent pairs across different situations. Nevertheless, problems exists when applying the co-occurrence based approach that it is difficult to deal with *synonym* and *polysemy* and the model has to endure heavy computational burden as the learning samples increase.

Other approaches using similar strategies will not be detailled, but *Hypothesis testing* [128, 190] is based on the idea that a learner selects the most plausible interpretation (i.e. one referent) of a new word, which would be confirmed (hence remained) or falsified in later exposure to this word. In the second case, the old referential candidate is replaced by a new hypothesis. *Associative learning* [93] stores word-object associations between all co-occurring stimuli (other than only the "one word - one referent" mapping applied in hypothesis testing) and evolves this co-occurring matrix with preferential bias towards some specific word-object associations while gradually forgetting others.

1.4.1.2 Deep neural networks

The idea of deep neural networks is derived from Artificial Neural Networks (ANN) and the idea of using a large number of layers to augment the data representation capacity. Taking inspiration from how biological neural networks (in particular of the brain) work, a classical (feed forward) neural network, as shown in Figure 1.8, consists of several layers: input, hidden layer(s) and output. Each layer is composed of neurons (drawn as nodes) that are linked together and perform elementary computations.

The operation on data (see Figure 1.9) takes place between linked neurons, by first computing the weighted sum of input neural signals and then going through an activation function. The resulting value is taken as the input of the next layer. A training procedure based on gradient descent makes it possible to adapt the weights in order to learn a function from examples.

The traditional approach to learn the weights of deep neural networks is based on back propagation and gradient descent [84, 85], where we first set a cost function which normally is the calculation of errors between the data of the output layer and the predefined labelling data and then an iterative procedure minimizes the cost function according to the layer-wise gradient in terms of weights.

Originally, ANNs had been applied to pattern recognition for linear cases, before being extended to non linear cases. To find a better presentation of complex data structures, [14] argues that the observed data are generated by the interactions of many different factors on different levels (ie. distributed), and higher the level is, further the abstraction of concept it acquires. The definition of Deep Neural Networks (DNNs) therefore emphasizes that it should have multiple layers of hidden units between inputs and outputs. Each hidden layer of neural network contains the feature representation of the data from the previous layer, existing in a form that is more abstract and compressed.

Another variation of ANN, apart from DNNs, is the *autoencoder* (also known as *autoassociator* or *Diabolo network*) [13], a particular type of neural network for learning a lower-dimensional representation (ie. encoding) of raw data for the purpose of extracting their innate features of correlation as well as saving memory and also used for learning generative models of data [105]. The most prominent feature of the autoencoder is that it tries to output the data (by means of reconstruction) approximating the input as much as possible. And in the central hidden layer, the number of nodes is far less than that of the visible (input/output) ones, therefore we could define that the first half (from the input layer to the central hidden layer) of the neural networks is the procedure of *encoding* by compressing the dimensionality of inputs and the second half (from the central hidden layer to the output layer) represents the *decoding* trying to reconstruct the data to its original dimensionality. The task of training is to minimize an error of reconstruction and find the most efficient compact representation (encoding) for input data. Besides, another application of the *autoencoder* is to initialize the inter-layer weight distribution from "shallow" to "deep" (ie. starting layer-wise from the input layer onwards) as an effective pre-process to improve the results of the subsequent back propagation, especially when training deep neural networks.

As an example of applications of these approaches to word-referent learning, Noda [136] extended the time-delayed neural networks (TDNN) [197] to a deep neural network of time-delayed autoencoder, with the goal of relating sequences of sound, vision and robot motion to recognize actions. In that framework, there are for every single modality an individual autoencoder whose central hidden layer is used as input of a main auto-encoder that encode sequences of perceptions, as shown in Figure 1.10. As a result, the main auto-encoder manages to *encode* behaviors (eg. the bell-striking motions performed by a robot) in a way similar to the *meaning* in the semiotic square in Figure 1.1, thus giving the basis to associate words to

actions.

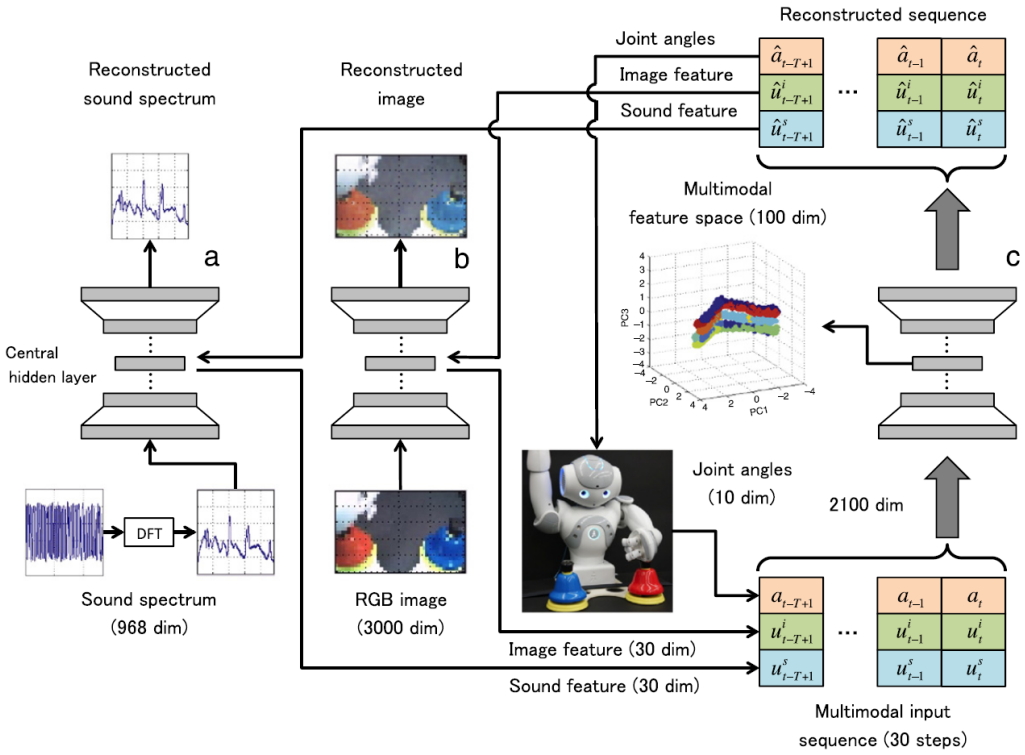


Figure 1.10: The structure diagram of DNNs in Noda [136].

A special note should be put here that although ANN can be explained from the point of view of restricted Boltzmann machine (RBM)[6], therefore theoretically very close to the topic model, still in practice they are customarily regarded distinct and implemented for different purposes.

1.4.2 Matrix decomposition topic models

Several matrix decomposition methods such as Singular Value Decomposition (SVD), Principal Components Analysis (PCA) and Non Negative Matrix Factorization (NMF) can be used to decompose data into sum of elementary components. The main objective of these methods is to reduce the dimensionality of data so as to focus on the important underlying elements and to reduce the presence of noise. Due to their different underlying principles, the results of the decomposition by applying these algorithms will be different. However, the decomposed matrix can be regarded as a set of component features (i.e. topics) that are much less in number compared to the original sample size yet available to reconstruct samples, either learned or totally new. Hence in this part, on one hand, special attentions will be paid on the properties of the decomposed results; on the other hand, since this thesis has relied in a large part on NMF, a particular attention will be devoted to NMF's relations with other widely used algorithms of this category.

[6]https://en.wikipedia.org/wiki/Restricted_Boltzmann_machine

One of the important properties of a decomposition algorithm is the interpretability of its result by comparison with the original data. In particular, it is interesting to see that if the results are non-negative because in the fields like computer vision, document clustering, speech recognition, bioinformatics, etc., most data appear non-negative. For example, Lee [113] illustrates the fact that the decomposition of images of faces into non-negative components (using NMF) is naturally interpretable as a sum of local elements, whereas its decomposition using PCA has not similar meanings. In Figure 1.11, positive values are illustrated with black or grey while red marks the negative. By using NMF, decomposition result is a dictionary of facial parts of mouths, noses and so on and so forth. On the contrary, the PCA basis contains some distorted versions of whole faces [192].

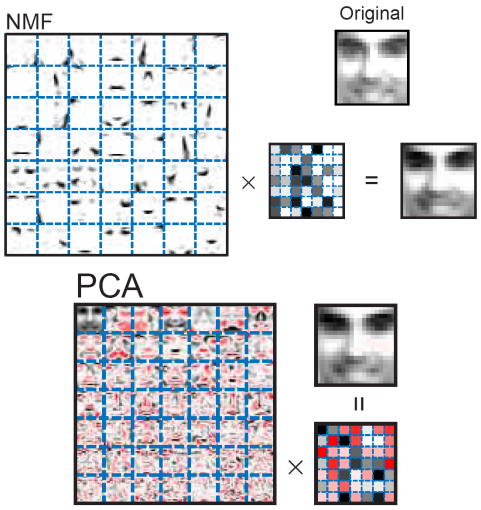


Figure 1.11: Comparison of NMF and PCA in an example of face decomposition. Example taken from [113].

1.4.2.1 Latent Semantic Analysis using Singular Value Decomposition

Singular value decomposition (SVD) [99] decomposes a matrix to two orthogonal matrices and a singular value diagonal matrix as:

$$M = U\Sigma Q^T$$

Commonly, the singular values in Σ are listed in a descending order to guarantee the uniqueness of the decomposition. The low-rank matrix approximation [180] is proposed to approximate a matrix M(of rank n) with a truncated matrix $\tilde{M}$ (of rank r, with $r \ll n$) as

$$\tilde{M} \approx U\tilde{\Sigma}Q^T \tag{1.2}$$

where $\tilde{\Sigma}$ contains only the r largest singular values while all others are zeros. With this approach, the Frobenius norm of the difference between $\tilde{M}$ and M is minimized. One advantage of this method is that noise factors (indicated by the discarded singular values) are filtered out by keeping only the most essential parts.

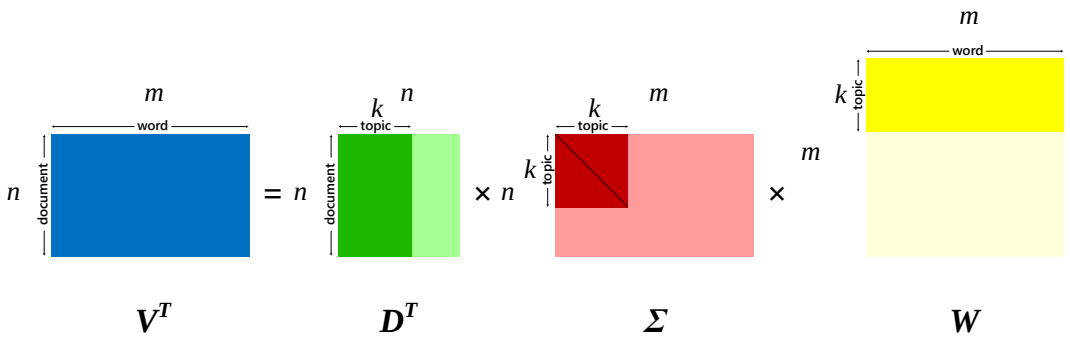


Figure 1.12: Illustration of SVD/LSA.

Considering that all the information to be processed constructs a corpus, which is made up of documents, we define a matrix $V_{m\times n}$ where each row i $(i = 1, 2, ..., m)$ denotes a (feature) word and each column j $(j = 1, 2, ..., n)$ a single document. The element m_{ij} is denoted as the number of occurrences of a particular word i in document j. Latent semantic analysis (LSA) [108] decomposes V^T via singular value decomposition (SVD) [65] as

$$V^T = D^T \Sigma W \tag{1.3}$$

where D is an orthogonal occurrence matrix of document-topics, each column of which indicates the topics of a document;Σ is a diagonal matrix of singular values which indicate the importance of each topic in the corpus; W is the topic-word orthogonal matrix whose row and column represent topic and word information respectively (Figure 1.12).

In order to let relevant topics appear, dimensionality reduction is performed by keeping only the largest r among all k singular values in Σ while replacing the others with 0. Thus, we get an approximate singular matrix Σ_r and also an approximate word-occurrence matrix (ie. the rank r approximation to V with the smallest error of Frobenius norm) as

$$V_r^T \approx D_r^T \Sigma_r W_r \tag{1.4}$$

Since D_r^T indicates the information in the documents at the topic level, we are interested in the following relation:

$$D_r^T \approx V_r^T W_r^T \tilde{\Sigma}^{-1} \tag{1.5}$$

This means that when a new document vector $v_i (i \in \{1, ..., n\})$ appears, we could project it into a semantic space by

$$d_i \approx v_i W_r^T \tilde{\Sigma}^{-1} \tag{1.6}$$

The main merit of LSA is the use of a simple technique (SVD) for uncovering hidden or "latent" data structures.

However, the main limitation is that LSA lacks a sound interpretation in both physical and probabilistic way. Moreover, the choice of the number of singular values (ie. the rank r) seems more empirical than standardized.

1.4.2.2 Principal Component Analysis

Another algorithm for matrix decomposition is Principal Component Analysis (PCA) [142] which draws from data the most representative components which possess as much information as possible for presenting the data, while avoiding the overlap of information between components, by choosing them orthogonal.

Assume that the matrix of $\hat{V}_{m\times n}$ represents n samples of observable data with m dimension of features. We first create the modified matrix $V_{m\times n}$ such that $E(V_j) = 0$ by mean centering every column of $\hat{V}_{m\times n}$ ($v_{i,j} = \hat{v}_{i,j} - \frac{1}{m}\sum_{q=1}^{m}\hat{v}_{q,j}$, where $i \in \{1, 2, ..., m\}$, $j \in \{1, 2, ..., n\}$).

PCA will find H, a $k \times n$ transfer matrix, such that

$$W = V_{m\times n}H^T_{n\times k} \tag{1.7}$$

where W is the vector space after projection. In order to respect the two above constraints (maximizing information and having orthogonal components), it can be show that H should contain the eigenvectors of $Cor(V)$ (denoted as the covariance of V) whose diagonal elements are the variances of each row vectors while the remaining positions are covariances. If H is composed of eigenvectors placed in the decreasing order of related eigenvalues, this decomposition is unique.

Similar to the low rank approximation in SVD, we can choose the most important r ($r \leq n$) components for the compressed presentation

$$\tilde{W}_{m\times r} \approx V_{m\times n}\tilde{H}^T_{n\times r} \tag{1.8}$$

So for each new piece of data, after being projected into the eigenvector space by premultiplication of the transfer matrix $\tilde{H}^T$, it could be compared with other data for the applications like clustering and classification, which lay the basis of the word-referent learning.

We may notice that PCA, as well as SVD, makes use of orthogonality in the vector space projection, and both matrix operations of PCA and SVD could be proved as mathematically equivalent, which even makes SVD become one of the standard ways to preform PCA (the other one is based on the calculation of the eigenvalues and eigenvectors of the covariance matrix), despite the fact that SVD is more referred in the field of natural language processing (NLP) while PCA is more involved in applications of computer vision.

PCA can be applied in many real world applications such as data representation, pattern recognition (eg. eigenfaces [192, 151]) or image compression (eg. Hotelling or Karhunen-Loeve transform). It is often used as pre-processing (dimensionality reduction, orthogonal feature projection, etc.) for more complex tasks like symbol referent learning.

The advantages of PCA are: 1) no parameter settings are required; 2) the reduction of dimensionality is effective for maintaining the most important data, while compressing data and removing redundancy. Nevertheless, some major drawbacks can not be neglected like 1)

it is ineffective for the samples with non-linear properties (yet Kernel PCA [162] could solve this problem); 2) there is no standardized rule for the number of principal components; 3) if negative values of decomposed data appear, there is no physical interpretation for them, which remains a serious problem for many other matrix decomposition algorithms as well (see Figure 1.11).

1.4.2.3 Non-negative Matrix Factorization

As a breakthrough to deal with the problem of interpretability of decomposed results, Non-negative Matrix Factorization (NMF) [112, 113] is a decomposition method that requires that all elements in matrices, both the original and decomposed, should be non-negative.

NMF is also known as non-negative matrix approximation seeks non-negative matrices W and H so that:

$$V_{m\times n} \approx W_{m\times k} H_{k\times n} \tag{1.9}$$

where m and n are the number of features and the count of samples respectively, and k is the number of components. This same equation can also be written of each matrix element $V_{i,j}$:

$$V_{i,j} \approx (WH)_{i,j} = \sum_{r=1}^{k} W_{i,r} H_{r,j} \tag{1.10}$$

A physical explanation of the above decomposition is that the column vectors of V are n observable samples (of m dimension), each of which is the linear combination of component vectors (represented by the k column vectors in W) weighted by the k coefficients of the related column vector in V. Obviously, computing the NMF basis of W is one of the most important issues.

Comparing NMF with PCA, by post-multiplying both sides of Equation 1.7 with the orthogonal H, we come to

$$V = WH \tag{1.11}$$

and choose the k largest principal components for approximation, we get

$$V_{m\times n} \approx W_{m\times k} H_{k\times n} \tag{1.12}$$

From this point of view NMF shares the same format of matrix decomposition with PCA (thus with SVD as well), yet the difference lies in non-negativity of NMF, which permits better interpretability as illustrated in Figure 1.11.

Another comparison of NMF can be drawn with neural networks [78]. As illustrated in Figure 1.13, the hidden layer of k nodes in a simple auto-encoder network can be considered as the NMF basis, which is decomposed from the training matrix $V_{m\times p}^{training}$. When reconstructing the data using the k components, the link between the hidden layer and output layer (denoted by $H_{k\times q}^{compose}$) will perform the linear combination of the k nodes to each of the output data.

From this model, it is easy to observe the data compression capability when $k \ll p$. And also from this structure, hierarchical extension of NMF can be defined [41, 11, 102, 175].

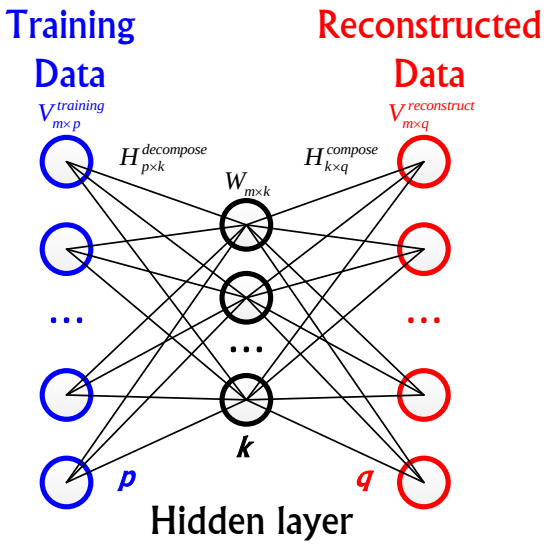


Figure 1.13: Explanation of NMF from the model of neural networks. (Derived from Figure 3 in [113]).

The applications of NMF can be found in data analysis [113, 77], signal and image processing [38, 154], language modeling, text analysis [16, 50], music transcription [37], and bioinformatics [26]. In [123], the author made full use of the decomposition property of NMF to decompose the choreography and audio sentence into motion and sound primitives. This multimodal model is able not only to perform the decomposition of testing data, but also to perform the recovery of one modality of information when the other modal information is missing.

NMF can be realized using several algorithms. Using notations from Equation 1.9 and concerning a non-negative approximation with $V_{m\times n} \approx W_{m\times k}H_{k\times n}$ such that $W_{m\times k} \geq 0$, $H_{k\times n} \geq 0$, there basically exist two objective functions that can be used to measure the approximation error: 1) least squares error or 2) the generalized Kullback-Leibler divergence. Before further detailing the two different optimization methods, we remodel Equation 1.9 as a probabilistic model with additive noise as

$$V_{m\times n} = W_{m\times k}H_{k\times n} + E_{m\times n} \tag{1.13}$$

where $E_{m\times n}$ is the noise matrix. Depending on the assumption on the noise, the two methods can be derived as follows [104]:

Least squares error based optimization

Supposing that the noise follows a Gaussian distribution, we can write

$$p(V_{ij}\,|W,H\,) = \frac{1}{\sqrt{2\pi}\sigma_{ij}} e^{-\frac{1}{2}\left(\frac{V_{ij}-(WH)_{ij}}{\sigma_{ij}}\right)^2} \tag{1.14}$$

where $\sigma_{ij}(i \in \{1,2,...m\}, j \in \{1,2,...n\})$ is a weight for each observed sample. Assuming that each data is observed independently, the overall distribution of the model is

$$p(V\,|W,H\,) = \prod_{ij} p(V_{ij}\,|W,H\,) \tag{1.15}$$

and according to the maximum likelihood theory [7] we come to log likelihood function as

$$L(W,H) = \sum_{ij} \frac{[V_{ij} - (WH)_{ij}]^2}{2\sigma_{ij}^2} + \sum_{ij} \log(\sqrt{2\pi}\sigma_{ij}) \tag{1.16}$$

Now supposing that every observation is of the same importance, ie. $\sigma_{ij} = 1$, and neglecting the coefficient and constant terms, we arrive at

$$L_{LS}(W,H) = \sum_{ij} [V_{ij} - (WH)_{ij}]^2 \tag{1.17}$$

Equation 1.17 will be used as the objective function for the least squares error approach.

So the optimization problem can be formulated as

$$\begin{aligned} &\min\ L_{LS}(W,H) = \textstyle\sum_{ij} [V_{ij} - (WH)_{ij}]^2 \\ &s.t.\ \ W,H \geq 0 \end{aligned} \tag{1.18}$$

Computing the gradients on W and H, we get:

$$\begin{aligned} \tfrac{\partial L_{LS}}{\partial W_{ik}} &= -2[(VH^T)_{ik} - (WHH^T)_{ik}] \\ \tfrac{\partial L_{LS}}{\partial H_{kj}} &= -2[(W^TV)_{kj} - (W^TWH)_{kj}] \end{aligned} \tag{1.19}$$

which lead to an additive update rule for minimization using gradient descent:

$$\begin{aligned} W_{ik} &\leftarrow W_{ik} + \phi_{ik}[(VH^T)_{ik} - (WHH^T)_{ik}] \\ H_{kj} &\leftarrow H_{kj} + \varphi_{kj}[(W^TV)_{kj} - (W^TWH)_{kj}] \end{aligned} \tag{1.20}$$

if we choose $\phi_{ik} = \frac{W_{ik}}{(WHH^T)_{ik}}$, $\varphi_{kj} = \frac{H_{kj}}{(W^TWH)_{kj}}$, we finally obtain a simple multiplicative update rule:

$$\begin{aligned} W_{ik} &\leftarrow W_{ik} \tfrac{(VH^T)_{ik}}{(WHH^T)_{ik}} \\ H_{kj} &\leftarrow H_{kj} \tfrac{(W^TV)_{kj}}{(W^TWH)_{kj}} \end{aligned} \tag{1.21}$$

Generalized Kullback-Leibler divergence based optimization

In the case of generalized KL divergence, the noise is supposed to follow the Poisson

[7] https://en.wikipedia.org/wiki/Maximum_likelihood_estimation

distribution, therefore:

$$p(V_{ij}\,|W,H\,) = \frac{(WH)_{ij}^{V_{ij}}}{V_{ij}!}e^{-(WH)_{ij}} \tag{1.22}$$

where $V_{ij}!$ is the factorial of V_{ij}. By using the same method under the same condition of independence as described above, the likelihood function is

$$L(W,H) = \sum_{ij}\left[V_{ij}\log\left(WH\right)_{ij} - (WH)_{ij} - \log(V_{ij}!)\right] \tag{1.23}$$

Note that V_{ij} is actually a constant, so is $log(V_{ij}!)$. Therefore the task of maximizing the likelihood is equivalent to minimizing the generalized Kullback-Leibler divergence between V and WH:

$$L_{KL}(W,H) = \sum_{ij}\left[V_{ij}\log\frac{V_{ij}}{\left(WH\right)_{ij}} - V_{ij} + (WH)_{ij}\right] \tag{1.24}$$

Note that $V_{ij}\log(V_{ij})$ is added in Equation 1.24 yet does not interfere the effect of minimization.

Now the optimization problem is

$$\begin{aligned} &\min\ L_{KL}(W,H) = \textstyle\sum_{ij}\left[V_{ij}\log\frac{V_{ij}}{(WH)_{ij}} - V_{ij} + (WH)_{ij}\right] \\ &s.t.\ \ W,H \geq 0 \end{aligned} \tag{1.25}$$

then once again through the calculation of gradient, we first get the additive updating rule as

$$\begin{aligned} W_{ik} &\leftarrow W_{ik} + \phi_{ik}[\textstyle\sum_{j} H_{kj}\frac{V_{ij}}{(WH)_{ij}} - \sum_{j} H_{kj}] \\ H_{kj} &\leftarrow H_{kj} + \varphi_{kj}[\textstyle\sum_{i} W_{ik}\frac{V_{ij}}{(WH)_{ij}} - \sum_{i} W_{ik}] \end{aligned} \tag{1.26}$$

by setting the learning step as $\phi_{ik} = \frac{W_{ik}}{\sum_{j} H_{kj}}$ and $\varphi_{kj} = \frac{H_{kj}}{\sum_{i} W_{ik}}$, we arrive at the multiplicative iteration rule corresponding to generalized KL divergence as

$$\begin{aligned} W_{ik} &\leftarrow W_{ik}\frac{\sum_{j} H_{kj}\frac{V_{ij}}{(WH)_{ij}}}{\sum_{j} H_{kj}} \\ H_{kj} &\leftarrow H_{kj}\frac{\sum_{i} W_{ik}\frac{V_{ij}}{(WH)_{ij}}}{\sum_{i} W_{ik}} \end{aligned} \tag{1.27}$$

The theoretical proof of convergence regarding both methods is detailed in [112].

Apart from the above two classical rules of update, other methods for the iteration exist, for example projected gradient descent [117, 118], active set [68, 103], block principal pivoting [104], alternating least squares [17, 193], etc. However all current methods can only guarantee the convergence to a local minimum.

As mentioned before, the strongest advantage of NMF consists in the interpretability supported by its non-negative constraints and the fact that its model can be well explained theoretically. However, NMF is highly initialization-dependent, in other words easy to trap in a local optimum. Moreover, the results of NMF is not unique. Consider for example a given WH decomposition:

$$\begin{aligned} WH &= WBB^{-1}H \\ \hat{W} &= WB, \hat{H} = B^{-1}H \end{aligned} \tag{1.28}$$

If B is an invertible matrix, then there will possibly be unlimited pairs of $\hat{W}$ and $\hat{H}$. The other significant drawback lies in the determination of the number of components k, which still remains an open issue.

In the next chapter, we will detail our solutions to the two problems of initializing NMF and choosing k (the number of NMF basis in W) regarding word-referent learning scenarios.

1.4.2.4 SV-NMF

In the early phase of the Ph.D project, the determination of number of components k in Equation 1.9, has been systematically studied. As an indispensable step of applying classical NMF methods, the number k, which is vital to the decomposed feature presentation, has to be chosen manually. To tackle this hard issue, many researches have studied various criteria and rules [195, 167, 140, 130, 114, 97, 92, 63, 32, 7], however, as reported in [122], none of the above methods for the purpose of estimating k seems obviously better than the others.

Recent studies concerning the choosing of k include beta process sparse NMF [115], automatic relevance determination projective nonnegative matrix factorization (ARDPNMF) [199] and SV-NMF (single-class SVM based NMF) [58]. In the experimental evaluations to be described in the subsequent chapters, only SV-NMF is found to produce somewhat satisfactory results. Therefore SV-NMF is introduced here and will be used for the comparative experiments in Chapter 3.

Convex cone interpretation

Unlike the traditional algebra perspective only, SV-NMF tries to find a geometrical explanation of NMF. Figure 1.14 is an illustration of a two dimensional convex cone where observed data are distributed. Since each data point represents a vector by linking the origin, a cone area is formed by finding two vectors (denoted as w_1 and w_2) with the largest aperture angle, capable of encompassing all other data points. The very property of a convex cone is that through the non-negative linear combination of "border vectors", any data encompassed inside can be represented. This property can also be generalized to higher dimensional space, where number of "border vectors" (called vertices of the cone) might be unequal to the

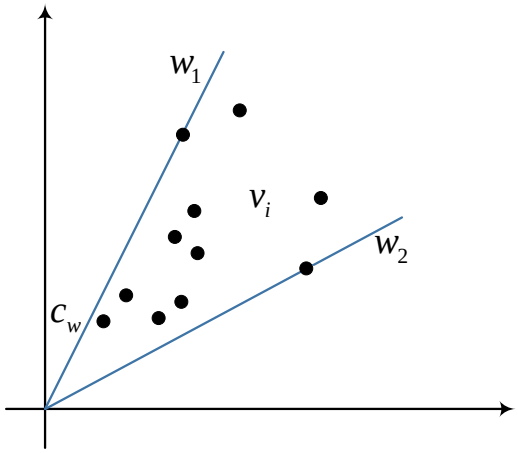


Figure 1.14: Example of convex cone in 2D space.

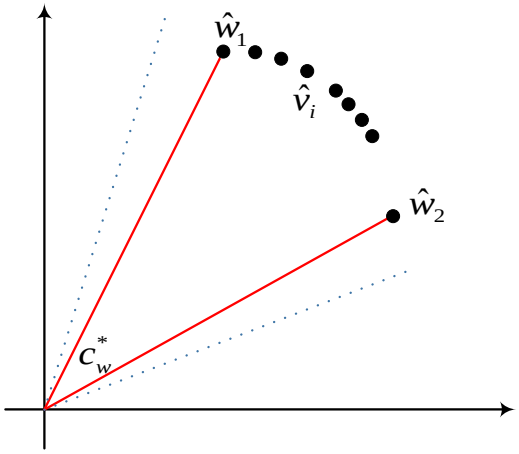


Figure 1.15: Example of normalized convex cone in 2D space.

number of dimensions, and mathematically the convex cone can be represented as

$$c_W = \{ \sum_{k=1}^{k=K} \lambda_k w_k \,|\lambda_k \geq 0 \} \tag{1.29}$$

where w_k represents a vertex and λ_k denotes its non-negative coefficient. Now if we define the space as m dimension, then the n samples can be termed as $V_{m\times n} = [\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_n]$, then the vertices $W_{m\times k} = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_k]$ and coefficients $H_{k\times n} = [\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_n]$. Thus comes the linear combination as

$$\begin{aligned} V_{m\times n} &= W_{m\times k} H_{k\times n} \\ [\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_n] &= [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_k]\,[\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_n] \end{aligned} \tag{1.30}$$

$H_{k\times n}$ is by definition as non-negative, if we further define that the cone exists in the area only composed of the non-negative half of each dimension, in other words, $V_{ij}, W_{ij} \geq 0$, we obtain the non-negative constraints of NMF. Therefore, the convex cone is a valid geometric model of NMF, which has the following properties: 1) the NMF basis acquires a clear interpretation of its role to form the observed data; 2) if we further normalize the $\mathbf{w_i}(i \in \{1, 2, ..., K\})$ (see Figure 1.15), the data points which form the "smallest cone" (ie. conic hull) can be uniquely found, thus solving the problem of choosing k.

Computation using SV-NMF

In order to solve the approximation of $V \approx WH$, the SV-NMF method first compute the basis matrix W using the Single-class SVM algorithm, then followed by the computation of the coefficients of H.

Derived from the traditional Support Vector Machine (SVM) [163], which tries to separate data patterns with the maximum margin by searching the optimal hyper plane in the feature

space, single class SVM (also known as one class SVM) first defines only one class from the training data and then solves the problem of data discrimination by setting up a hyperplane that separates the feature vectors (located on the positive side of the hyperplane) from the origin with maximum margin. The hyperplane will be calculated in the final as defined by a set of support vectors automatically chosen from the training data. One detail which we should note is the fact that in the formation of the hyperplane the margin errors always exist, hence a parameter ν (which is often noted nu later on) is used as a penalization parameter allowing for a trade-off, which proves to be both an upper bound on the fraction of outliers and a lower bound on the fraction of support vectors, or even both asymptotically equal to them under mild conditions on the form of the data distribution and the kernel [161].

SV-NMF seeks to find a conic hull of input data to determine W. Since normalization is applied on all input data, all observations are on a unit hypersphere, as illustrated by Figure 1.15. The smallest conic hull can be found by seeking the hyperplane separating the data from the origin with the maximum margin using single class SVM. By using this method, the margin support vectors (which are also the vertices of the smallest conic hull, see Figure 1.16) will be computed and serve as the basis vectors w_k in W. Obviously, another result is the automatic determination of k, which is equal to the number of margin support vectors.

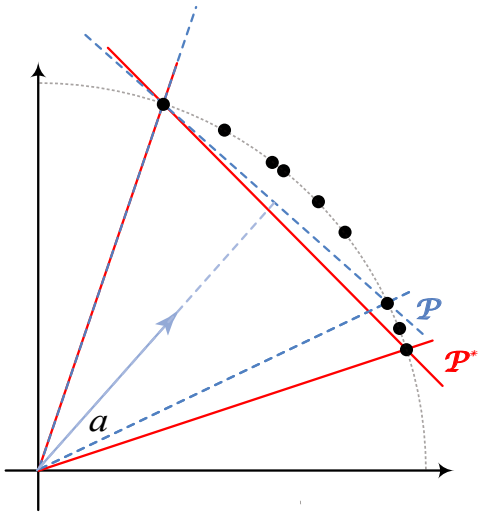


Figure 1.16: Example of two different hyperplanes for input data on a unit hypersphere in 2D space computed by SV-NMF with different nu values. Data points which lie on the hyperplane are the margin support vectors and the vertices of the smallest conic hull. The hyperplane in dashed line shows some errors, failing to incorporate all the input data.

As for the computation of H, it is a simple linear regression problem with positive constraints after obtaining W formulated as :

$$\begin{aligned} &\min_{h_j} C(h_j) = \|v_j - W h_j\|^2 \\ &s.t.\ h_{k,j} \geq 0,\ k \in \{1, ..., K\},\ j \in \{1, ..., n\} \end{aligned} \tag{1.31}$$

The detailed solution can be found in [111].

1.4.3 Probabilistic topic models

Probabilistic topic models have been designed to solve some defects in the previous models, such as their limitation in dealing with *polysemy* defined as the problem of words having several different meanings such as *bank* in the two sentences: "The currency reserve is sufficient in this *bank*" and "There are windmills on the other side of the *bank*". Intuitively, a probabilistic representation is better able to capture these possible variations than a representation as a vector used in the previous methods.

We will present two algorithms : Probabilistic Latent Semantic Analysis (pLSA) and Latent Dirichlet Association (LDA). In both models, topic allocation per document and word allocation per topic are presented as probabilities. An important difference is that the number of parameters of pLSA increases linearly as new documents appear, which will bring problems of increasing computational burden and over-fitting, while LDA sets hyper parameters to control the generation of allocations in a probabilistic manner that could be timely modified in processing the documents.

1.4.3.1 pLSA

Probabilistic Latent Semantic Analysis (pLSA) [88], which is regarded as a probabilistic modeling of LSA, depicts the generation of a document by introducing a latent class of topic as $z_k \in \{z_1, z_2, ..., z_K\}$. In a whole corpus of M documents, $p(d_i)$ denotes the appearance probability of the $i_{th}(i = 1, 2, ..., M)$ document, $p(z_k|d_i)$ the semantic distribution of topics in a document and $p(w_j|z_k)$ the probabilistic distribution of words regarding a topic.

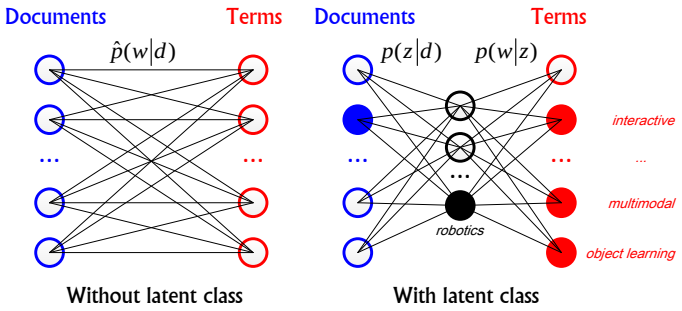


Figure 1.17: Three layers of pLSA. [137].

Figure 1.17 illustrates how pLSA models the document generation process using a neural-network analogy. First, a document is selected according to $p(d_i)$, then at the semantic layer a topic z_k is chosen conforming the distribution of $p(z_k|d_i)$, finally given a particular topic z_k, a word comes out following $p(w_j|z_k)$.

Another view of this model could be illustrated by a graphical model in which a filled circle represents an observable variable, a hallow circle a latent (thus unobservable) variable,

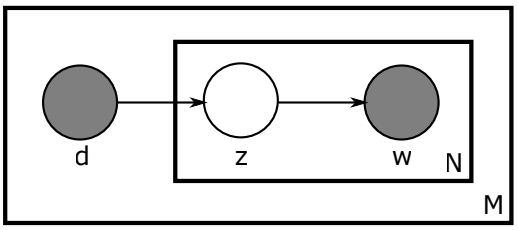


Figure 1.18: Diagram of pLSA.

an arrow the dependence of the variable pointed by its tip on the variable connected by its tail, a square with a capital letter (eg. M or N) the repetition of the enclosed procedures. In Figure 1.18, d (document) and w (word) are observable variables and z (latent class, ie. topic) is primarily hidden.

In order to find document-topic and topic-word distribution, we will maximize the joint distribution of (d_i, w_j):

$$p(d_i, w_j) = p(d_i)p(\,w_j|\,d_i) \tag{1.32}$$

$$p(\,w_j|\,d_i) = \sum_{k=1}^{K} p(\,w_j|\,z_k)p(\,z_k|\,d_i) \tag{1.33}$$

where $p(\,w_j|\,z_k)$ and $p(\,z_k|\,d_i)$ are two distribution parameters to be estimated, which conform to multinomial distribution.

As a BoWs model where each choice of word is regarded independent, the joint probability distribution of the pLSA model concerning collecting words for all documents is $\mathcal{L} = p(D, W) = \prod_{i=1}^{M}\prod_{j=1}^{N_i} p(d_i, w_j)^{n(d_i,w_j)}$, where $n(d_i, w_j)$ represents the number of word w_j appeared in document d_i. Besides, the indirect causal effect takes place in the process of $d \to z \to w$, which gives conditional independence as $p(w_j, d_i|z_{\,k}) = p(w_j|z_{\,k})p(d_i|z_{\,k})$. Therefore the log likelihood function is formulated as

$$\begin{aligned}
\log \mathcal{L} &= \log(\prod_{i=1}^{M}\prod_{j=1}^{N_i} p(d_i, w_j)^{n(d_i,w_j)}) \\
&= \sum_{i=1}^{M}\sum_{j=1}^{N_i} n(d_i, w_j) \log p(d_i, w_j) \\
&= \sum_{i=1}^{M}\sum_{j=1}^{N_i} n(d_i, w_j) \log[p(d_i)\sum_{k=1}^{K} p(w_j\,|z_k\,)p(z_k\,|d_i\,)] \\
&= \sum_{i=1}^{M} n(d_i)\left(\log p(d_i) + \sum_{j=1}^{N_i}\frac{n(d_i, w_j)}{n(d_i)} \log \sum_{k=1}^{K} p(w_j\,|z_k\,)p(z_k\,|d_i\,)\right) \\
&= \sum_{i=1}^{M} n(d_i)\left(\log p(d_i) + \sum_{j=1}^{N_i}\frac{n(d_i, w_j)}{n(d_i)} \log \sum_{k=1}^{K} \phi_{k,j}\theta_{i,k}\right)
\end{aligned} \tag{1.34}$$

where $n(d_i) = \sum_{j=1}^{N_i} n(d_i, w_j)$. And the estimations of $\phi_{k,j}$ and $\theta_{i,k}$ can be solved as an optimization problem by using maximum likelihood estimation (MLE) and expectation-maximization (EM) algorithm.

The most widely used applications of pLSA include large-scale image retrieval [133, 90, 89, 22], which usually relies on one modality of visual features. [116] proposed a multilayer multimodal pLSA model, making use of both visual and tag based modalities. This model is proved effective for image retrieval when it is combined with its proposed fast initialization settings.

Note that NMF and pLSA are closely related and thus convertible. But we should pay very careful attention to the difference regarding notations between Equation 1.9 and Figure 1.18, because the term "feature" and "sample" in Equation 1.9 corresponds to "word" and "document" here, in other words $n = M$ and $m = N$. Then we define that $\hat{V} m \times n$ is the co-occurrence matrix of m words and n documents, whose element $\hat{V}_{m,n}$ is the count of m_{th} word in the n_{th} document and certainly non-negative. After the normalization of each column vector of $\hat{V}_{m\times n}$, we get $V_{m\times n}$ such that $\sum_{i=1}^{m} V_{i,j} = 1$. And $V_{i,j}$ is just the conditional distribution of word $\mathbf{w}$ given document $\mathbf{d}$, ie. $p(\mathbf{w}\,|\mathbf{d}\,)$. Similar to Equation 1.9, we get the decomposition as

$$
\begin{aligned}
V_{m\times n} &= p(\mathbf{w}\,|\mathbf{d}\,) \\
&= \Big(\ p(\mathbf{w}|\,d_1), p(\mathbf{w}|\,d_2), \cdots, p(\mathbf{w}|\,d_n)\ \Big) \\
&= \Big(\ \sum_{r=1}^{k} p(\,z_r|\,d_1)p(\mathbf{w}|\,z_r), \sum_{r=1}^{k} p(\,z_r|\,d_2)p(\mathbf{w}|\,z_r), \cdots, \sum_{r=1}^{k} p(\,z_r|\,d_m)p(\mathbf{w}|\,z_r)\ \Big) \\
&= \Big(\ p(\mathbf{w}|\,z_1), p(\mathbf{w}|\,z_2), \cdots, p(\mathbf{w}|\,z_k)\ \Big) \begin{pmatrix} p(\,z_1|\,d_1) & p(\,z_1|\,d_2) & \cdots & p(\,z_k|\,d_n) \\ p(\,z_2|\,d_1) & p(\,z_2|\,d_2) & \cdots & p(\,z_2|\,d_n) \\ \cdots & \cdots & \cdots & \cdots \\ p(\,z_k|\,d_1) & p(\,z_k|\,d_2) & \cdots & p(\,z_k|\,d_n) \end{pmatrix} \\
&= p(\mathbf{w}|\,\mathbf{z})\Big(\ p(\mathbf{z}|\,d_1), p(\mathbf{z}|\,d_2), \cdots, p(\mathbf{z}|\,d_n)\ \Big) \\
&= p(\mathbf{w}|\,\mathbf{z})p(\mathbf{z}|\,\mathbf{d}) \\
&= W_{m\times k}H_{k\times n}
\end{aligned}
\tag{1.35}
$$

with the constraints that $\sum_{i=1}^{m} W_{i,r} = 1(r \in \{1, 2, ..., k\})$, $\sum_{r=1}^{k} H_{r,j} = 1(j \in \{1, 2, ..., n\})$. In fact, [67] argues that PLSA solves NMF with KL divergence, and [51, 52] also state that both NMF and pLSA optimize the same objective function, therefore these two models are highly equivalent.

1.4.3.2 LDA

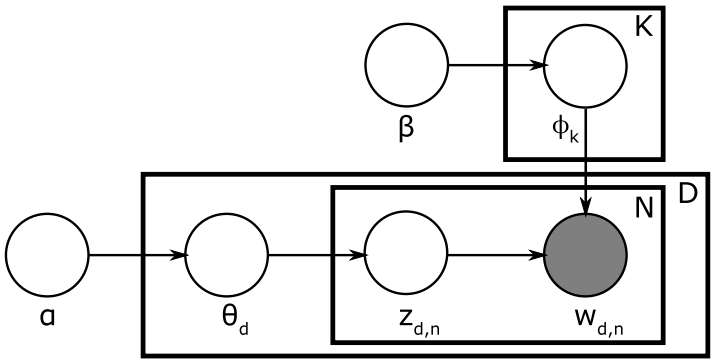


Figure 1.19: Diagram of LDA, see text for details.

The structure of LDA [20] is illustrated in Figure 1.19, by using the graphical model as described before, where D, K, N represent the total number of documents, topics and words within each document, θ_d the topic distribution for the document d ($d = \{1, 2, ..., D\}$), ϕ_k the distribution over words for the topic k ($k = \{1, 2, ..., K\}$), $w_{d,n}$ the observed n_{th} word in document d, $z_{d,n}$ the topic assignment of $w_{d,n}$. Special attention should be paid to α and β, which are parameters of the Dirichlet prior on θ_d and ϕ_k which are regarded as random variables. This is the main difference with pLSA which on the contrary holds a deterministic point of view towards the distributions of both topics per document and words per topic. That's why LDA is regarded as a Bayesianized version of pLSA.

The events of choosing a topic given a document and choosing a word given a topic conform to a multinomial distribution and should therefore be conditioned on a prior Dirichlet distribution (by making use the conjugate prior theory, the conjugate prior of multinomial distribution [20] is a Dirichlet distribution). By choosing the prior distribution of topic and word generation process parameterized by α and β respectively, their posterior probabilities will still remain Dirichlet distributed.

The generative model of LDA for a corpus of D documents goes as follow:

1. Compute corpus-document distribution $\theta_d \sim Dir(\alpha)$, where $d \in \{1, ..., D\}$ and $Dir(\alpha)$ is a Dirichlet distribution under the parameter of α;

2. Compute document-topic distribution $\phi_k \sim Dir(\beta)$, where $k \in \{1, ..., K\}$ and $Dir(\beta)$ is the Dirichlet distribution under the parameter of β;

3. For the generation of word at position d, n, where $d \in \{1, ..., D\}$ and $n \in \{1, ..., N_d\}$ (N_d is the count of words in the d_th document, and $N = \sum_{d=1}^{D} N_d$):

   (a) Choose a topic $z_{d,n} \sim Multinomial(\theta_d)$

   (b) Choose a word $w_{d,n} \sim Multinomial(\phi_{z_{d,n}})$

Therefore, we come to the total distribution as

$$p(\mathbf{w}, \mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\phi}; \alpha, \beta) = \prod_{k=1}^{K} p(\phi_k; \beta) \prod_{d=1}^{D} p(\theta_d; \alpha) \prod_{n=1}^{N_d} p(z_{d,n} \,|\theta_d) p(w_{d,n} \,|\phi_k, z_{d,n}) \tag{1.36}$$

However, the posterior (ie. the conditional distribution of the topic structure given the observed documents) in which $\mathbf{w}$ is observable while $\mathbf{z}$, $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ are hidden random variables as shown below

$$p(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\phi} \,|\mathbf{w}\, ; \alpha, \beta) = \frac{p(\mathbf{w}, \mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\phi}; \alpha, \beta)}{p(\mathbf{w}; \alpha, \beta)} \tag{1.37}$$

can not be computed directly due to the fact that $p(\mathbf{w}; \alpha, \beta)$ is intractable, which can be explained by two aspects: 1). from the practical point of view, the marginal probability is through the summation of the joint distribution over every possible instantiation of the hidden topic structure (concerning $\mathbf{z}$, $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$), exponentially large in number thus appearing impractical to calculate; 2). from the mathematical point of view, the marginal probability is by integrating the joint probability in terms simultaneously of $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$, but the problem of *coupling* between $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ makes the calculation of integral nearly impossible (as detailed in the original paper [20]).

Therefore, in order to infer the posterior distribution and therefore the latent topics, various approximation methods are developed, which generally fall into two categories – *variational algorithms* and *sampling-based algorithms*. We give details about these approaches in Appendix A. The variational algorithms are usually highly dependent on their initialization of parameters and get often trapped in local optimum. We therefore preferred sampling-based algorithms, in particular Gibbs Sampling, which are more robust for inferring posterior variables, despite the fact that variational methods are faster in terms of convergence.

1.5 Learning strategies

We now focus on the learning strategy, i.e., the way information is gathered in order to learn the correct word-referent associations. As introduced in Section 1.1, there are two main problems we need to address: the ambiguity of the word and referent when multiple words and objects are presented and the potential method to speed up the incremental learning process, which might be used in the interactive learning scenarios. These two problems will be addressed by *cross-situational learning* and *active learning*.

1.5.1 Cross-situational learning

We should first note that when trying to learn a word-referent association, there are possible ambiguities both on the side of the referents and on the side of the words.

The first type of ambiguity appears in the presence of multiple referent, for example when

a scene containing two objects is described by "There are object A in color B and object C in color D", there is ambiguity on which of the two unknown features (out of four) are indicated by A, B and C, D. Another illustration has been given by Quine's experiment on the "Gavagai" problem [145], where the word "Gavagai" is pronounced while pointing to a rabbit in a field, and therefore its meaning can be "rabbit", "field", or even the color of the rabbit. Potential solutions to reduce these ambiguities include the joint attention [86] which makes both teacher and learner focus on the same object, but not all ambiguities can be solved in this way. For example, saying "red apple" in association to a red apple still leaves the ambiguity about which word points to the type of object and which one points to its color.

The second type of ambiguity is linked to language understanding. For example, when describing a scene with a single object with "Look at this red apple", the verb "look", preposition "at" and pronoun "this" are actually not related to the object identity and have other functions in the sentence. The syntactic constraints of the language itself [70] can obviously help identify the function of the words and therefore make it clearer about which one is potentially describing an object for example.

However, these ambiguities can be solved in a larger context by using *cross-situational learning* [168, 203, 95, 150, 94]. The general idea of this strategy is that by displaying various objects and associated words in different situations, the learner is supposed to analyze the commonalities between them and solve the ambiguities so as to recover the correct word-referent associations. For human beings, it has been observed that as early as 12 months old we are naturally relying on cross-situational learning [174] so as to solve the above referring ambiguities. Take the series of studies performed by Kachergis et al. (eg. in [96]) for instance, participants are asked to learn the referent of novel words by watching a series of training trials, on each of which, learners see an array of unfamiliar objects while hearing pseudowords. The ambiguity occurs due to the unspecified intention about the referent of each pseudoword on a given trial, although each word is sure to refer to a single onscreen object, however, by observing the pairs occurring on multiple trials spread across trainings as well as appearing with different concurrent pairs, people can learn some of the intended word-object pairings given that the accumulated word-object co-occurrences are well utilized in some fashion.

1.5.2 Active learning

An efficient strategy used by humans to improve learning speed is the use of active learning with which the learner is able to actively choose future training samples instead of simply passively observing incoming information.

Let's first introduce the data acquisition possibilities for machine learning purpose. The first distinction is between *batch learning* where learning takes place on the entire training data and *incremental learning* in which the whole training data are divided and received sequentially by the learner in order to progress step by step. This second approach is interesting since in some cases it is computationally infeasible to train over the entire dataset and

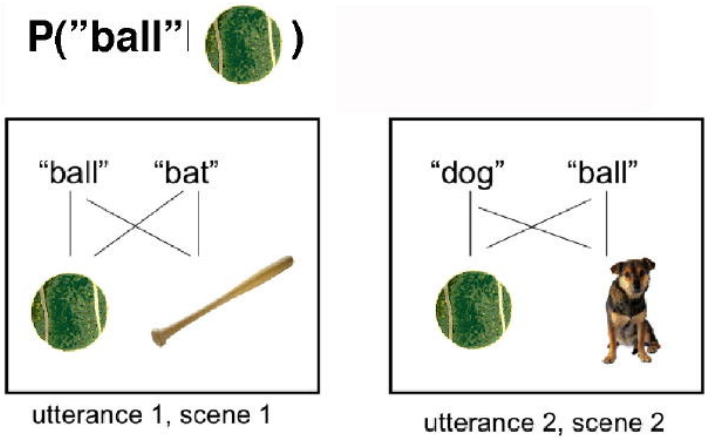


Figure 1.20: A cross-situational learning demo in which a young learner by calculating co-occurrences frequencies across these two trials can find the proper mapping of "Ball" to BALL (from [174]).

moreover it provides adaptability to the changing environments. Therefore the *incremental learning* gets more and more attention, especially from a developmental learning perspective.

Incremental learning can be further categorized as *off-line learning* and *on-line learning*. *Off-line learning* assumes that the training data are known to the learner in advance and have no restriction on the computation time. Although seemingly not so adaptable from an algorithmical point of view, it has significance of further developing (ie. enhancing, consolidating) acquired skills between practice sessions (eg. during sleep) [147]. *On-line learning*, on the contrary, does not require the complete knowledge of training data and takes place in a sequential order, i.e., it should generate a temporary result which is executable for action or decision after each data, even if it is not optimal. Recently, the methods to improve the efficiency of incremental learning, in particular on-line learning, have been investigated and among the theories of computationally modeling the thinking and exploration behaviors of humans, active learning has been a hot topic.

Incremental and on-line learning is therefore influenced by the order of training data. In *passive* learning, the learner can only follow the sequential training data whose order has already been determined, while in *active* learning, *motivations* will serve as a driving force for the learner to autonomously choose which object is to be learned next. [139] gives a detailed classification of various motivations and in particular makes a classification between "intrinsic and extrinsic" (also see [57]). A general definition is that a motivation is intrinsic if it is formed by factors exclusively from the learner or the continuity of the action, otherwise it is extrinsic.

Intrinsic motivations, from a psychological point of view, are rewarding situations which include novelty, surprise, incongruity, and complexity (see [15]) or the self-engagement in activities, requiring skills just above their current level, which represents a learning challenge [44]. It is found in neurosciences that dopamine neurons in the midbrain report the error, predicting expected reward delivery, not only for the extrinsic reward but also for the intrinsic motivation associated with novelty and exploration [47, 98]. Therefore, applying intrinsic

motivation from the computational modeling perspective, a mechanism should be built in order to evaluate operationally the degree of "novelty", "surprise", "complexity" or "challenge" together with an associated reward whose maximal value corresponds to a proper degree of the above attributes so as to achieve the most rewarding result. This modeling is denoted as "artificial curiosity".

The problem of "artificial curiosity" in the context of machine learning is therefore to implement the rule to choose the next sample for learning in order to either minimize the number of samples necessary to achieve a given level of performance or maximize the level of performance with a certain limited number of examples. While the theory of "optimal experiment design" [60] is regarded as one of the fundamental works, other particular learning models like statistical models [39], neural networks [82] have already adopted active learning strategy to enhance the learning performance in applications like language learning [100].

In the field of developmental robotics, [138] reviewed computational models of intrinsic movitation and pointed out roughly two categories, that is *Error Maximization based* and *Progress Maximization based*.

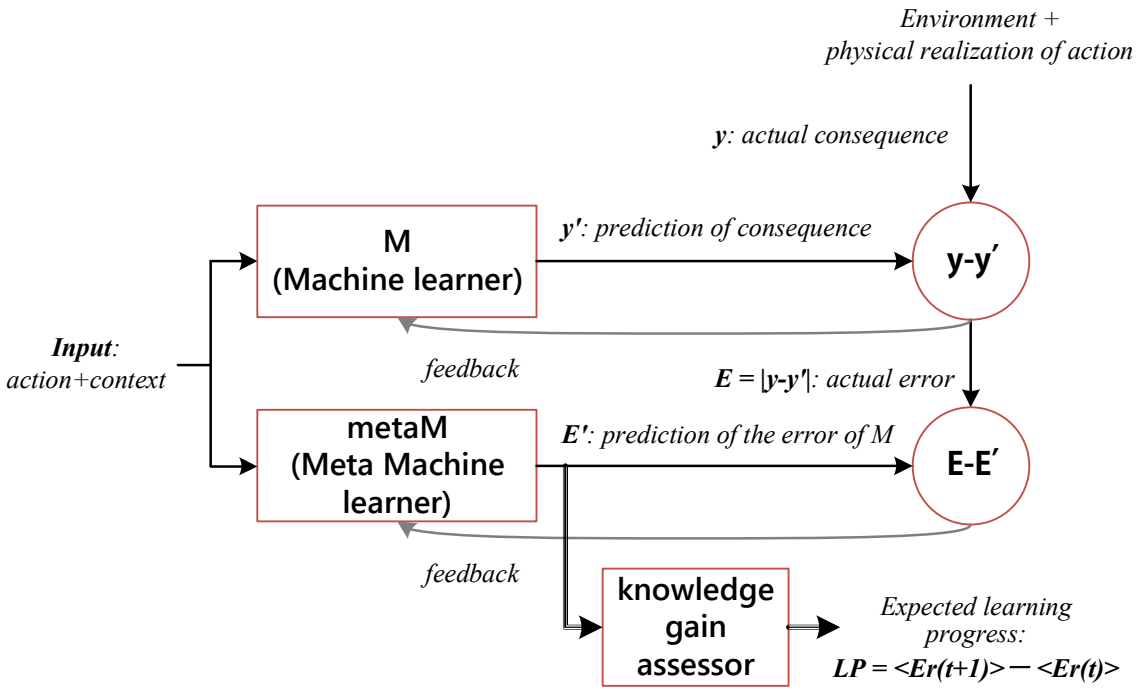


Figure 1.21: The architecture of two intrinsic motivation systems in [138], and $< Er(t+1) >$ and $< Er(t) >$ represent the expected mean error rate and the mean error rate in the close past respectively.

For the understanding of the two systems, the notations of **M** for the learner and **metaM** for the meta-learner from [138] are used here (see Figure 1.21). **M** predicts the consequence ($y\prime$) of an action performed in a sensorimotor context and $y\prime$ will be compared to the actual consequence y (as the ground truth) for the actual error $E = |y\prime - y|$. **metaM** is used for the evaluation (or the prediction, denoted as $E\prime$) of the actual error E. For the actual error at a specific learning step t, we denote it as E_t, and the derivative of the actual error E is

defined to signify the learning progress as $LP = E_{t+1} - E_t$, and estimated by the module **KGA** (knowledge gain assessor). Yet in real calculations, E_{t+1} and E_t are usually replaced by the mean error rate, in the proximity of learning step $t+1$ and t, noted as $< Er(t+1) >$ and $< Er(t) >$.

In the *Error Maximization based* system, the action that is chosen at each step is the one which is predicted by **metaM** to generate the maximum actual error by **M**. The precondition of applying this method is that **M** should have an effective capacity to learn in all the sensorimotor space so as to be able to decrease this error in all cases. However, in real scenarios, there are situations where the learner may be unable to learn correctly a model. These situations will cause a constant and high error, thus leading the system to get stuck in one area, although showing the highest predicted error yet making zero progress from the learner's perspective.

Progress Maximization based system, instead of depending on the prediction of error to choose an action, prefers the choice of the next action which would bring the maximum learning progress. This choice prevents the system to get stuck in area where the error remains high because the learning progress appears small there. This approach therefore improves the learning speed as well as the coverage of the learnable space while avoiding unlearnable situations. In practice, the estimation of the learning progress is based on the prediction of the continuity of the error curves based on the error data (calculated as mean error rate) recorded in the near past steps. By using the proper modeling of the sensorimotor context, via categorization and measuring the similarity of situations [160], this system has extended applications and provides a way to control the complexity of learning situations [164] in the frame of developmental robotics [138]. The main practical limitation of this strategy is that the real word consequence should be accessible for the prediction of learning progress.

Examples of applying active learning strategies in word-referent learning could be found, for instance, in Kachergis' study in [96], where the active learning strategy of immediate repetition of word-referent pairs to deal with referential ambiguities was evidenced in human behavior and then modeled computationally. More interestingly, it is found that humans who repeat only one pair per trial (which is an easy way to rapidly infer this pair) perform worse than those who repeat multiple pairs per trial.

Schueller [164] studied the application of active learning to Naming Games where, at each learning step among two agents (a "speaker" and a "hearer"), the decision of whether to explore (ie. try to learn an unknown word) or teach (ie. review or consolidate what has been learned) meaning-word pairs is addressed by applying several active learning policies. Besides the *Naive Strategy* where the meaning-choice policy is simply uniform (resulting in random sample learning), *Success-Threshold Strategy* suggests exploring new meanings other than involving the already associated meanings once the success rate of past records exceeds a certain threshold (that is to say only when the learner seems confident enough about what has been learned). They also proposed the *Last Result Strategy* where the agent considers only its last result of interaction and explore in the next trial only if he succeeded for the last time. Finally, they proposed *Decision Vector Strategies* that ignore results of past interactions and base decision only on the currently known vocabulary size. The decision vector will

indicate at which dictionary size exploration should take place. There are two models to build a decision vector, whose length covers the increasing size of the acquired vocabulary of a "hearer" agent and whose coordinates represent the probability (which for simplicity is set as 1 or 0) of an exploring action at the corresponding vocabulary size. The first model design triggers an exploration only at fixed vocabulary sizes, following a predefined rule in which the gap between adjacent triggering steps is in line with a geometric progression with the ratio of 0.5. Then the second model is denoted as *Gain Maximization Decision Vector Strategy*, in which every coordinate value of the decision vector is set to be 1 or 0 according to the calculation that whether the expected information gain of the vocabulary matrix of the "hearer" is positive or not. The simulation performance shows that *Success-Threshold Strategy* and *Gain Maximization Decision Vector Strategy* could lead to the overall convergence and reach the highest values of acquired information, meanwhile *Last Result Strategy* does not proves much better than the *Naive Strategy*.

Facing situations of both referential and linguistic ambiguities and the challenge to improve the learning speed, active learning methods can be jointly applied with the use of cross-situational learning strategy. In this thesis, we will study how different learning algorithms are suited to an adequate definition of intrinsic motivation in order to support active learning and improve the speed of interactive word referent learning.

1.6 Conclusion

Within the framework of the semiotic square illustrated in Figure 1.1, the word-referent learning model proposed by this thesis focuses on the algorithmic solution for the implementation of infant-inspired learning abilities such as cross-situational learning and active learning and its validations by means of simulation experiments, underlying the future interactive applications with real robots.

An important principle of this thesis is to build models on top of comparatively simplified setup and simple presentations so as to concentrate more on the analysis of algorithms. The simple yet effective Bag of Words (BoW) representation is therefore applied to process visual raw data in terms of shape and color as well as audio voice input which will be converted to text words using existing speech recognition techniques.

As for learning algorithms, the topic model approach is favored mainly due to its capability of discovering latent components, which brings about the flexibility of reaching learning goals as well as the possibility of carrying out more complex learning plans, and we adopt Non Negative Matrix Factorization (NMF) and Latent Dirichlet Association (LDA) as representatives respectively from matrix decomposition methods and probabilistic models to undertake the learning executions in all related experiments. In addition, active learning strategies will be proposed, integrating ideas of some of the previous works into our simplified interactive scenarios, which are to be detailed in the next chapter.

Chapter 2

Cross situational word-meaning association using topic models

Contents

In this chapter, we describe our application of topic models to the word-meaning association learning. We describe first how data are pre-processed as input for our system, and how we apply NMF and LDA to our problem.

2.1 Multimodal signal processing

The goal of signal processing is the preparation of the data in a format well-suited for our proposed computational models. In the following experiments, object perception and signal processing are carried out on two modalities: visual and audio. The visual information provides the descriptions of an object in terms of sub-modalities of shape, color, and HOG while audio information is from the linguistic descriptions of the object by humans. The overall framework is depicted in Figure 2.1.

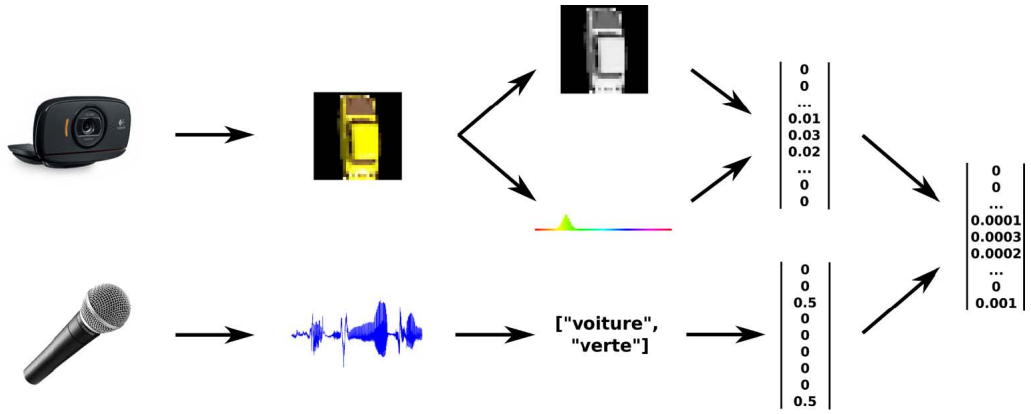


Figure 2.1: Formation of histogram from multimodal information. Note that the shape description using pixel could be replaced by HOG (to be introduced in Section 2.1.1.2) if we adopt HOG as the shape descriptor.

As previously stated, since topic models are well adapted to histogram representations, data are presented in the form of histogram by applying feature descriptors on different modalities. All respective histograms would be concatenated to form a longer histogram as the result of the multimodal observation of an object (see the far right in Figure 2.1), and all these multimodal histograms, each corresponding to an observation of objects, come to form a corpus, we denote as V of vectors V_i $(i = 1, 2, ..., n)$ representing the appearance of an object and an associated sentence pronounced by a human partner.

The first part of each vector is a continuous channel that represents features obtained through computer vision. These features are constructed to represent color (V_i^{color}), shape (V_i^{shape}) or HOG (V_i^{HOG}) of the object, but they could be the results of a more generic feature computation algorithm. The features are encoded as vectors of constant size, and multiple objects of interest are represented (as would be detailed in Section 2.1.4) by summing the description of each individual object, thanks to the fact that the features are histograms, which can be added. The second part of each vector is a binary vector of the size of the dictionary of all known words (V_i^{word}) and represents the word occurrences in the sentence, as will be explained in Section 2.1.2, and the dictionary is created incrementally, starting from an empty dictionary and adding each new word encountered in sentences at the end.

The following sections detail the methodology of acquiring data for each modality.

2.1.1 Visual input

Visual input is acquired through a standard webcam in order to capture shape, color and HOG information. A more advanced RGB-D camera was also used in some experiments in order to benefit from the depth information to improve object segmentation.

2.1.1.1 Object segmentation

Objects are first individually segmented from the camera image. Two approaches were used: the first one using only color information that requires a simple black background for the images, and the second one using depth information that makes it possible to use a less constrained experimental setup. In each of these methods, a reference orientation is computed so that object shape description is invariant to the orientation of the object in the image frame.

Color based object segmentation

In the case of an experimental setup using a black background, the images are first read at the rate of 15 Hz and the black background is removed by using mask functions available in the OpenCV library, which reads the HSV (to be detailed in Chapter2.1.1.2) information of every pixel and accepts only those whose V (ie. Brightness) value is higher than a threshold (a predefined value depending on the lighting condition). The remaining pixels are segmented into connected areas and the group whose size are above a given threshold are extracted as detected object(s) (see Figure 2.2).

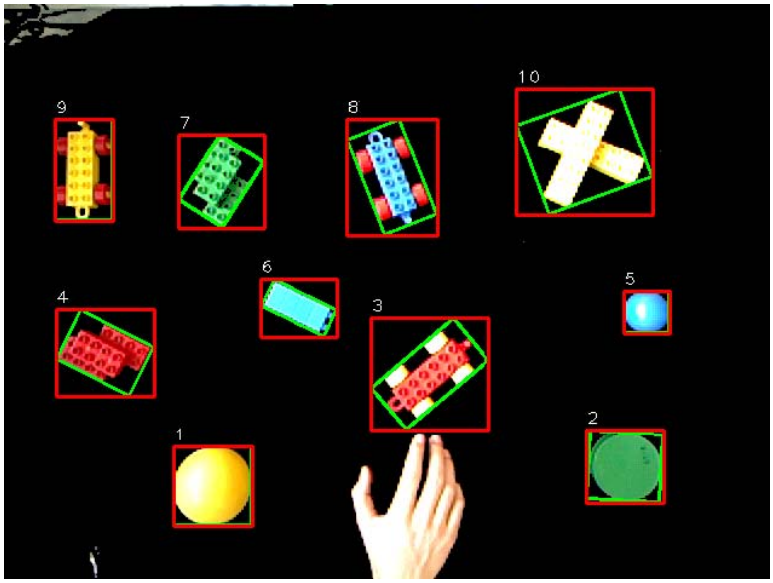


Figure 2.2: Example of object detection on a black background.

One special case that may add confusion to the image extraction is the pixels caused by the forearm and hand of human tutors. In order to avoid this, the group of pixels which are found in contact with the edge of the image are removed.

Another issue is the rotation of object. Indeed, for any asymmetric object, its shape description is highly dependent on the angular position when it is put on a table. Hence, only when all shape descriptions share the same angular position reference, can the shape representations be comparable among objects. We propose a first rotation method in which an object is first enclosed by a minimum area rectangular box[1] (see the green boxes in Figure

[1]Based on the built-in OpenCV functions *cv2.minAreaRect*

2.2) and then a larger rectangle in a standard up-right profile (ie. bounding rectangle, see the red boxes in Figure 2.2) is shaped according to the positions of four corners of the minimum area rectangle. The observed object would then be rotated from the angular position of the green box to its nearest up-right position specified by the red box. This rotation function evidently only works at a local scale because by adapting an object to one of the four standard positions (ie. up, left, down, right), there still remains ambiguities if the object is rotated for more than 90 degrees.

Therefore, a second rotation method is proposed in two steps. First of all, we calculate for any pixel its gradient value by applying the Sobel operator[2]. Since every gradient value corresponds to an angular value, thus a histogram which describes the occurrences of these angular values is generated for all image pixels. Mean shift[3] is then used to detect the mode of this histogram which is used as the angular value for rotation. However, after being rotated by the recommended angular value, there exists situations where the rotated image still remains unstable and would switch among four standard positions. As shown in Figure 2.3, we therefore split the image into four quadrants and we define the final standard angular position as when the sum of all pixel values of *Quadrant 1* and *Quadrant 3* is larger than that of the other two adjacent quadrants.

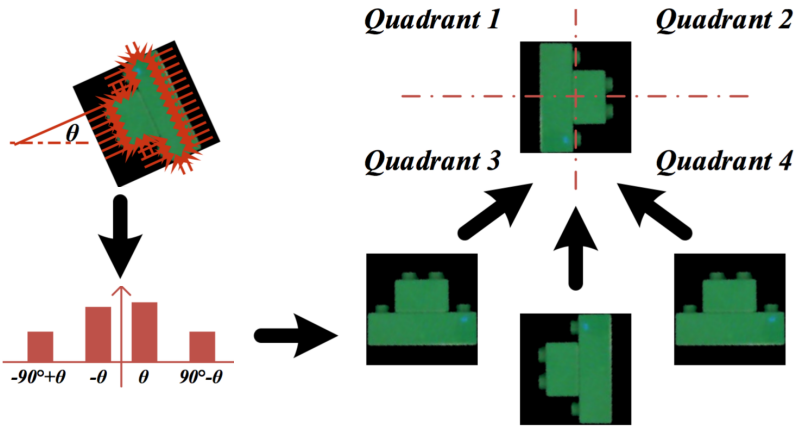


Figure 2.3: Rotation of an image based on gradient orientation and the split of an image into four quadrants.

Depth based object segmentation

The camera with depth sensor (eg. Xtion or Kinect) provides a depth image where each pixel contains a value encoding the object distance in meters (see Figure 2.4). Note that the black regions in the image represent areas where the distance could not be estimated.

We tried a simple depth-based segmentation method which first finds the closest valid point with distance X and adopts a distance threshold $d_{threshold}$ which is user-defined, to

[2]https://en.wikipedia.org/wiki/Sobel_operator
[3]https://en.wikipedia.org/wiki/Mean_shift

Figure 2.4: A demonstration of a depth image

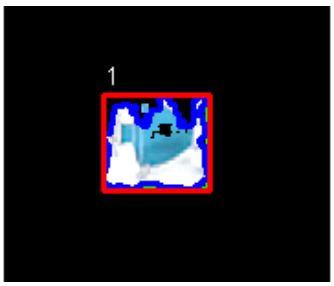


Figure 2.5: Unstable object segmentation

Figure 2.6: In purple the contour enclosing the pixels which meet the threshold and in red the Minimal Area Bounding Rectangle.

select all the points with the depth distance in $[X, X + d_{threshold}]$. We then keep all the corresponding pixels in the color image, in which a contour is searched[4] and then used to indicate possible objects. However, one of the problems with this approach is that due to the imprecision of the camera sensor, the segmented image sometimes does not obtain a clear border and momentary black spots are observed inside the segmented area because of scene noise, as shown in Figure 2.5.

The solution which is applied in our experiments makes some assumptions on the object shape, which are white cubes with drawings in our case. Instead of directly taking the pixels meeting the distance threshold, we use a minimal area bounding box that encloses those pixels with the constraints that $0.65 < \text{Height/Width} < 1.35$ (Figure 2.6). This reduces greatly the instability generated by the noise in the depth image since it prevents small changes which might propagate to the segmented image. We finally crop the box contour so as to remove the border pixels that belong to the background.

The last procedure, as implemented in the color-based object segmentation, is to rotate the well-cropped image to a reference angular position. The previous method of maximum-gradient based rotation in Section 2.1.1.1 does not give satisfactory results because the segmented image is in a square with the background full of (theoretically) white pixels, which lead to too much noise in the histogram of angular values.

We had to resort to a solution using machine learning to solve this problem. First, the detected bounding rectangle is turned parallel to the images borders (ie. up, left, down or right), and sent to a classifier that has been trained to recognize the already seen objects in an arbitrary reference orientation. Then, depending on the classifier output:

1) if a new object is classified as one object from the training data with a very high confidence score, then this object is supposed to have the same shape and orientation with that of the targeted object and thus no rotation is needed;

[4]by applying OpenCV's findContour function

2) if it is classified as one existing object with a relatively high confidence score, but below than that in the previous case, it is thought to has the same shape but a different orientation, thus rotation takes place (ie. by anti-clockwise 90°, 180°, 270° respectively) and the new object will stay at the rotated angular position where the highest confidence score is attained;
3) if it is classified with a confidence score lower a threshold for consecutive frames, then this object is regarded as a novel object with regards to classifier's training database. Therefore, the classifier is trained with data concerning this new object before it becomes capable of dealing with new data sharing the same shape of this novel object.

The image descriptor used here is the HOG (which has been introduced in Section 1.3.2 and will be detailed in Section 2.1.1.2) yet with different parameter settings[5] than those for later visual features descriptions.

As for the classifier, we needed a classifier with small training time, possibility of partial fit (for the purpose of online learning) and reliability of the output confidence scores. We finally chose a linear model trained with Stochastic Gradient Descent (SGD) [6] while Support Vector Machine[7] (quadratic with the number of samples in time complexity and compulsory to relearn the whole database when new samples come) and K-nearest neighbors vote[8] (which requires high number of voting neighbors and outputs unreasonable confidence scores) have been tried but failed.

In short, the proposed rotation method guarantees that the same object is always shown in the same orientation, given that it has been trained when it was first seen.

2.1.1.2 Visual features

Shape

In our work we first used a very simple shape descriptor whose goal is to encode the identity of the objects, assuming that the different objects have stable and clearly distinguishable shapes.

The shape is therefore described by an array of normalized intensity values of pixels from a grayscale converted image. In allowance for the trade-off between lowering the computational burden and the effectiveness of presenting a shape, all images of recognized objects are resized to be 30×30 pixels, which avoids the problem of recognizing the same object category caused by different sizes. Therefore, the final result of shape presentation is a histogram by raveling

[5] the captured RGB image is resized to 128×128, orientation bins = 6, block size = 64×64 and block stride = cell size = 32×32

[6] http://scikit-learn.org/stable/modules/sgd.html#sgd

[7] http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

[8] http://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html

the array of pixel values so as to constitute a large vector of 900 elements, as shown in Figure 2.7.

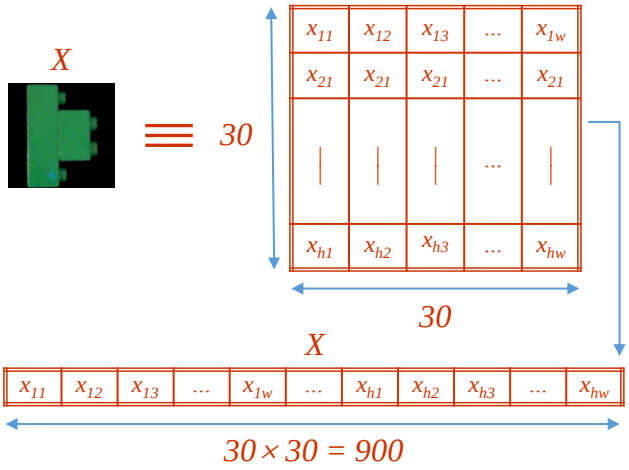


Figure 2.7: The histogram presentation in terms of the object's shape from an image.

Color

In our experiments, in order to lower as possible the variance of color values due to changing illumination, we controlled the lighting condition and we used the HSV (Hue Saturation Value) color model in order to represent object's color.

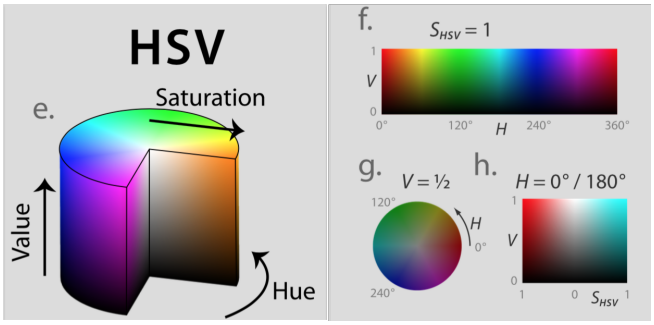


Figure 2.8: The graphical model of HSV.

The model of HSV is graphically shown in Figure 2.8. Unlike the cartesian representation of colors adopted by RGB, HSV uses a cylindrical description where the vertical axis in the center characterizes V (value, also known as brightness) and H (hue, also known as color) and S (saturation) are represented in a cross-section area by the angle and the distance to the vertical axis respectively. In OpenCV[9], hue range is [0, 179], saturation range is [0, 255] and value range is [0, 255]. For example, the minimum (ie. 0) or the maximum(ie. 255) of V means total black or white. In our experiment, only the H information is used to form a

[9]http://docs.opencv.org/3.0-beta/doc/py_tutorials/py_imgproc/py_colorspaces/py_colorspaces.html

color histogram of length 80 that represents the number of pixels of the image for each color, whereas both purely white and black pixels are excluded by setting a threshold where the saturation value - S is set to be zero.

A Gaussian blur is then applied to smoothen this histogram, taking into account the fact that this representation is circular (both 360° and 0° in H means "red").

HOG

We used the Histogram of Oriented Gradient (HOG, see description in Section 1.3.2 and Figure 2.9) in order to replace the shape description presented above in more realistic scenarios. In practice, the HOG function which we adopt for the acquisition of visual features is imported from the built-in toolkit of Scikit-Learn[10] and the parameter settings (units in pixel) in the experiments are listed as follows so as to reach a trade-off between the data precision and complexity:

$$\begin{aligned} \text{original image size} &= 80 \times 80 \\ \text{block size} &= 8 \times 8 \\ \text{block stride} &= 0 \\ \text{cell size} &= 8 \times 8 \\ \text{number of orientations} &= 9 \end{aligned}$$

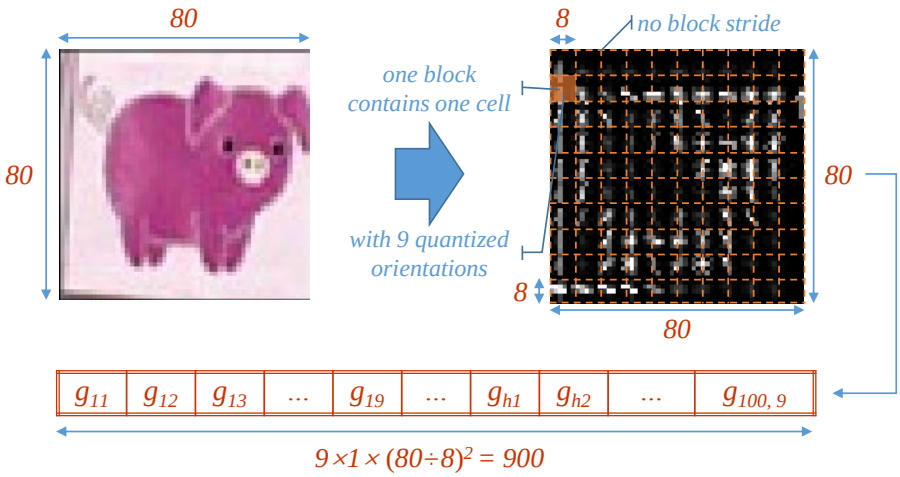


Figure 2.9: HOG parameter settings in the experiments.

In our setting, the RGB image is resized to 80 by 80. Given this small size, one block was set to contain only one cell with the size of 8 by 8 meanwhile no block stride (ie. the shift of a block that results in some pixels overlapping between two adjacent blocks) was used. Finally, the original image is described by $(80 \div 8)^2 = 100$ blocks, each of which contains 1 cell, and there are by default 9 gradient orientation bins for the histogram in each cell, therefore the

[10] http://scikit-learn.org/

HOG information can be raveled to a histogram presentation of 900 elements, whose length is equal to that of shape descriptor using pixels, which was found a good compromise between data precision and the computational burden.

2.1.2 Audio input

The solution of perceiving audio information in terms of speech is based on the voice-to-text conversion technique as described in Chapter 1.3.1.2.

The vocal sentences uttered by the human tutor are sampled at 16 kHZ by a microphone to get audio frames for the vocal-word conversion. The Google speech-api[11] is applied, which accepts audio flac files and returns a set of potential sentences in json format, where linguistic words can be extracted.

Two distinct sentences recording policies are used, differentiating subsequent experimental scenarios as "keywords only" and "full sentence". For instance, a vocal description of a red car has been converted by Google speech-api as "C'est une voiture rouge (this is a red car)", then following the policy of "full sentence", a histogram is created with the length of 4, whose indices correspond to the words - "C'est", "une", "voiture", "rouge" respectively; however, in the "keywords only" scenario, only keywords concerning nouns and adjectives are retained, that is to say, a histogram is created only for the two keywords "voiture" and "rouge".

We should also consider the problem about how this histogram recording adapts itself to the situation when new words appear in the upcoming new sentences. For this, a dictionary is defined starting from empty at the beginning to contain all known words, and a histogram just represents the occurrences of every known words in a sentence for the description of an object. If new words appear in next descriptions, the dictionary augments incrementally by adding each new word encountered in sentences at the end meanwhile the histogram is also prolonged by the length equivalent to the number of new words in order to present their occurrences in the sentence. For instance, in the "full sentence" scenario, if there comes the second sentence like "Voici une tasse verte (Here is a green cup)", then updated dictionary would be $\{c'est, une, voiture, rouge, voici, tasse, vert\}$ and the histogram regarding this new sentence would be $[0, 1, 0, 0, 1, 1, 1]$.

2.1.3 Feature quantification

In some of our approaches, the perceived data can not be used in their original formats of feature presentations, but need to be clustered. As an effective tool, vector quantification (VQ) will be applied in this thesis for the subsequent *statistical word filtering* (see Section 2.2) and *LDA learning model* (see Section 2.4).

It is the non symbolic (visual) channel in the observation vectors in V that needs to be

[11] `https://github.com/gillesdemey/google-speech-v2`

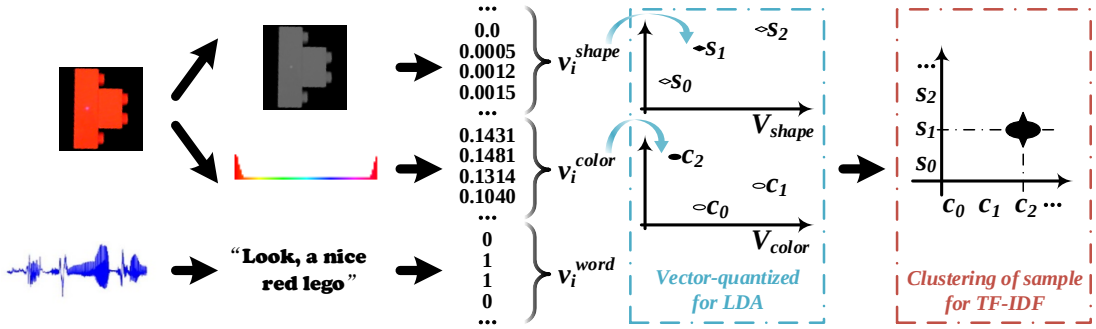


Figure 2.10: Diagram of vector quantification and clustering. Note that the shape description using pixel could be replaced by HOG (see Section 2.1.1.2) if we adopt HOG as the shape descriptor.

quantized. The clustering is performed by a simple incremental clustering that puts each observation in the same cluster as a previous observation if its distance is smaller than a threshold (whose value will be specified in related chapters of experiments), or creates a new cluster otherwise. We use the χ^2 distance which is well adapted for histogram features :

$$\chi^2(x,y) = \sum_{k=1}^{d} (x_k - y_k)^2/(x_k + y_k)$$

Each of the resulting shape cluster will be labelled as $s_t \in S$, while all member vectors within a cluster will be averaged as $v_{s_t} \in V_S$, then S and V_S act as entries and corresponding contents of the shape dictionary. The same procedure takes place for the formation of the HOG dictionary $\{H : V_H\}$ and the color dictionary $\{C : V_C\}$ as shown in Figure 2.10. A corpus (D) of vector-quantized samples d_i, $(i = 1, 2, ..., n)$ is then established by finding the items $s_i \in S$ or $h_i \in H$ together with $c_i \in C$ whose member vectors are most similar to V_i^{shape} or V_i^{HOG} and V_i^{color} respectively by applying χ^2 distance. Using the words $\mathbf{w_i}$ whose corresponding indices in V_i^{word} are positive, d_i indicates a collection of symbols, containing all words in $\mathbf{w_i}$ plus s_i or h_i as well as c_i (which we denote as VQ symbols). To facilitate the narrative, $W = \{w_1, ..., w_i, ..., w_n\}$ is denoted as a set for all recognized spoken words during the whole experiment.

The far right of Figure 2.10 illustrates a further clustering process where samples with the same VQ symbol pairs, either $\{s_i, c_i\}$ or $\{h_i, c_i\}$, fall into the same group. This serves as the pre-processing work for the proposed model of statistical word filtering, to be described in Section 2.2.

Note that the advantage of the proposed incremental clustering based VQ is that there is no need to pre-determine the number of cluster centers before hand, thus facilitating the case of incremental (and even online) learning; however, one limitation resides in the fact that this method is affected by the order of samples, therefore in some of the subsequent experiments, we just do the feature quantification for all the data at the beginning before the learning starts in order to have reproducible results independent of clustering variations.

2.1.4 Multiple objects representation

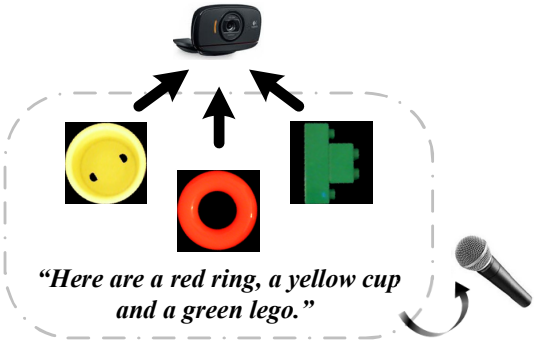


Figure 2.11: Example of an ambiguous teaching situation.

As described in Section 1.5.1, learning algorithms should deal with ambiguities regarding both the referent and the language (see Figure 2.11). These ambiguities therefore have to be represented in the data which, in subsequent experiments, occur especially in the case where multiple objects are presented.

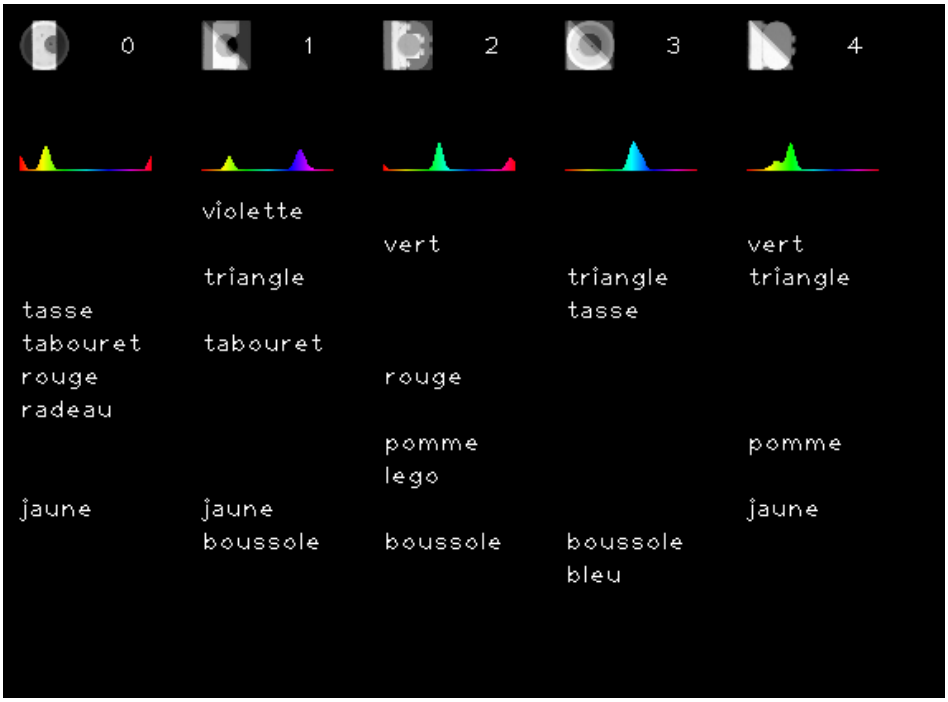


Figure 2.12: Examples of data recording regarding ambiguities of multi-object referring and full sentence descriptions.

In practice, thanks to the fact that all features are presented as histograms, multiple objects of interest can simply be represented by summing the description of each individual object. An example of several input sample recordings in cases similar to Figure 2.11 can be found in Figure 2.12, benefiting from the additive property in terms of histogram presentation.

In order to make our experiments representative enough of both the referential and the linguistic ambiguities, we define two different cases of linguistics ambiguities and three dif-

ferent cases of referential ambiguities that lead to six different learning scenarios as shown in Table 2.1.

Table 2.1: Data ambiguities defined in different scenarios and cases.

SCENARIO \ CASE	**Single**	**Double**	**Triple**
Keywords only	KW&S	KW&D	KW&T
Full sentence	FS&S	FS&D	FS&T

In this table "**Keywords only**" indicates the scenario where a human tutor only speaks feature-related words of objects (ie. nouns and adjectives) in contrast to the scenario of "**Full sentence**" in which the speaker uses natural sentences (including articles, pronouns, verbs, etc...) to describe an observed scene, and **Single**, **Double** and **Triple** refer to how many objects (one, two and three respectively) the teacher would present to a learner robot. In the remaining of the thesis, notions of KW&S, KW&D, KW&T, FS&S, FS&D, FS&T will be used for short, referring specific experiment settings of different level of ambiguity in Table 2.1.

2.2 Statistical word filtering

In order to tackle the linguistic ambiguity, this thesis adopts statistical models as a solution. Since LDA is just a statistics-based generative model, it is supposed to be able to deal with linguistic ambiguity in the scenario of "full sentence". However, for NMF, which is matrix operation based, an initial filtering of keywords has proved to be required to obtain good performances.

We applied the Term Frequency-Inverse Document Frequency (TF-IDF) approach [155], which is popular in text processing, to build a statistical word filtering model tailored as a pre-processing for NMF.

First of all, by applying the clustering method described in Section 2.1.3, we group all the observed samples so as to put together all the observations that share a common (non symbolic) feature for the analysis of the associated word statistics. Then for every such cluster, all recognized words used for the descriptions of samples in the cluster are grouped to form a *document* (the term expressed in the original theory) corresponding to this cluster, and the total set of words describing samples from all groups thus becomes the so called *corpus*.

The *term-frequency* (TF) of the word i associated with each cluster j is then computed :

$$tf_{ij} = n_{ij}/n_j$$

where n_j is the total number of words observed in samples from cluster j, and n_{ij} is the number of occurrence of word i observed in samples from cluster j. This value is high for

words that occur often with a given object. The words with tf below or equal a threshold are considered as noise and removed from the observations (their entries are put to 0).

The *inverse document frequency* (IDF) of the remaining words i is then computed:

$$idf_i = log[N/(1 + N_i)]$$

where N is the number of clusters, and N_i is the number of clusters where word i appears at least once. This measure is high for words that appear in very few clusters and low when they appear in many clusters. The words with *idf* above a threshold are common words (such as articles) and removed from the observations. The words with *idf* below a threshold are also removed, making the assumption that each word will eventually be associated to several different objects (e.g. two objects will have the same color). We define thresholds on IDF in order to remove too common or too rare words as

$$\begin{aligned} idf_{low} &= idf_{min} + \eta_{low}(idf_{max} - idf_{min}) \\ idf_{high} &= idf_{min} + \eta_{high}(idf_{max} - idf_{min}) \end{aligned} \tag{2.1}$$

where idf_{min} and idf_{max} are the maximum and minimum of *idf* values for all words. For the subsequent experiments, η_{low} and η_{high} values are optimized to reach the highest possible final performance in each scenario.

Theoretically, when sample size of input data is sufficient, the remaining words after these two steps should contain little noise and represent the words for which we have enough cross-situational data to learn their meaning, however, due to the limited number of objects we used in the experiments, the thresholds concerning TF and IDF can not be set with fixed values and our choice of thresholds will be further described in related chapters on experiments. In addition, the total number of all selected keywords could suggest k, a parameter known as the number of NMF components which will be used in the next section (Section 2.3).

2.3 Learning with NMF

Taking inspiration from [123], we applied NMF for word-referent learning using the previously described data. Regarding the mathematical notations, because every object is described in terms of shape and color and the shape feature could be represented by using pixels or HOG, we take by default that if *shape* as subscript appears in the remaining of this section, it refers to modalities of either shape pixel or HOG, so as to facilitate the following description.

The NMF decomposition in our experiments could be formulated as

$$\begin{aligned} V_{m\times n} &= W_{m\times k} H_{k\times n} \\ \begin{bmatrix} V_{shape} \\ V_{color} \\ V_{word} \end{bmatrix}_{m\times n} &= \begin{bmatrix} W_{shape} \\ W_{color} \\ W_{word} \end{bmatrix}_{m\times k} [H_1, H_2, ..., H_n]_{k\times n} \end{aligned} \tag{2.2}$$

where V denotes the matrix of data, w the learned dictionary of topics and H the coefficients to reconstruct the data from the topics.

Additionally, let $W_i (i \in 1, 2, ..., k)$ be denoted as a column vector of W, then

$$W_i = \begin{bmatrix} W_{i-shape} \\ W_{i-color} \\ W_{i-word} \end{bmatrix}$$

where $W_{i-shape}$ and $W_{i-color}$ represent the visual information elements while W_{i-word} represents the associated words.

The main adaptations of the NMF algorithm for the applications in our experiments consist in two parts:

1) initialization of $W_{m \times k}$ and $H_{k \times n}$

2) normalization of the non symbolic channel of $W_{m \times k}$ during iterations.

With these adaptations, NMF is expected to produce results that are both stable and clear in interpretation as concepts, provided with a proper number of components k.

2.3.1 Initialization

We expect that the learned result from NMF processing could be regarded as a concept, which is in the form of "one symbolic label - one feature pattern" similar to the "one entry - one explanation" format in a dictionary. With this structure, it would be easy to construct compound or complex meanings through linear combination. To favor that we use the following initialization:

For every column vector in W, we first randomly initialize and normalize the non symbolic channel, ie $\begin{bmatrix} W_{shape} \\ W_{color} \end{bmatrix}$. Then, for the symbolic channel whose size is $k \times k$ since only k keywords are present or k words are retained from the module of statistical word filtering when the number of components is k, we initialize it to the identity matrix where only one element is made 1 while the others are all 0 in a column.

Hence we can get the initialization form of W as

$$\begin{bmatrix} \cdots & W_{i-shape+color}^{normalized} & \cdots \\ - & - & - \\ & 1 \quad \cdots \quad 0 & \\ & \vdots \quad \ddots \quad \vdots & \\ & 0 \quad \cdots \quad 1 & \end{bmatrix}_{m \times k}$$

To note that it is not strict that the symbolic channel part should be an identity matrix, if every column as a binary vector has only one element as value 1, whose index position in each column ought to be different. But for purpose of facilitating the mathematical expression and operation as well as coding, the term - "identity matrix" is preferred without losing any generality for this problem.

When W is initialized, the initialization of H matrix is processed as

$$H^{init} = W^T V \tag{2.3}$$

From the mathematical point of view, since NMF always converges to a local optimum, the proposed initialization design tries to help NMF algorithm to start iterations from a nearby region of a local optimum which corresponds to a desired format of expressing a learned concept.

2.3.2 Normalization and convergence criteria

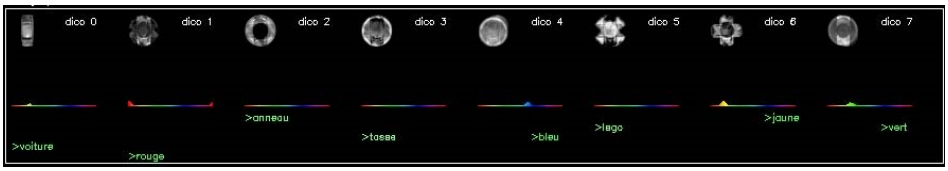
The normalization operation is not only used in the initialization, but also in every iteration when a new basis matrix $W^{(p)}$ (where p is number of iterations) is formed.

Therefore, since $W_i^{(p)} (i \in 1, ..., k)$ is composed of $W_{i-shape}^{(p)}$, $W_{i-color}^{(p)}$ and $W_{i-word}^{(p)}$, $\begin{bmatrix} W_{i-shape}^{(p)} \\ W_{i-color}^{(p)} \end{bmatrix}$ and $W_{i-word}^{(p)}$ are normalized respectively, before the next iteration for $W^{(p+1)}$ until the stopping condition is met.

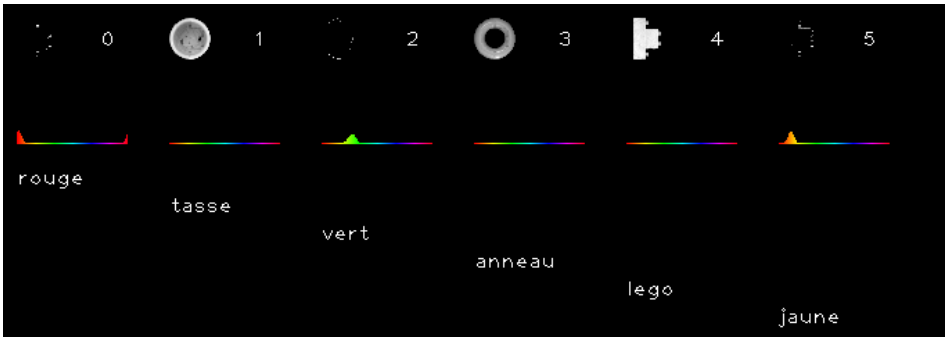
The idea underlying this normalization process is to enforce some structure in W_i in order to have as much as possible one word associated to its definition in one of the feature spaces. The evolution of a dictionary entry W_i during NMF iterations would therefore change its content but not its structure (of a symbol plus a real word meaning).

Although the effectiveness of our proposed model will be detailed later, an obvious advantage of our proposed NMF models is illustrated in Figure 2.13, where two examples of learned results by applying two NMF models, the original version proposed by Mangin [123, 124, 125] and the version proposed in this thesis, dealing with the KW&S data (every object is presented once) as input. We can observe that in the learned result by the original NMF model, a symbol concerning a shape or a color could be associated with a mixed visual presentation of both sub-modalities, which indicates a degree of confusion in understanding a concept; meanwhile every component in the result from the proposed NMF model is much more clearer, up to the definition of a simple concept.

As for the convergence rule of the proposed NMF model, the KL-divergence is adopted as the optimization criteria, and we take $e^{(i)}$ as the error in terms of KL-divergence after the i_{th} iteration. Therefore, the condition on which the iterative calculation would stop is either $\left| e^{(i+1)} - e^{(i)} \right|$ should be under a defined threshold or p reaches the max iteration number.



(a) An example of the learned result by the original NMF model.



(b) An example of the learned result by the our proposed NMF model.

Figure 2.13: Comparison between the learned results by using different NMF models.

One last thing about the coding of NMF algorithm is that an offset number (10^{-8}) is added to some variables who might be the denominator in the update rule. Obviously, this avoids pure zero number when applying division operation.

2.3.3 Incremental learning

We applied NMF in several different scenarios. In the batch learning case (see Chapter 3), NMF deals with a total of all collected samples V observed by the robot and described by a human teacher, and produce a topic matrix W used for subsequent evaluation.

In the incremental learning case (see Chapter 4), whenever a new observed sample is described and perceived, it is added as the last column in parallel with all previous samples V_{t-1} to create an up to date matrix V_t as input. It is then processed by the same NMF method as for batch learning to produce the current learning state in the form of the matrix W_t. While dedicated incremental learning algorithms could be used (avoiding to process the full V_t matrix), we choose to retrain the models using all the data of the updated matrix V_t, so as to give an upper bound of the performance an incremental learning algorithm could achieve.

2.4 Learning with LDA

In parallel with the NMF based topic modeling of the cross situational word-meaning association, a probabilistic topic model is build using LDA. While no major modification is proposed to change the theoretical structure of LDA, this thesis tries to let the experimental

data be fitted to the learning model of LDA, which processes the same source of input data and undergoes the same kind of tests for comparison with NMF (mainly described in Chapter 4).

By using the Vector Quantified symbols, i.e., the results of the feature quantification of input raw data (as stated in Section 2.1.3), we have a corpus of observed data $D = \{d_1, ..., d_i, ..., d_n\}$ where d_i is the VQ representation of the i_{th} input (which might be of single, double or triple data) concerning *shape/HOG*, *color* and *word* label, and n denotes the total number of inputs for the current LDA training. As for every d_i, in the **Single** object case, $d_i = \{\mathbf{w_i}, s_i, c_i\}$ or $d_i = \{\mathbf{w_i}, h_i, c_i\}$ according to whether we use pixel or HOG as shape descriptor during experiments. However in **Double** and **Triple** cases, d_i would consist of more than one $s_i \in S$ (or $h_i \in H$) and $c_i \in C$. As for $\mathbf{w_i}$, regarding the modality of word label, it contains all the recognized words for the description of the i_{th} input if it is for "**Full sentence**" or only keywords of shape noun(s) and color adjective(s) in the case of "**Keywords only**".

Every topic $k \in \{1, ..., K\}$, in our experiment settings, is expected to only contain (ideally) a keyword label from $\mathbf{w_i}$ paired with a definition in s_i, h_i or c_i. The parameter ϕ_k (see the LDA diagram in Figure 1.19) is thus assumed to be a sparse word distribution per topic over all words of W, S or H and C. Also note that the number of VQ symbols in every d_i might not be the same, which differs from the case of N in the innermost box in Figure 1.19.

The parameter inference of proposed LDA learning model will maximize the $p(\mathbf{w}, \mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\phi}; \alpha, \beta)$ in Equation 1.36 by using Collapsed Gibbs Sampling[12] which is presented in Chapter A.2. In practice, we observe that for a given k, the distribution $p(., z_k, \beta)$ is only significant for a couple (s_i, w_i), (h_i, w_i) or (c_i, w_i) where $w_i \in \mathbf{w_i}$.

2.5 Active learning

In the previous chapter, following [138], we categorize active learning strategy into *Error Maximization based* and *Progress Maximization based*. The original methods proposed in [138] require that the learning agent be capable of testing its performances on the task to learn (here the word-referent association) so as to estimate its learning progress. This, however, does not fit in our experimental scenario that simulates the natural parent-infant interactive learning where the infant just keeps on observing objects with descriptions without taking tests (like school exams) each time. In this scenario, the parents would only occasionally make conversations to test child's knowledge about a certain object.

The testing measures in all experiments in this thesis therefore only play a role of "third-party" evaluation which is inaccessible for both the parent and the child. There should therefore be another, indirect, criteria to decide if a training sample is useful or not. We will see that the reconstruction error in NMF and the likelihood of data in LDA can play

[12] We use the implementation from `https://github.com/ariddell/lda` with all parameters initialized with default settings

this role. The active learning strategies proposed below belong to the Error Maximization category. Note that we performed experiments with Progress Maximization approaches using the same criterion but were not able to obtain satisfactory results. We believe this is due to the limited size of the dataset used (compared to classical experiments in the field) that did not made it possible to evaluate a correct learning progress estimate.

Many active learning algorithms exist in the literature, but a lot of them are either unsuitable to the experimental scenario of the thesis or not showing superiority over random learning performance. For instance, in applications of active learning to the Naming Games [164] where the goal is towards a global agreement of the word-meaning associations among a group of agents, teacher and learner exchange roles and are initialized each with random word-meaning associations. In this context, it is possible to use *Decision Vector* strategies for the teacher [164] that depends on the current vocabulary size to decide when to explore new meanings because the vocabulary size can augment when an agent becomes a learner. In our scenario, where the agent is always a learner, such strategies will get stuck into situations where only a limited part of the vocabulary is learned because the size of the vocabulary where new meanings can be explored could never be reached. We will see however that a variant of the *Success-Threshold Strategy* and *Last Result Strategy* [164] can be applied in our experiments.

2.5.1 Maximum reconstruction error based selection (MRES)

The maximum reconstruction error based selection (MRES) holds the belief that the sample which is the worst reconstructed is the best indicator of the current deficiency of the learned knowledge, therefore by further introducing such samples for training, the learner gets more raw data concerning the deficient components to process and the deficiency residing in the current knowledge structure is supposed to be compensated.

As for the internal evaluation of the acquired knowledge, using the notations in Section 1.4.2.3 and 1.4.3.2, we use the generalized KL divergence for NMF:

$$D_{KL}(V_{\cdot j} \| WH\ ) = \sum_i (V_{ij} \ln \frac{V_{ij}}{(WH_i)_j} - V_{ij} + (WH_i)_j)$$

and the likelihood function for LDA:

$$L_{LDA}(d) = p(\theta_d; \alpha) \prod_{k=1}^{K} p(\phi_k; \beta) \prod_{n=1}^{N_d} p(z_{d,n} | \theta_d,) p(w_{d,n} | \phi_k, z_{d,n})$$

as the reconstruction error measures, where $V_{\cdot j}$ represents the j_{th} sample in the training database for NMF and d indicates a single document for LDA.

The active selection of new samples is then formulated as finding the new test sample

such that:

$$\begin{aligned} \hat{j} &= \arg\max_{j}(D_{KL}(V_{\cdot j} \left\| WH^{(P2)} \right.)/Gini(H)) \\ \hat{d} &= \arg\min_{d}(L_{LDA}(d)) \end{aligned} \tag{2.4}$$

where $j \in \{1, 2, ..., n\}$, $d \in \{1, 2, ..., D\}$ are the new samples that can be considered for learning. $H^{(P2)}$ are the weight coefficients derived from H [13] by making use only of the two highest values while discarding all others and updated via the reconstruction by minimizing $D_{KL}(V_{\cdot j} \left\| WH^{(P2)} \right.)$. $Gini(H)$ is the Gini index which estimates the sparsity of the H vector (to be detailed in Equation 3.4 in Chapter 3). These two modifications have the objective of promoting the samples that need more than 2 dictionary components to be correctly reconstructed, based on the idea that these samples are not currently well known. In fact, we tried originally $\hat{j} = \arg\max_{j}(D_{KL}(V_{\cdot j} \| WH))$ directly derived from the NMF algorithm, however, the version in Equation 2.4 would produce prominently better results in practice.

As previously described, we use **single**, **double** and **triple** object sets to represent different level of referential ambiguities. In these cases, the learner could choose 1, 2 or 3 samples accordingly by applying Equation 2.4. However, in practice, we found problems when the learner strictly executes this strategy, for instance, if certain sample(s) which are chosen do not efficiently improve the current knowledge status, this/these sample(s) will be selected over and over again, resulting in a stagnation of the performance.

Here, a slack strategy (which has also been referred as drop-out strategy in some previous literature) is proposed to help the learning agent jump out of the described "dead loop". First, a pool of selected candidate samples (more than necessary) is created using the previous strategy: we choose the 6, 9 or 12 samples of maximum reconstruction error values for **single**, **double** and **triple** cases respectively. Then the learner will randomly choose from this pool the needed (ie. 1, 2 or 3) sample(s) for the next incremental training.

2.5.2 Confidence base exploration (CBE)

The second tested strategy is derived from *Success-Threshold Strategy* and *Last Result Strategy* in [164]. The general idea is that a learner seeks to explore unknown new things only when confident enough about what has been learned already, therefore we should first take measure to define such a confidence. For this, we have the knowledge of all the previously learned samples, and every sample has its associated reconstruction error. It is natural to define a self-evaluation criteria by calculating the average reconstruction error of all used samples as the confidence indicator, and if this value is below a certain threshold, the learner is thought to be confident about the knowledge that has been learned.

We define $D_{KL}^{n'}$ and $L_{LDA}^{D'(d)}$ respectively as the confidence indicators for NMF and LDA by

[13] computed by finding the H that minimize $D_{KL}(V_{\cdot j} \| WH)$ first

utilizing the average of D_{KL} and L_{LDA} values of all already used samples:

$$\begin{array}{l} D_{KL}^{n'} = \sum_{j=1}^{n'} [D_{KL}(V_{\cdot j} \| WH\ )]/n' \\ L_{LDA}^{D'} = \sum_{d=1}^{D'} [L_{LDA}(d)]/D' \end{array} \tag{2.5}$$

where n' and D' represent number of used samples in NMF and LDA learning.

When feeling confident, in other words $D_{KL}^{n'} \leq threshold_{D_{KL}}$ or $L_{LDA}^{D'} \geq threshold_{L_{LDA}}$, the learner will explore by randomly choosing candidate objects among those that does not contain the features (relating color and shape) which have already existed in any used samples. However, when the confidence is not sufficient, that is to say $D_{KL}^{n'} > threshold_{D_{KL}}$ or $L_{LDA}^{D'} < threshold_{L_{LDA}}$, sample selection will favor already encountered color and shapes: we choose the worst reconstructed sample(s) by using the previous MRES method[14] from the pool of samples that have only one feature in common with the already known samples.

Note that as the incremental learning proceeds, there are less and less unknown features up to the point where all features have been seen at least once, while there still remain unused samples. In that case, the exploration have no samples to choose from, so we resort to random choice among all unused samples instead.

[14] without using the slack strategy

Chapter 3

Learning with NMF and automatic estimation of the NMF dictionary size

Contents

In this chapter, we present experiments on a small object dataset that focuses on the use of NMF, whose structure has been modified in Chapter 2.3, with the goal of setting algorithms parameters and evaluating the quality of the learned dictionary of word-meaning associations. We are primarily concerned with setting the value of k, the size of the NMF dictionary, which has to be chosen manually. Moreover, we will take advantage of the interpretability of the NMF result thanks to the non-negativity of the dictionary to explicitly compare the resulting dictionary with a reference dictionary that can be easily constructed for such a small dataset.

Our first exploration towards the determination of k was by making use of a reference dictionary for the similarity measure with a group of learned dictionaries given a range of candidate values for k. We verified that the value of k corresponding to the real number of words to learn indeed lead to a dictionary very close to the reference dictionary. In parallel, SV-NMF [58], as a representative of the current studies on the topic of auto-determination of k, was evaluated to see if it is applicable to our experimental scenarios by comparing its result to reference dictionaries. However, as this method does not prove to be effective, we proposed a quality measure of the learned dictionary that does not require a reference dictionary and apply it with the standard NMF algorithm to automatically find the value of k. We then evaluated its effectiveness by comparing the learned dictionary with its reference.

3.1 Experimental data

The interactive learning procedures take place on a table, which is decorated in black as the background. The set of objects for learning is composed of *lego toys*. It is denoted as *S-OBJ-A* so as to differentiate it from *S-OBJ-B*, *S-OBJ-C*, *S-OBJ-D* used in the next chapter. Note that in order not to make the experiments too complicated, yet without loosing generality, we deliberately ignore the feminine as well as masculine forms of adjective keywords in French by simply adopting either its masculine (eg. *bleu*) or feminine (eg. *violette*) if necessary.

3.1.0.1 S-OBJ-A

There are four types of objects in four colors (therefore 8 concepts relating to shape and color) as objects to be learned: "ring (*anneau* in French)", "lego","cup (*tasse*)" and "car (*voiture*)" in "red (*rouge*)", "yellow (*jaune*)", "green (*vert*)" and "blue (*bleu*)" respectively, as shown in Figure 3.1. Every object is described by a human participant using normal descriptive sentences including the keywords concerning the shape and color like "*C'est une voiture rouge* (This is a red car)", "*Voici un lego bleu* (Here is a blue lego)" according to his/her own will. The training data base contains 50 input samples, with every object in different color being described and recorded around 3 times. Yet for all the experiments in this chapter, where the statistical word filtering is not needed to achieve the goal of this chapter, all recorded data will be processed as "keywords only" data related to a single object's description (ie. *KW&S* in Table 2.1).

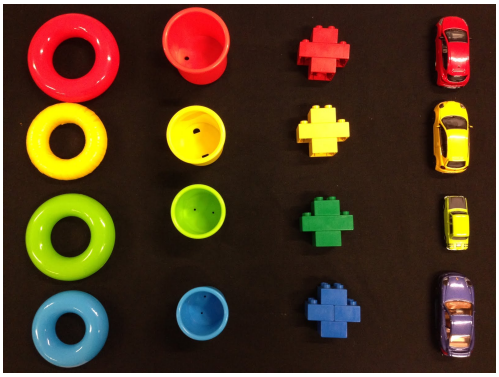


Figure 3.1: *S-OBJ-A*: Four types of objects in four colors.

To obtain the modal information of the object from *S-OBJ-A* by using the setup described in Chapter 2.1, a small camera is installed over the table at a proper height, perpendicularly facing down to capture the view of the table, then a microphone is settled at a proper distance from the human participant in order to receive the audio linguistic information. The raw dataflow of both images and voice acquired from the camera and the microphone will be transmitted to a computer via USB interface, where data files are created. The image descriptor used for these experiments are the shape histograms and the H histograms (see Section 2.1.1.2). Note that, because the *S-OBJ-A* data had been recorded before the rotation

method (introduced in Chapter 2.1.1.1) was fully developed, a few samples have different angular positions, which will bring some errors in the following experiments.

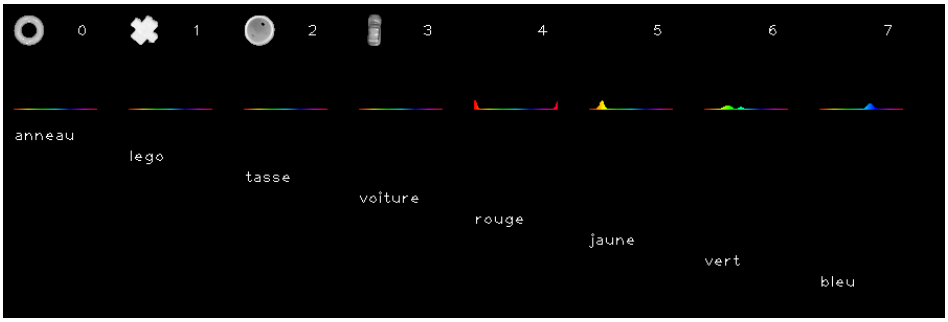


Figure 3.2: Visualization of the reference dictionary (Full dictionary).

In order to perform an systematic analysis, we prepared reduced experimental data sets besides the *Full dictionary* data (ie. four objects in four colors as described above, whose recorded data are visualized in Figure 3.2) and they are:

- *2c-2s*, with keywords of 2 colors and 2 shapes, consisting of "lego" and "tasse" in "rouge" and "bleu", as visualized in Figure 3.3;

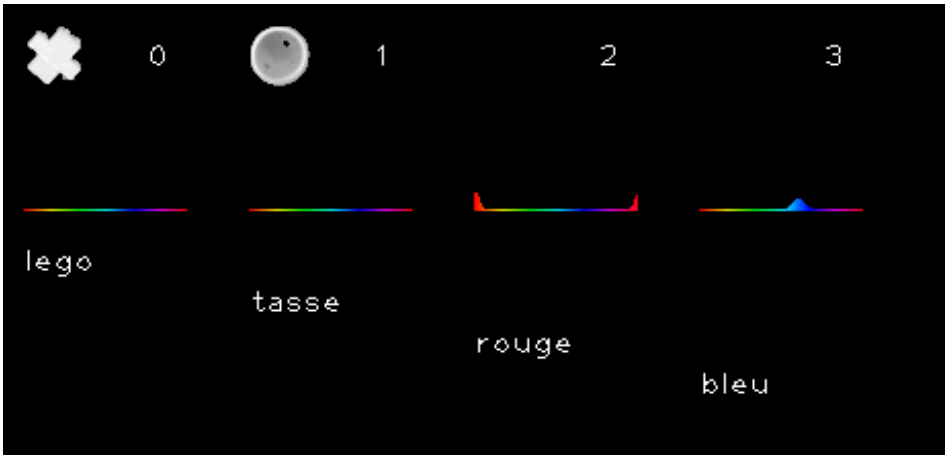


Figure 3.3: Visualization of the reference dictionary (2c-2s).

- *3c-3s*, with keywords of 3 colors and 3 shapes, consisting of "lego", "tasse" and "voiture" in "rouge", "jaune" and "vert", as visualized in Figure 3.4;

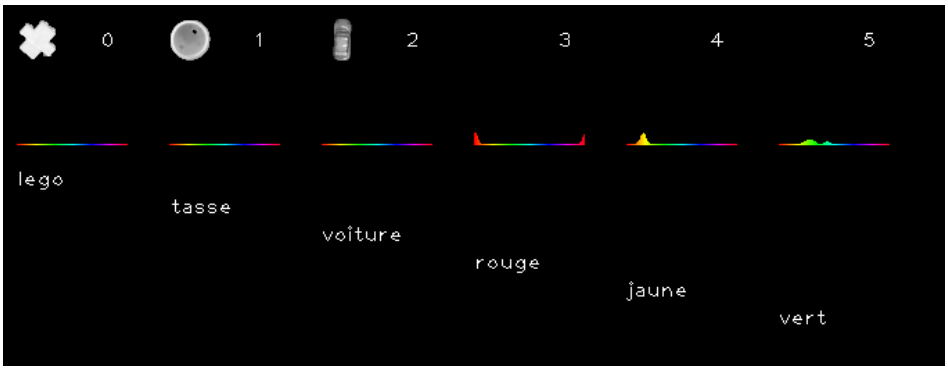


Figure 3.4: Visualization of the reference dictionary (3c-3s).

- *4c*, with keywords of four colors of “rouge”, “jaune”, “vert” and “bleu”, as visualized in Figure 3.5;

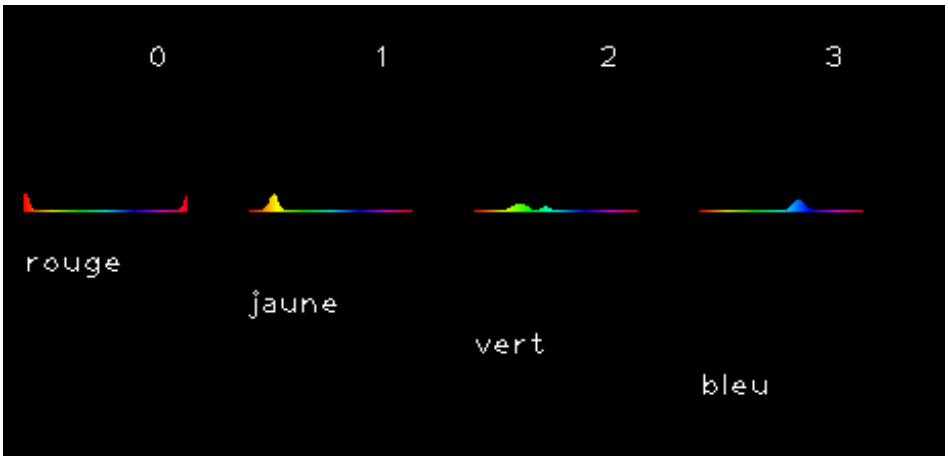


Figure 3.5: Visualization of the reference dictionary (4c).

- *4c-2s*, with keywords of 4 colors and 2 shapes, consisting of “lego” and “tasse” in “rouge”, “jaune”, “vert” and “bleu”, as visualized in Figure 3.6;

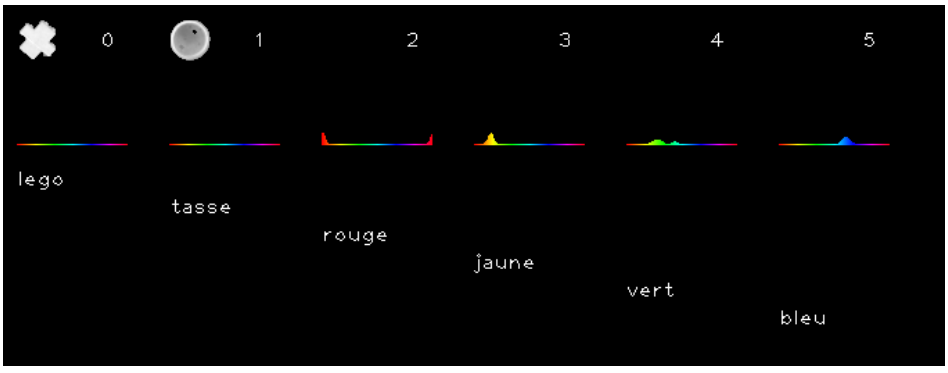


Figure 3.6: Visualization of the reference dictionary (4c-2s).

- *4s*, with keywords of shapes, consisting of “anneau”, “lego”, “tasse”, “voiture” with no color informations, as visualized in Figure 3.7;

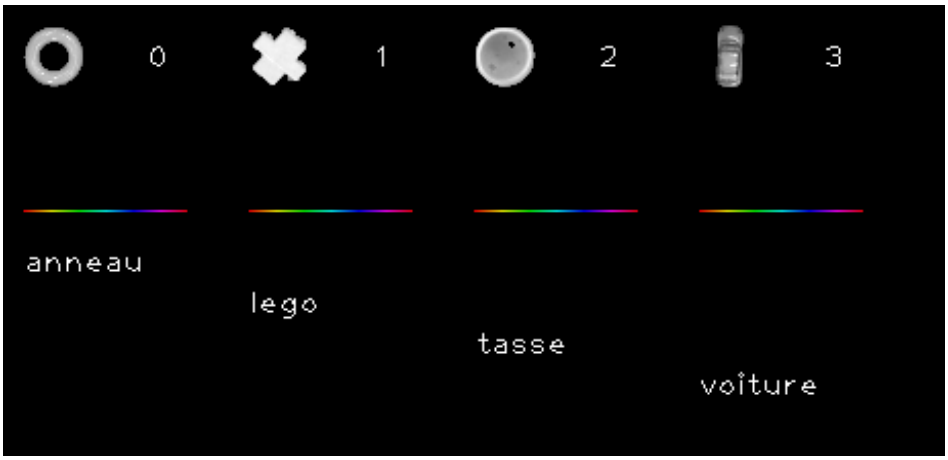


Figure 3.7: Visualization of the reference dictionary (4s).

- *4s-2c*, with keywords of 2 colors and 4 shapes, consisting of “anneau”, “lego”, “tasse”, “voiture” in “rouge” and “bleu”, as visualized in Figure 3.8.

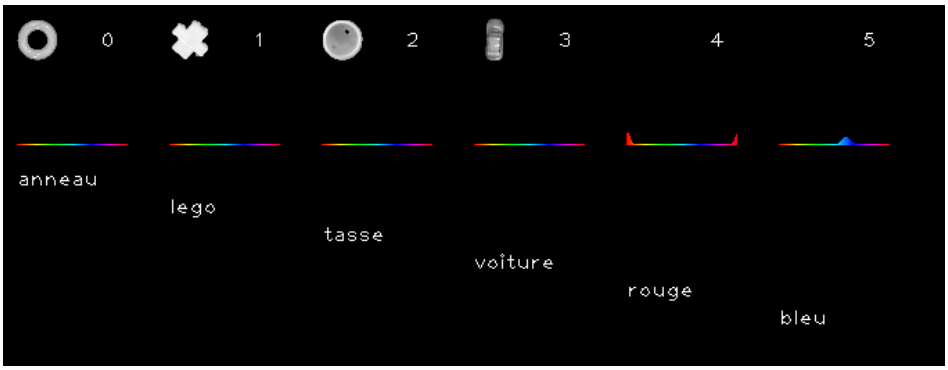


Figure 3.8: Visualization of the reference dictionary (4s-2c).

3.2 Measuring the quality of word-learning association

We consider two cases for evaluating the quality of a learned dictionary of concepts. In the first case, we assume that a reference dictionary is available and compare the result to this reference. This measure helps evaluate the performance of the algorithms and makes it possible to find the optimal parameters, but is not usable in real scenarios where the reference dictionary is not known in advance. In the second case, we introduce a quality measure that is based on the reconstruction error and the sparsity of the reconstruction and this measure can be applied in a realistic scenario.

3.2.1 Comparing with a reference dictionary

A common idea of evaluating a learned dictionary with regard to a reference dictionary is to make a pairwise comparison (eg. similarity measure) of items, which are from both

dictionaries and possibly directing to the same concept, and then make a summation of these comparison results as a measure of quality. The reference dictionary (as shown from Figure 3.2 to Figure 3.8) is formed by first averaging the histogram descriptions of all the valid samples concerning a keyword (of 4 shape concepts and 4 color concepts), then only keeping the part that is for the modality indicated by this keyword while replacing other positions in the histogram by zeros.

The above comparison idea can be carried out by using a matrix comparison, with the assumption that the order of entries are the same in both dictionaries and that every entry is unique. In this case, there exists one-to-one entry mappings between dictionaries, which can be formulated as

$$\begin{aligned} W^{\text{reference}} &\approx W^{\text{learn}} C \\ \left[W_1^{\text{reference}}, W_2^{\text{reference}}, ..., W_k^{\text{reference}}\right]_{m \times k} &\approx \left[W_1^{\text{learn}}, W_2^{\text{learn}}, ..., W_{k'}^{\text{learn}}\right]_{m \times k'} C \end{aligned} \tag{3.1}$$

where C is a matrix which will be the identity in the ideal case of the learned dictionary being equal to the reference dictionary. When $k = k'$, that is say the two dictionaries have the same entry numbers, C is considered to be an k times k eye matrix or resembles a diagonally dominant matrix; however, when $k \neq k'$, i.e. the learned dictionary is not perfect, C is expected to be an eye matrix, augmented with row or column vectors of all zero elements.

3.2.1.1 Conversion to a similarity measure of matrices

Thus, instead of using the item-wise similarity comparison between $W^{\text{reference}}$ and W^{learn}, we could find a more mathematical-friendly and convenient criterion to evaluate a learned dictionary by measuring the similarity of C with an identity matrix. Yet before coming to the algorithm of similarity measure, we should solve the problem of how to compute C, which boils down to the solution of linear system of equations.

Interestingly, we first notice that the solution of Equation 3.1 can be computed using a variant of NMF. The objective is formulated as finding C such that $\min_C arg(W^{\text{reference}} \left\| W^{\text{learn}} C\right)$, where $W^{\text{reference}}$ is the reference dictionary as ground truth and W^{learn} is kept fixed.

The resulting C matrix from Equation 3.1 could not at all be like an identity matrix, but thanks to the property of linear invariance we can exchange the row or column in C. This is simply because once we change two columns of C, say $C_{.i}$ and $C_{.j}$, if W_i^{learn} and W_j^{learn} change their positions correspondingly, then the linear relation expressed in Equation 3.1 remain the same. This property also fits the case of simultaneous row exchange of $C_{i.}$, $C_{j.}$ and the position change of $W_i^{\text{reference}}$, $W_j^{\text{reference}}$.

We therefore developed an algorithm (Algorithm 1) for reordering the matrix C so as to have it in the form of a dominant diagonal.

Algorithm 1 Reordering of matrix.

Input: $X_{m\times n}$
Output: $Y_{m\times n'}$
1: $m,\ n \leftarrow X_{m\times n}$
2: **if** $m > n$ **then**
3: $\quad Z_{m\times n'} \leftarrow (X, zero(m, m-n))$
4: **else**
5: $\quad Z_{m\times n'} \leftarrow X_{m\times n}$
6: **end if**
7: $m,\ n' \leftarrow Z_{m\times n'}$
8: **for** $i = 1 \rightarrow m$ **do**
9: $\quad z_{pq} \leftarrow max(Z_{m\times n'})$
10: $\quad exchange(Z_{p\cdot}, Z_{i\cdot})$
11: **end for**
12: **for** $i = 1 \rightarrow m$ **do**
13: $\quad z_{pq} \leftarrow max \begin{pmatrix} Z_{i\cdot} \\ \cdots \\ Z_{m\cdot} \end{pmatrix}$
14: $\quad exchange(Z_{p\cdot}, Z_{i\cdot})$
15: **end for**
16: **for** $i = 1 \rightarrow m$ **do**
17: $\quad z_{pq} \leftarrow \begin{pmatrix} z_{ii} & \cdots & z_{in'} \end{pmatrix}$
18: $\quad exchange(Z_{p\cdot}, Z_{i\cdot})$
19: **end for**
20: $Y_{m\times n'} \leftarrow Z_{m\times n'}$

The underlying reason for such a transform is based on the fact that every entry in a reference dictionary is unique and distinct from the others, that is to say, if one entry in a learned dictionary has already found its corresponding entry in the reference dictionary, it is unlikely that there exist another entry which might match it for the second time. By using the above algorithm, the largest elements in C will move to the diagonal position in descending order from C_{11} to C_{mm}. Noticeably, there exists the possibility in which some consecutive largest elements are in the same row or column, thus making it impossible to put them all in the diagonal positions. However, as explained above, if one diagonal element dominantly indicates the inter-dictionary relationship of two entries, the other elements from the same row or column should be regarded as noise and disturbances.

3.2.1.2 Similarity measure with an identity matrix

After the reordering of matrix C, we get C$'$. And the next step of dictionary evaluation is the comparison between C$'$ with an identity matrix of the same size. Among various similarity measures in machine learning, we here adopt two of them: *correlation coefficient* and *Euclidean distance (2-norm).*

Let's take I as an identity matrix. The Euclidean distance will be:

$$d = \|C'_{m\times n} - I_{m\times n}\|_2 = \left(\sum_{p=1}^{m}\sum_{q=1}^{n}\left(c'_{pq} - i_{pq}\right)^2\right)^{1/2} \tag{3.2}$$

and the correlation coefficient is

$$r = \frac{\sum_m\sum_n (C'_{mn} - \bar{C}')(I_{mn} - \bar{I})}{\sqrt{\left(\sum_m\sum_n (C'_{mn} - \bar{C}')^2\right)\left(\sum_m\sum_n (I_{mn} - \bar{I})^2\right)}} \tag{3.3}$$

where $\bar{C}'$ and $\bar{I}$ are the average values considering all matrix elements. Both Equation 3.2 and Equation 3.3 are available by calling functions *norm* and *corr2* in Matlab.

The Euclidean distance is an absolute measure of distance, thus the smaller d is (0 as the minimum), the more C' approaches I and the learned dictionary is of higher quality. As for correlation coefficient, it is more measuring the structure similarity, while disregarding its absolute scale, the closer r approaches 1, the better the result is.

Finally, cases should be considered when the number of components in the learned dictionary is not equal to that in the reference dictionary so as to make this evaluation framework complete. Let's denote k the number of components in the reference dictionary, k' for the learned dictionary and $Y_{m\times n}$ for the coefficient matrix by applying Algorithm 1.

When $k \neq k'$, a modification of the comparison score is needed. We have two cases:

1) $k > k'$: there are insufficient components created in the learned dictionary compared to the reference dictionary. That is to say, the learned dictionary only succeeds in creating $\frac{k'}{k}$ of the correct content. This why $\frac{n}{m}$ (ie. $\frac{k'}{k}$) is multiplied to *corr2*, whose maximum value is 1. As for *norm*, for the same reason, it should be added with the penalty representing the $(1 - \frac{n}{m})$ failure part. We assume that the maximum distance in terms of *2-norm* is n (ie. k) from $\|I_{n\times n} - \mathbf{0}_{n\times n}\|_2$, where $I_{n\times n}$ corresponds to the case when the learned dictionary exactly matches the reference dictionary and $\mathbf{0}_{n\times n}$ the result that none of the learned dictionary components could be used for the linear combination to construct the reference dictionary. Since a smaller value of *norm* index indicates a better performance, the penalty term $n \cdot (1 - \frac{m}{n})$ is added to $index_{norm}$;

2) $k' > k$: there are superfluous components in the learned dictionary compared to the reference dictionary. In this case we determine the percentage of the results that correspond to the correct components. In Algorithm 2, this percentage is $\frac{m}{n}$ (ie. $\frac{k}{k'}$) and modifications related to the penalty are exactly from the same concern as described in the previous case.

Algorithm 2 summarizes our method to estimate the quality of a learned dictionary.

Algorithm 2 Evaluation index of learned dictionaries.

Input: $Y_{m\times n}$
Output: $result_{norm}, result_{corr2}$
1: $m,\ n \leftarrow Y_{m\times n}$
2: **if** $m > n$ **then**
3: $I \leftarrow \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}_{n\times n}$
4: $Z \leftarrow Y(1:n,\ :)$
5: $index_{norm} \leftarrow norm(Z - I) + n \cdot (1 - \frac{n}{m})$
6: $index_{corr2} \leftarrow corr2(Z, I) \cdot \frac{n}{m}$
7: **else**
8: $I \leftarrow \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}_{m\times m}$
9: $Z \leftarrow Y(\ :, 1:m)$
10: $index_{norm} \leftarrow norm(Z - I) + n \cdot (1 - \frac{m}{n})$
11: $index_{corr2} \leftarrow corr2(Z, I) \cdot \frac{m}{n}$
12: **end if**

3.2.2 Quality estimation without reference dictionary

Considering more realistic scenarios, for example online learning where a robot continuously acquires new words, in which it is impossible to build a reference dictionary, we have to find alternative methods to evaluate the learned dictionary without reference. Here, two important features that are intrinsic to the learned dictionary, *reconstruction error* and *sparsity*, are adopted as the basis on which the quality estimation is built.

The reconstruction error is a measure of how much the decomposed matrices approximate the original one. For NMF, the KL divergence (see Equation 1.24) is used as the criterion to represent reconstruction error. An obvious property of reconstruction error is that if there are insufficient number of components in W, the reconstruction error would be huge; on the contrary, if the learned dictionary has the exact number or superfluous components, the error is expected to be quite small.

The sparsity is a measure of the distribution pattern of data, for example in a matrix, lager sparsity index means that non-negative values are located only on a few elements while most positions are filled with zeros or values very close to zeros, on the contrary, smaller sparsity index indicates the case where major values might appear evenly among most positions. During dictionary learning, the most goal dictionary is of the form “one label - one modality”, which is guaranteed by our modifications on NMF as described in Chapter 2.3 and appears sparser than other formats, for example the composite concept in which one label is associated

with multiple modalities. However, we have tested the sparsity measure of related histogram of every keyword, by using the data from the reference dictionary, and the result shows that the sparsity index is proved different for each keyword related histogram even within one modality (eg. *rouge*, *vert*) let along between two modalities (eg. *rouge*, *voiture*). To avoid the above complexity, the sparsity analysis is only applied on word label histograms. For this part, if a learned dictionary has less components than (or equal to) what it should be, the sparsity value should be at a comparatively high level due to the fact that every word modal histogram contains only one position with a non-zero value while others are all zeros; on the other hand, if the dictionary contains redundant components, the sparsity measure would be lower.

Among all the available algorithms to compute sparsity, the Gini index (see Equation 3.4)[1], is adopted to evaluate every sub-column vector relating to the words of a learned dictionary and the average value of these Gini indices is used as the overall sparsity measure. Assuming that a vector is reordered as monotonically increasing, denoted as $\mathbf{x} = [x_1, ..., x_i, ..., x_n]$, the Gini index is formulated as

$$Gini(\mathbf{x}) = \frac{n + 1 - 2\sum_{i=1}^{n}(n+1-i)x_i / \sum_{i=1}^{n} x_i}{n-1} \tag{3.4}$$

Finally, we come to a comprehensive indicator combing both the reconstruction error and the sparsity measure to evaluate the quality of a learned dictionary as

$$idx_{dictionary} = a \cdot error + b \cdot (1 - Gini) \tag{3.5}$$

where a and b are parameters that weights the two terms respectively. We used $a = 1/150$ and $b = 50$ so as to make the two criteria curves comparable.

3.3 Evaluation of NMF with reference dictionary

In this section, we rely on using a reference dictionary for the purpose of finding a recommended k to run the NMF learning model, and verify that with such a parameter setting the learned dictionary is close to the reference dictionary. For this, k as a parameter, will be endowed a value from a wide range (eg. all integer numbers from 2 to 14), and we will record the quality of the learned dictionary in terms of criteria regarding both *correlation coefficient* (a.k.a *matrix coefficient*) and *Euclidean distance* (a.k.a *norm-2*) given every possible k value, followed by visualizing the optimal results recommended by the two criteria. In order to achieve a comprehensive understanding and a systematic study, every one of the seven data sets from *S-OBJ-A* will be used for the evaluation respectively. Note that all the results are obtained with a maximum number of iterations of 80, and 10^{-4} as the error threshold for convergence.

[1] https://en.wikipedia.org/wiki/Gini_coefficient

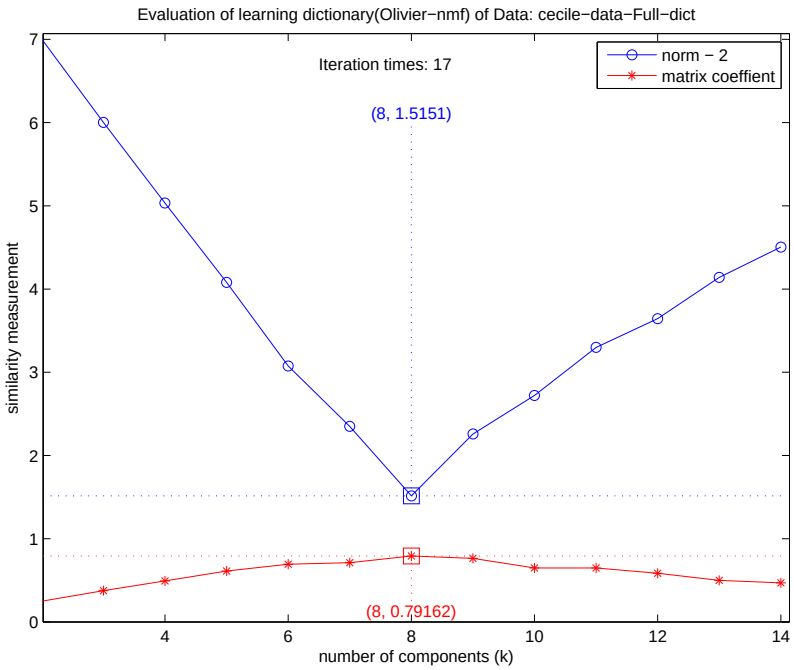


Figure 3.9: Evaluation of NMF learned dictionary (Full dictionary).

We first applied our evaluation to the full *S-OBJ-A* dataset. Since in this case the reference dictionary is of eight components, both criteria show (Figure 3.9) that NMF applied with the correct number of components results in a good learned dictionary. Besides, the visualized illustrations (in Figure 3.10 and 3.11) show that the optimal learned dictionary really approximates the *Full dictionary* reference to a very high extent, despite some visual errors. In particular, the shape description is a bit noisy and the shape associated to the color words are not completely zero, but very close.

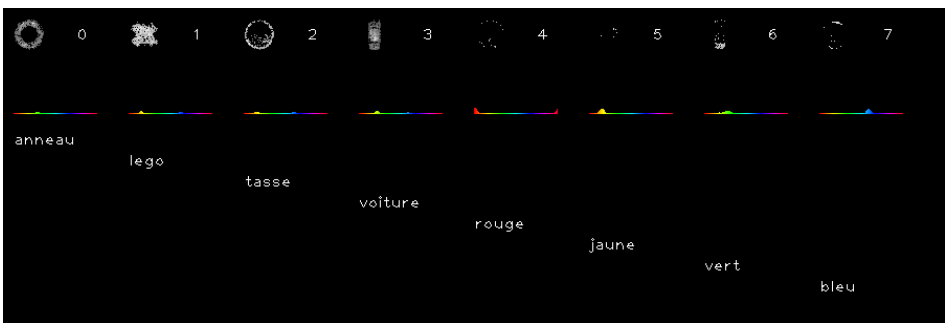


Figure 3.10: Visualization of the optimal NMF learned dictionary (Full dictionary) in norm-2 criterion.

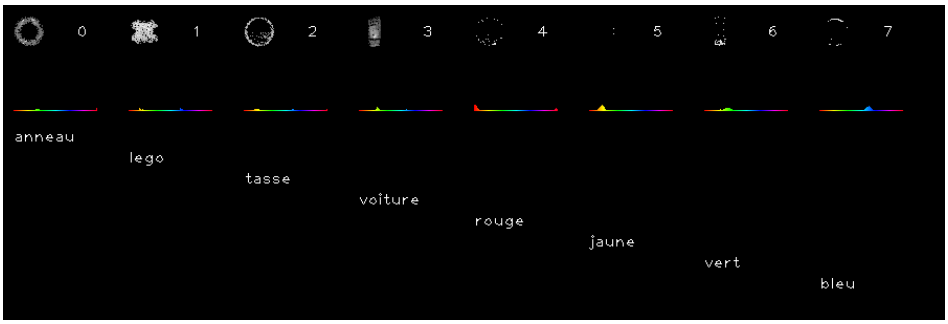


Figure 3.11: Visualization of the optimal NMF learned dictionary (Full dictionary) in correlation coefficient criterion.

Table 3.1 summarizes the results obtained for the reduced datasets. The complete results can be found in Appendix B.1. We observe that, in all cases, the value of k corresponding to the real number of words to learn consistently leads to the lowest error and a good learned dictionary.

Criteria \ Data set		***Full dictionary***	***2c-2s***	***3c-3s***	***4c***	***4c-2s***	***4s***	***4s-2c***
Correlation coefficient	max value	0.79162	0.9171	0.75409	0.96796	0.785	0.74471	0.78409
	optimal k	8	4	6	4	6	4	6
Euclidean distance	min value	1.5151	1.0969	1.2551	1.0313	1.582	1.4139	1.4499
	optimal k	8	4	6	4	6	4	6

Table 3.1: Performance of NMF learning compared to reference for all data sets of *S-OBJ-A*. For each criteria and dataset, the table reports the lowest error for all values of k tested and the value of k corresponding to this lowest value.

From this seven different cases, we can conclude firstly that the proposed evaluation algorithms using matrix similarity measure are efficient to evaluate the learned dictionary with regard to the reference one, thus providing suggestion about the right value of k, with which NMF has the best performance. Secondly, the learning algorithm of NMF with initialization of W and the normalization rule of "shape+color" and "label" respectively during iteration, as described in Section 2.3, is capable of generating a good learned dictionary of pure concepts in the format of "one label - one modality", approximating the reference dictionary, given the right number of components. In addition, repeated experiments with the above data also result in the same and stable performances.

3.4 Auto-determination of k by applying SV-NMF

Although we have shown that NMF with the correct value of k will lead to good dictionaries in the previous section, we still have to find a method for finding k when the number of relevant words is unknown. As introduced before, SV-NMF (see Section 1.4.2.4) can determine automatically the size of the learned dictionary, and we can evaluate a learned result by comparing it with the reference dictionary as well as the applicability of SV-NMF in our

experiments based on its overall performance. Note that all the results are obtained with *nu* (ie. the positive penalization parameter used to allow for a trade-off, between margin maximisation and training error, see Section 1.4.2.4) ranging from 0.05 to 0.95 with the step of 0.05, "linear kernel" and no normalization for column vectors in W.

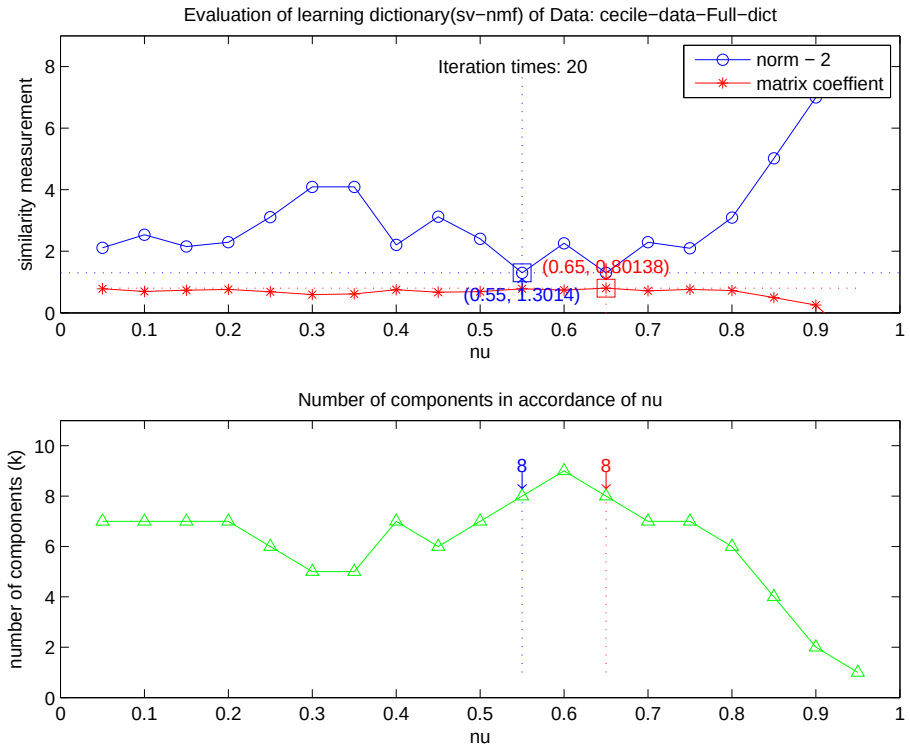


Figure 3.12: Evaluation of SV-NMF learned dictionary (Full dictionary).

Just like before, using the seven data sets from *S-OBJ-A*, the following descriptions will first illustrate the curves of criteria (ie. *correlation coefficient* and *Euclidean distance*) against *nu*. We will then show the found value of k according to the *nu* value recommended by the above criteria, followed by visualizing the learned results of SV-NMF with the corresponding optimal parameter setting.

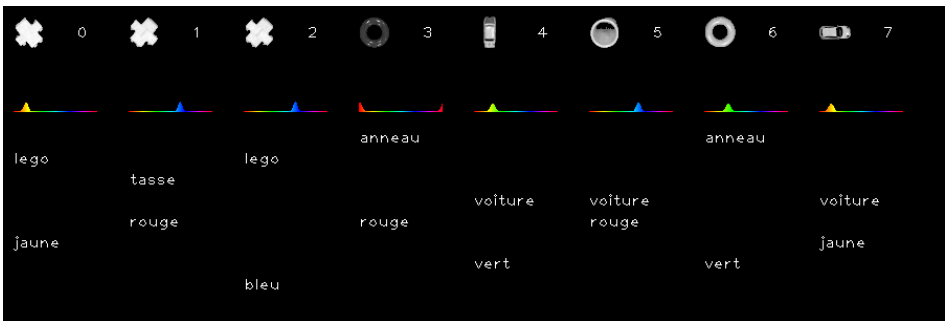


Figure 3.13: Visualization of the optimal SV-NMF learned dictionary (Full dictionary) in norm-2 criterion.

Figure 3.12 shows that with the *nu* values of 0.55 or 0.65 (depending on the used criteria), the learned dictionary reaches the best quality and that the number of components is

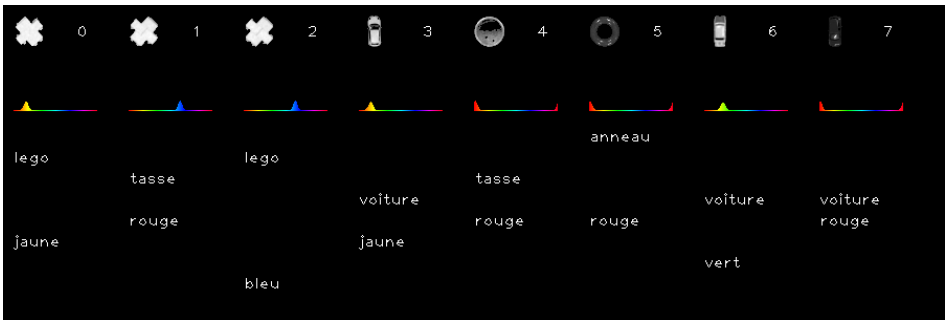


Figure 3.14: Visualization of the optimal SV-NMF learned dictionary (Full dictionary) in correlation coefficient criterion.

correctly chosen as 8, equivalent to the ground truth number. However, by comparing the visualized dictionaries (Figures 3.13 and 3.14), we notice that instead of producing component as simple concept (in "one label - one modality" format), SV-NMF tends to generate composite concepts, which does not conform to our assumption of a good dictionary. Further more, some entries of the dictionary are wrong in contents.

Table 3.2 summarizes the results obtained for the reduced datasets. The complete results can be found in Appendix B.2. We observe that, in all cases, similar conclusions can be drawn. The number of components is often, but not always, correctly estimated, but the resulting dictionaries are always quite far from the reference ones. Moreover, the value of *nu* leading to the best dictionary is always different.

Criteria \ Data set		***Full dictionary***	***2c-2s***	***3c-3s***	***4c***	***4c-2s***	***4s***	***4s-2c***
Correlation coefficient	max value	0.80138	0.966	0.91964	0.86686	0.80875	0.7742	0.86358
	nu (ν)	0.65	0.4	0.25	0.8	0.5	0.45	0.75
	optimal k	8	4	6	4	5	5	6
Euclidean distance	min value	1.3014	1.0496	1.1558	1.2083	1.887	1.6507	1.3653
	nu (ν)	0.55	0.3	0.25	0.8	0.45	0.45	0.75
	optimal k	8	4	6	4	7	5	6

Table 3.2: Performance of auto-determination of k by applying SV-NMF for all data sets of *S-OBJ-A*. Values in red indicate errors in the determination of k by SV-NMF.

From these experiments, we can conclude that 1). the automatic determination function for k from SV-NMF model proposed in [58] is proved effective to some extent, given a correct parameter setting of *nu*; 2). however, in different experiments, the optimal *nu* variates, which makes this model not applicable in our case because we have to decide *nu* empirically and manually; 3). in addition, the quality of learned dictionary via SV-NMF is not so good as we expect, usually consisting of missing, repeated or contradictory items.

3.5 Determination of k for NMF without reference

Now that it seems not easy to find applicable ready-made methods to auto-determine k without using the reference, we therefore propose a comprehensive quality measure for this purpose. Using the same methodology, we evaluate the *reconstruction error* and the *1-Gini* index for every learned dictionary with values of k ranging from 2 to 14. The comprehensive indicator formulated in Equation 3.5, i.e., the combination of the above two criteria, will be adopted as the indicator to find the optimal k. And we will also illustrate the performance of NMF with such a parameter.

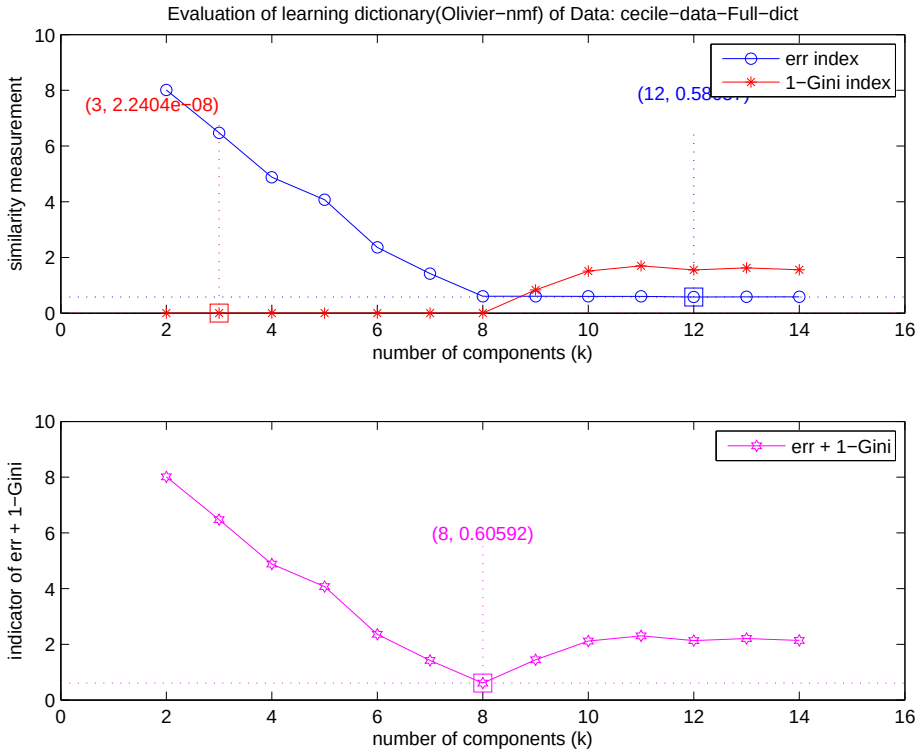


Figure 3.15: Evaluation of NMF learned dictionary (Full dictionary) in criteria of error and Gini index.

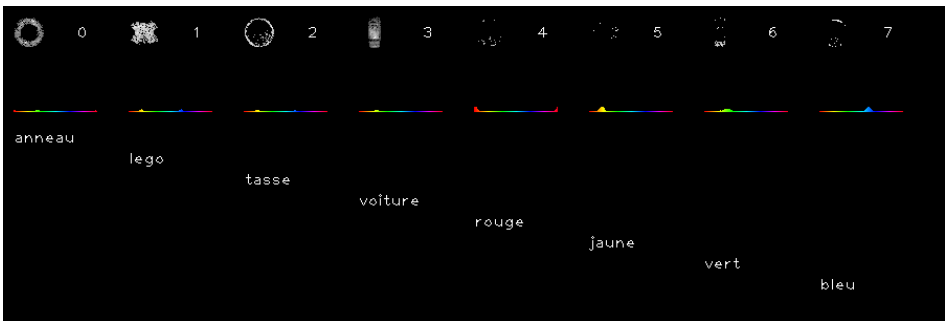


Figure 3.16: Visualization of the optimal NMF learned dictionary (Full dictionary) in criteria of error and Gini index.

Figure 3.15 shows the values of the two criteria and their combination as a function of the value of k for the NMF algorithm. We can see that the optimal values of error curve and (1-Gini) curve lie apart (see the top), but the combined curve (see the bottom) reaches its minimum value when $k = 8$, exactly the right number of components of the reference. With $k = 8$, learning via NMF results in a very good dictionary (Figure 3.16, as was previously shown).

Table 3.3 summarizes the results obtained for the reduced datasets. The complete results can be found in Appendix B.3. We observe that, in all cases, similar conclusions can be drawn. The number of components is always correctly estimated, and the resulting dictionaries are very close to the reference dictionaries.

Criteria \ Data set		***Full dictionary***	***2c-2s***	***3c-3s***	***4c***	***4c-2s***	***4s***	***4s-2c***
Reconstruction error	min value	0.58007	0.2837	0.3826	0.28633	0.42	0.19472	0.38529
	optimal k	12	14	12	12	12	4	10
1-Gini index	min value	2.2404×10^{-8}	4.2991×10^{-7}	1.0546×10^{-7}	4.7441×10^{-7}	8.8561×10^{-8}	6.0488×10^{-7}	1.2396×10^{-7}
	optimal k	3	2	2	2	2	2	2
Comprehensive quality measure	min value	0.60592	0.33183	0.39992	0.34057	0.47421	0.19472	0.41137
	optimal k	8	4	6	4	6	4	6

Table 3.3: Performance of determination of k by applying NMF without reference for all data sets of *S-OBJ-A*.

From the above results, we could find the common points that with every experimental data set, using k determined by the comprehensive index combing both reconstruction error and (1-Gini) index, the NMF model gives in every case highly stable and correct learned dictionary by comparison with its reference as ground truth. In fact, for every purple curve (corresponding to the comprehensive index), there is a local minimum, left to which is a descending trend which is dominated by the reconstruction error while right to which is a ascending trend mainly influenced by the sparsity measure, and this is concordant with what have been analyzed in Chapter 3.2.2, making the optimal k as the right number of the ideal dictionary components being indicated by the turning point of two trends.

In a realistic scenario, this method can by applied by starting the experiment with a value of $k = 1$, and then, at each new sample, evaluate the result of the learning with the values of k and $k + 1$. If using $k + 1$ lead to a lower value of our combined coefficient, it can be used as the new reference for processing the next sample.

3.6 Discussion

In this chapter, we tried to validate the performance of our proposed NMF learning model by evaluating the quality of a learned dictionary representing the word-meaning associations, compared with a corresponding reference. Thanks to the non-negativity of NMF that give rises to good interpretability, we could simply conduct the comparison by visualizing the learned results. Another key issue of applying NMF is the determination of k as the number of hidden topics, on which all input samples could be reconstructed by means of linear combination. The first trial was to find an optimal k from a range of candidates, with which different learned dictionaries are created and then compared with references, and proved valid via case by case (in total seven) evaluations. Then as a representative of the current research focusing on this topic, SV-NMF, which is supposed capable of auto-determining k, was evaluated with the same experiment data as those processed by NMF so as to evaluate its suitability in our experiments. As a result, since SV-NMF seemed not adapted for our experiments, we proposed a comprehensive quality measure (as the combination of reconstruction error and sparsity measure) to determine k without the needs of using references and finally proved its effectiveness by evaluating every case of our experiment data sets.

Concerning the evaluation measures, when a reference dictionary is available as the ground truth, our solutions to estimate the similarity between a learned dictionary and a reference one proved to be effective. As it is not possible to compare reference and learned dictionaries directly, we sought to describe/reconstruct a learned dictionary by using the linear combination of reference dictionary bases, then reorder the linear coefficient matrix (computed also by NMF based method) in a way as similar as possible to an identity matrix. We then applied the least square or correlation measure between the obtained matrix and the identity matrix, with a penalty term considered when the learned dictionary is not of the same size as the reference one.

In the case where a reference dictionary is not available, the reconstruction error and sparsity measure were used jointly to find the optimal dictionary and shown to give the correct prediction of the number of components. For the reconstruction error curve, there is an obvious decline from the situation in which components are insufficient to the situation where component number is just right and a flat trend afterwards; on the contrary, the sparsity curve (for sub-column vectors of label section in W) will first be flat before ascending when the dictionary components appear superfluous. Hence, the minimum of the combination of the two curves gives the right number of components.

We evaluated the quality of learned dictionaries produced through a series of experiments given a comparatively small set of data in seven dictionary cases (1 full version and 6 reduced versions). Generally speaking, the NMF model manages to learn objects by extracting the pure concepts concerning shape and color, therefore learning that objects are the combination of different features. The learned dictionaries are very close to the reference and present the right format of “one label - one modality”. Besides, in cases of both “with reference” and “without reference”, k is determined correctly, thus ensuring an optimal NMF output in every experimental scenario. On the contrary, in our evaluation of SV-NMF, most learned

dictionaries do not contain pure concept and always contain bad entries (eg. missing, repeated, contradictory). Further more, SV-NMF does not have a fixed parameter setting of *nu* for the experiment series, and this prevents its further utilization before some improvements are made.

SV-NMF, proposed in [58], is an endeavour to interpret and solve the NMF model from a geometrical aspect, however, in practical use concerning our problem, it is not adapted for several reasons:

1) SV-NMF seeks to find a conic hull, as small as possible, of input data V to determine W, but as a matter of fact in our experimental scenario, any input sample itself is not qualified to be taken as one component in the learned dictionary. That's why the learned dictonary, ideally containing only pure concepts, always includes composite concepts, especially just copying one of the input samples (eg. the Component 1 to 4 in Figure B.36) and repeated results (possibly because they are in close proximity of a "border vector" of the smallest conic hull, eg. in Figure B.32);
2) in real world scenarios, the reasonable decomposition of a complex object is not unique, for instance, a *white horse* might be regarded as the combination of two concepts *white* (from the category of color) and *horse* (from the category of biology), yet it can also be decomposed as a *horse head*, a *horse body*, four *horse legs*, a *horse tail*... Therefore, instead of insisting on an universal solution regarding the uniqueness of decomposition (as proposed by SV-NMF), efforts should be taken to explore the conditions on which a desired and reasonable decomposition would appear.

In fact, the proposed NMF model as described in Chapter 2.3 succeeds in solving partially, if not all, the above problems.

Now that the pure concept acquisition is validated via our proposed NMF learning model in ideal conditions. We will carry on into experiments to deal with more realistic scenarios, where more objects are to be presented and we will go beyond the "keywords only" data for the evaluation in "full sentence" noisy circumstances. Moreover, we will study how active learning can improve the learning speed in incremental scenarios.

Chapter 4

Incremental word-meaning learning

Contents

In this chapter, we will study the behavior of the NMF and LDA algorithms on the task of incrementally learning word-meaning associations. Unlike the theoretical evaluations with only a small set of data presented in the previous chapter, we prepare new sets of objects, in higher quantity as well as higher complexity of image representation, to evaluate if our learning method is capable of dealing with more real world scenarios such as describing an image or recognizing a named object. Besides, some of the experimental scenarios will present both *referential* and *linguistic* ambiguities, demonstrating that the proposed learning models can tackle situations in which multiple objects are referred when only one descriptive

sentence is spoken or/and full sentences (including unrelated words) are used instead of just keywords. Regarding the automatic determination of the number of hidden topics (referred to as k throughout this manuscript), new methods are proposed to suit the above scenarios that, contrary to the solution proposed in the previous chapter, do not require a posterior evaluation of the learning result for several values of the parameter. In all this chapter, we evaluate the learning progress during incremental learning experiments and show that it can be improved by the proposed active learning strategies. Finally, we evaluate the potential for the betterment of our proposed learning model by applying more complex visual features.

To note that for every individual experiment, a temporary conclusion or short comment is attached after the description based on the local observation of the experimental results; however, comprehensive conclusions according to the overall analyses of all experiment series will be drawn finally in the chapter of Discussion and Conclusion.

4.1 Experimental data

Using the same experimental settings (introduced in Chapter 3.1, including the use of the rotation technique described in Chapter 2.1.1.1) in which we obtain *S-OBJ-A*, we recorded more objects and denoted them as *S-OBJ-B* and *S-OBJ-C*. Additionally, a fourth dataset, *S-OBJ-D*, contains the HOG descriptors of several objects (images printed on cubes) detected by applying depth based object segmentation (see Section 2.1.1.1).

4.1.0.1 S-OBJ-B

In this dataset, there are 24 objects which represent 4 colors as "blue (*bleu*)", "green (*vert*)", "red (*rouge*)", "yellow (*jaune*)", and 7 shapes as "cup (*tasse*)", "ring (*anneau*)", "lego", "apple (*pomme*)", "compass (*boussole*)", "car (*voiture*)", "book (*livre*)", constituting in total 11 word meanings, as shown in Figure 4.1.

A total of 77 samples were recorded from three different teachers and manually categorized as *correct* (i.e. containing both correct labels of color and shape along with other unrelated words), *half-correct* (i.e. containing only one of color or shape label) and *corrupted* (i.e. containing no correct label). For example in Figure 4.2, "Une tasse jaune" correctly describes a yellow cup. When describing a green lego, the speaker says: "Ha, c'est un lego vert", however the word "lego" fails to be recognized and leads to a half-correct sample on the right. A corrupted sample appears when none of the correct words are recognized, for example when the original sentence "c'est un anneau vert" is misunderstood as "c'est quand halloween". Among all recorded samples, there are 70.13% correct samples, 11.69% half-correct samples and 18.18% corrupted samples. Overall, the 11 words to learn represent only 17.74% of the total number of words present in the samples. Similar to the cases in Chapter 3, there is a reference dictionary created for evaluation.

Figure 4.1: Twenty-four objects with eleven concepts of shape and color

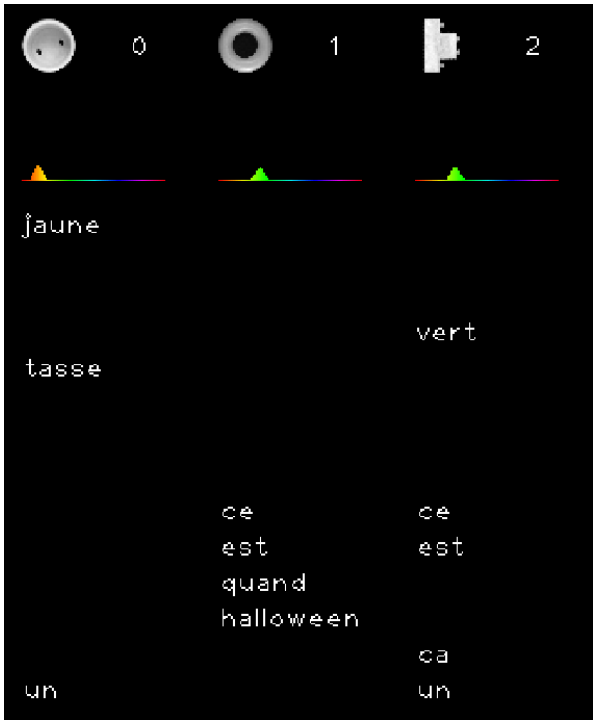


Figure 4.2: Examples of correct, corrupted and half-correct samples.

4.1.0.2 S-OBJ-C

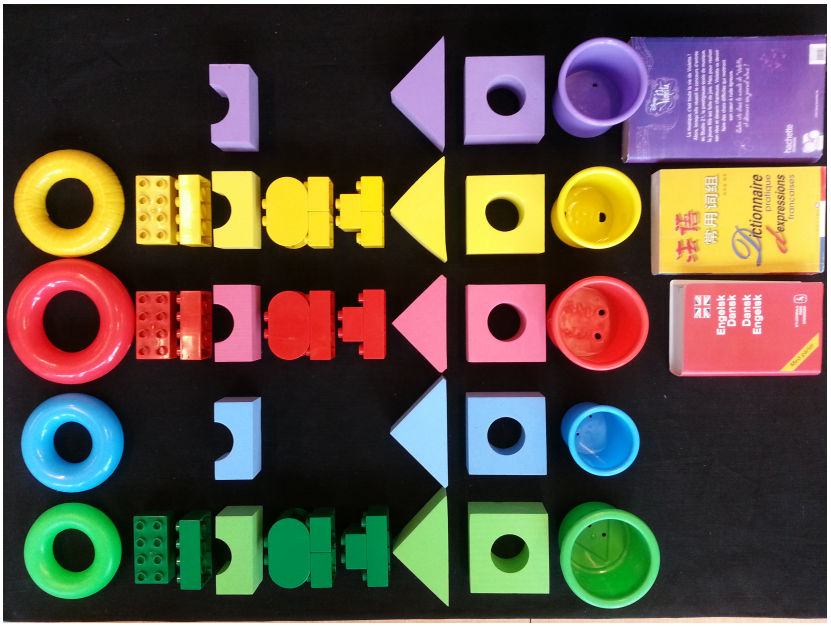


Figure 4.3: Thirty-nine objects with fifteen concepts of shape and color.

This dataset (Figure 4.3) is similar to the previous one, but is larger. It contains 39 objects presented in 5 colors and 10 shapes, thus giving 15 meanings of shape and color. They can be grouped by color as "green (*vert*)", "blue (*bleu*)","red (*rouge*)", "yellow (*jaune*)", "purple (*violette*)", and categorized by shape as "ring (*anneau*)", "raft (*radeau*)", "wall (*mur*)", "stool (*tabouret*)", "apple (*pomme*)", "lego", "triangle", "compass (*boussole*)", "cup (*tasse*)", "book (*livre*)".

We recorded 153 samples with the help of ten volunteers, in which every object was described at least three times and most of them four times. Each object was described by two keywords, but the mean sentence length is 4.026, thus containing in average 2.026 irrelevant words. We did not created a reference dictionary but we separated the data set into a training set by selecting 3 samples for every one of the 39 objects (thus summing up to a total of 117 samples) and keep the remaining 36 samples, which cover all the keywords, as testing data to monitor the performance of learning.

4.1.0.3 S-OBJ-D

Figure 4.4: Twenty cubics of object images with every side of cubic in different color.

In order to evaluate our proposed models with more complex visual representations and more shape and color concepts combinations, new objects are "created" by making cubes with printed objects on their sides. As shown in Figure 4.4, we made 20 cubes in total, each side of which represents a color, containing "blue", "cyan", "green", "purple", "red" and "yellow". This gives a total of 120 object instances. All cubes have the white background, which is supposed to have no influence on the HOG description, and are drawn with cartoon images of "bird", "boat", "book", "cake", "car", "cat", "cow", "dog", "dolphin", "fish", "frog", "hat", "house", "pig", "pumpkin", "rat", "sealion", "shark", "vase" and "whale".

By using the proposed segmentation (see Section 2.1.1.1) and HOG recording techniques (see Section 2.1.1.2), every side of a cube was described and recorded 7 times, thus forming a total number of 840 samples as database. In order to get the testing data, we first randomly chose one side of the object from which one sample was then randomly selected out of the seven samples, therefore we obtain 20 samples for testing. As for the remaining five cubic sides, we prepared two different sets of training data: "T3" in which 3 samples were randomly selected and "T1" with only one sample for each of the five cubic sides. In this way, every object from the testing data never appears during training (although its HOG and color properties separately do). Besides, the sentence collection procedure remains the same as before and finally every sample is represented by an histogram of three sections in terms of HOG, color and linguistic channel.

4.2 Performance evaluation

In the previous chapter, we used reference dictionaries for the evaluation. However, in most realistic scenarios, these reference dictionaries are not available, especially in online learning situations. In this chapter, we therefore applied other evaluation strategies by evaluating the learned dictionaries on recognition tasks. This approach is very similar to the one used in cognitive studies on human or animal learning performance, as they treat the system as a "black box". In all subsequent experiments, there are three evaluations criteria in total to illustrate the learning performances: the first two which are mostly applied are derived from the common interactive activities such as a parent asks a child to recognize a toy (by correctly telling its name or distinguishing it from other toys), and the third one which is used comparatively less simulates how an infant self-evaluates what has been learned.

Note that in order to exhibit the best performance results possible for each learning algorithm, we didn't specifically develop an online and incremental version, but we emulate incremental learning by performing standard "batch" learning with new samples added to the previous training data set. Besides, in most following experiments (except for those in Section 4.5.1), the feature quantification (see. Chapter 2.1.3) is applied for all data before learning starts so as to reduce the influence of quantification errors caused by the different appearing orders of samples.

4.2.1 Task of image recognition (T2img)

The first evaluation method called "Text to Images" (T2img) evaluates if the system makes it possible to recognize the image of an object given its spoken description.

"T2img" is the task of choosing a correct image from a group of candidate pictures by matching features described by a corresponding linguistic text (T_j). The whole process simulates the everyday situation where the teacher utters a textual description about an object and the learner has to designate the corresponding object. This evaluation criteria is implemented for both LDA and NMF with specific procedures.

For LDA, which is using clustered visual description, we first estimate the hidden topic distribution associated to the textual description T_j: $P(\mathbf{z}|T_j)$, and reconstruct the associated vision feature channel using $P(\omega_j|T_j) = \sum_k P(\omega_j|z_k) \cdot P(z_k|T_j)$ where $\omega_j \in S \cup C$ (resp. $\omega_j \in H \cup C$) and S (resp. H), C correspond to the sets of shape (resp. HOG) and color VQ symbols respectively as introduced in Section 2.1.3. Then for every testing image d_i^{test}, we compute the likelihood

$$\mathcal{L}(d_i^{test}|T_j) = \sum_l Cnt(\omega_l) \cdot \ln P(\omega_l|T_j) \tag{4.1}$$

where $\omega_l \in S \cup C$ (or $\omega_l \in H \cup C$) and $Cnt(\omega_l)$ is number of occurrence of visual cluster ω_l from the testing sample d_i^{test}. Note that penalty is imposed when $Cnt(\omega_l) > 0$ while $P(\omega_l|T_j) = 0$ by applying $\ln P(\omega_l|T_j) = -5000$ for the reason that the VQ feature ω_l totally fails to be reconstructed. The object whose likelihood is the highest is taken as the answer

and we compute the overall percentage of correct answers.

For NMF, which is using raw visual histograms, we adopted the approach proposed in [123]. Since visual histograms matching would easily introduce errors, we preferred to perform matching in the word space. For this, for each testing sample :

$$V_i^{test} = \begin{bmatrix} V_{i-shape}^{test} \\ V_{i-color}^{test} \end{bmatrix}$$

we first compute the coefficient vector of hidden topics H_i^{test} associated with the visual description of each testing object i by minimizing the distance

$$D_{KL}([V_{i-shape}^{test}, V_{i-color}^{test}]^T \| [W_{shape}, W_{color}]^T H_i^{test}) \tag{4.2}$$

and then reconstruct the textual description of this testing object:

$$V_{i-word}^{test} = W_{word} H_i^{test} \tag{4.3}$$

Finally, we find the object in the testing set whose textual description is the closest to T_j by computing

$$\chi^2(T_j, V_{i-word}^{test})$$

for all i and count the percentage of all right answers among the total number of testing objects.

4.2.2 Task of image description (Img2T)

The second evaluation method called “Images to Text” (Img2T) evaluates if the system makes it possible to correctly describe an image.

“Img2T” is the task in which an image is shown to a learner who has to give a related textual description of its features (eg. concerning shape or HOG and color). It is quite similar to the behavior of annotation of an image.

For LDA, given a test image $d_i^{test} = \{s_i, c_i\}$ (or $d_i^{test} = \{h_i, c_i\}$) where $s_i \in S$, $h_i \in H$ and $c_i \in C$, we also estimate the hidden topic distribution associated to d_i^{test}: $p(\mathbf{z}|d_i^{test})$, and reconstruct the associated word description $P(T_i|d_i^{test}) = \sum_k P(T_i|z_k, d_i^{test}) \cdot P(z_k|d_i^{test})$. The two words with the maximum $P(T_i|d_i^{test})$ values are chosen to form R_i, the set of reconstructed words while G_i is denoted as the ground truth words corresponding to d_i^{test}. Then we apply equation (4.4) to compute the “Img2T" performance (in percentage):

$$score_{Img2T} = \frac{100}{NT} \sum_{i=1}^{NT} \frac{Card(R_i \cap G_i)}{Card(R_i \cup G_i)} \tag{4.4}$$

where $Card(X)$ is the number of elements of X and NT the total number of testing samples.

For NMF, from $V_i^{test} = \left[V_{i-shape}^{test} V_{i-color}^{test}\right]^T$, we reconstruct the hidden topics by minimizing the distance from Equation 4.2 and reconstruct the word modality (using Equation 4.3) to get the reconstructed V_{i-word}^{test}. We then take the two words with indices of the largest two values from V_{i-word}^{test} and check if they are identical to the ground truth words concerning its shape and color. We come to the score for the overall testing objects in percentage also by using Equation 4.4.

4.2.3 Reconstruction error

Both NMF and LDA are approximation algorithms that reconstruct the data from a set of underlying topics, thus leading to a reconstruction error. This reconstruction error could be regarded as a self-evaluation procedure from a learner who tries to re-describe an object with the features that he/she has learned despite the errors which might occur. In our experiments, we focus only on the reconstruction error of visual modalities (ie. of shape/HOG and color) which makes it possible to estimate if our method has a good knowledge of an object.

Both NMF and LDA have expressions related to the reconstruction error. For LDA, we make use of the log-likelihood of the data as described in Equation 4.1: $log\mathcal{L}(d_i^{test}|T_i)$. It is in fact related to the opposite of the error as its value increases during training. For NMF, we directly utilize the KL divergence distance D_{KL} from Equation 4.2.

4.3 Policy of the determination of the dictionary size in incremental experiment settings

The estimation of the optimal value of k (ie. the dictionary size or the number of components for NMF and number of topics for LDA) as the number of components in the previous chapter is essentially posterior based: an optimal value is chosen from trials given a list of candidate parameter values. This method is however sometimes untractable as the training is too computationally heavy.

Therefore, alternative approaches were used to determine the optimal value of k in an incremental experiment setup, concerning two scenarios of "keywords only" and "full sentence" with respect to learning models of both NMF and LDA.

For LDA, the same strategy is used for both scenarios, which is based on the current number of VQ clusters concerning all described samples and the log likelihood measure of LDA model with a certain value of k as the parameter for the number of topics:

1) we give an initial value of k as $k_{LDA}^{(1)} = n_{cluster}^{(1)}$, with $n_{cluster}^{(1)}$, the size of the first set of VQ symbols concerning S or H and C, that is to say the number of vector quantized clusters about shape (or HOG) and color of the first input samples.

2) when LDA trains the input data at time i, we use the parameter $k_{LDA}^{(i-1)}$, and perform an additional training with parameter $k_{LDA}^{(i-1)} + 1$ as the number of topics, and both training models result in two log likelihood measures (ie. $\log \mathcal{L} = \log(p(\mathbf{w}, \mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\phi}; \alpha, \beta))$, derived from Equation 1.36). We then take the value of k that gives the best likelihood:

$$k_{LDA}^{(i)} = \underset{k=\{k_{LDA}^{(i-1)}, k_{LDA}^{(i-1)}+1\}}{\arg\max} (\log \mathcal{L}(\cdot; k))$$

additionally, we limit the value of $k_{LDA}^{(i)}$ as

$$k_{LDA}^{(i)} = \min[\max(k_{LDA}^{(i)}, n_{cluster}^{(i)}), 2 \times n_{cluster}^{(i)}]$$

so as to control the over growth of $k_{LDA}^{(i)}$.

In short, the general idea is that the number of topics for LDA is in correlation to the number of VQ clusters, which is a required pre-processing before applying LDA; however, in practice, we found that once a small amount of redundant topics are given, the overall performance would improve and this improvement would be measured just by the log likelihood of LDA training model.

For NMF, the choice of k in the "keywords only" scenario is just the number of keywords which have been used to describe all encountered samples and in the "full sentence" experiment, k takes the value of the number of selected keywords as the result of TF-IDF word filtering (see Section 2.2).

From the next section and on, we will test the learning ability of our proposed models to conduct cross-situational learning tasks with ascending complexities, starting from the case of the lowest ambiguity where object is given to the learner one by one and described with only keywords, then to the situation where the noisy words are added during the single object description, followed by the scenario when multiple objects are present each time while full sentences (containing more than keywords) are used for description, until to the final stage with active learning strategy and finally using a more complex vision descriptor.

4.4 Learning with unambiguous noise free data

In this section, we will demonstrate the learning ability of the two proposed models in the simplest experimental setting. An incremental learning scenario is considered which simulates the situation of a teacher who randomly chooses one single object and describes it with its associated keywords (corresponding to the KW&S scenario of Section 2.1.4). There is consequently no ambiguity regarding the referent and the linguistic ambiguity is at its lowest level. Both learning methods are used and their performances are compared.

Using the dataset *S-OBJ-C*, a sample is randomly chosen and presented which can not be

reused any more, therefore there are 117 learning steps for a learner to finish the whole learning process. From the point of view of the learner, at every learning step, a new sample will be added to its training database composed of all the used samples, and the learning model will retrain the whole data before the 36 testing samples are used to evaluate learning results via "Img2T" and "T2img". It should be noted that, as described in Section 2.3 before, we here do not develop a true online version of the learning algorithms, because the objective at the current stage is more focused on estimating the upper bound of the performance achievable by each method.

In order to present the learned results pertinently, we choose to report the testing performance as a function of the number of samples used for training, up to the total number of samples of 117. The curves display the 75_{th}, 50_{th} and 25_{th} percentile of performance among 50 repetitions of the experiments.

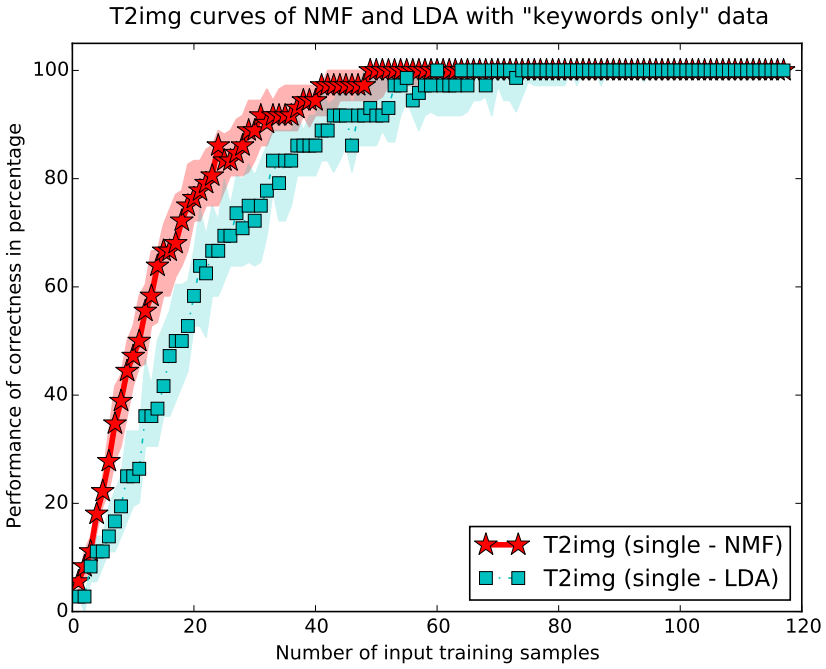


Figure 4.5: "T2img" performance of incremental learning with $KW\&S$ data from *S-OBJ-C*

Figure 4.5 and 4.6 show that both LDA and NMF are able to reach a performance of 100% before half of the training samples are given. Compared to [5], the best performance in which is less than 75%, the proposed learning methods perform better mainly due to two evident reasons: 1). within a specific category in terms of shape, the objects used in [5] are still different in shape while in this thesis most categories just have objects that possess almost the same shape but in different colors, which lowers the difficulty of recognition; 2). online version of LDA was used in [5], yet on the contrary, NMF and LDA still adopts the batch processing so as to achieve the potentially best performance.

Not beyond our anticipation of the two methods according to their mathematical properties, we observed that their learning progresses appear different: NMF reaches above 90% in

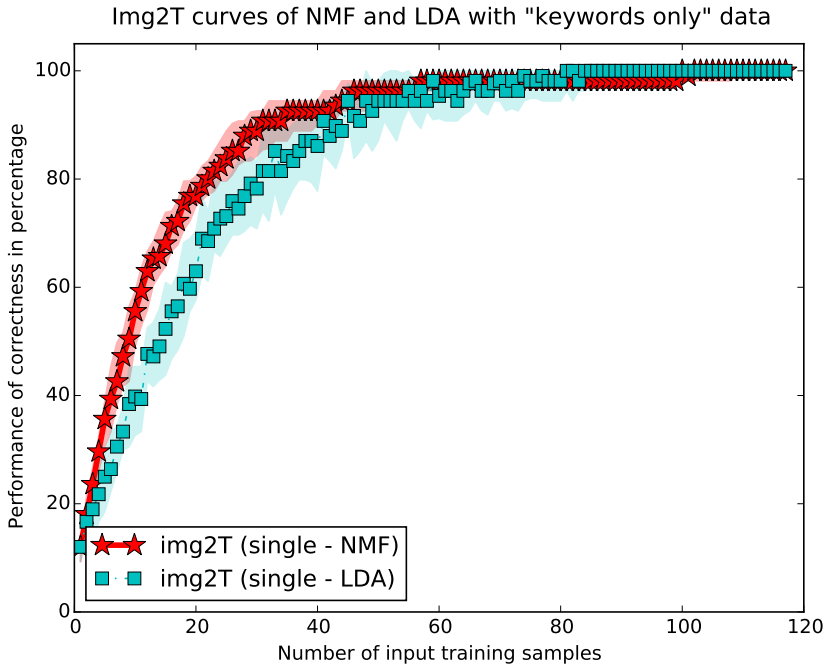


Figure 4.6: "Img2T" performance of incremental learning with *KW&S* data from *S-OBJ-C*

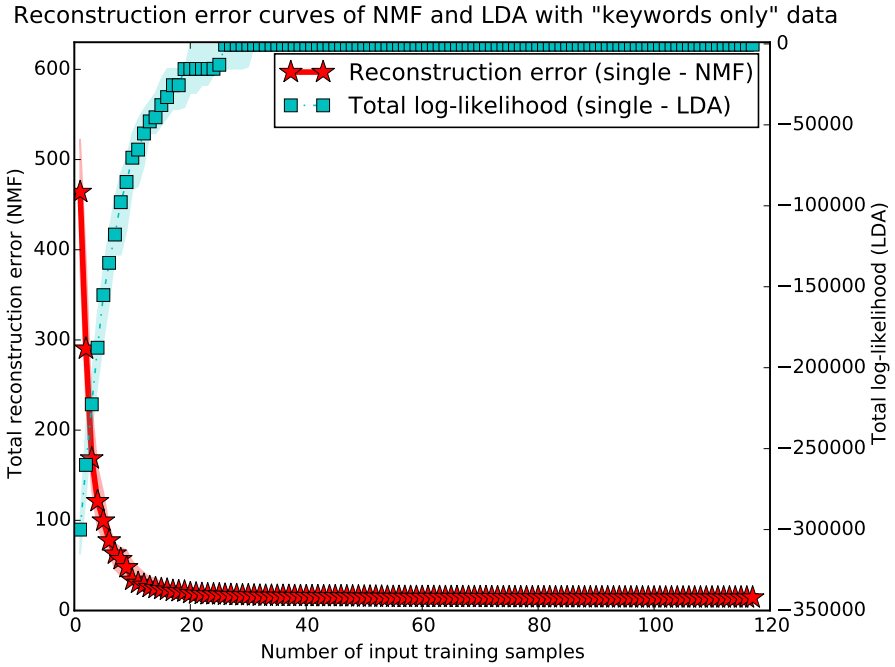


Figure 4.7: Reconstruction error curves of incremental learning with *KW&S* data from *S-OBJ-C*.

the task of "T2img" and 80% in "Img2T" after 25 samples while for LDA to attain the same level of performance, it needs around 35 samples. We can also observe that NMF consistently outperforms LDA regarding the learning speed in all criteria curves (in Figure 4.7, we can observe their different speeds of converging to 0 for their reconstruction error), showing its adaptation to the case of limited ambiguities in the language part. Two main facts could be used to explain the above differences: 1). when choosing the number of components (for NMF, noted as k_{NMF}) or topics (for LDA, noted as k_{LDA}), according to the policy in Section 4.3, k_{NMF} is exactly equal to the sum number of shape and color clusters while k_{LDA} is set redundant[1], thus more than the ground truth cluster numbers, making the learning more stable, but less precise; 2). unlike NMF which is directly specialized in dimensionality reduction of raw data, LDA with its statistic approach needs more samples to train accurately the generative model and therefore remains comparatively slow in learning.

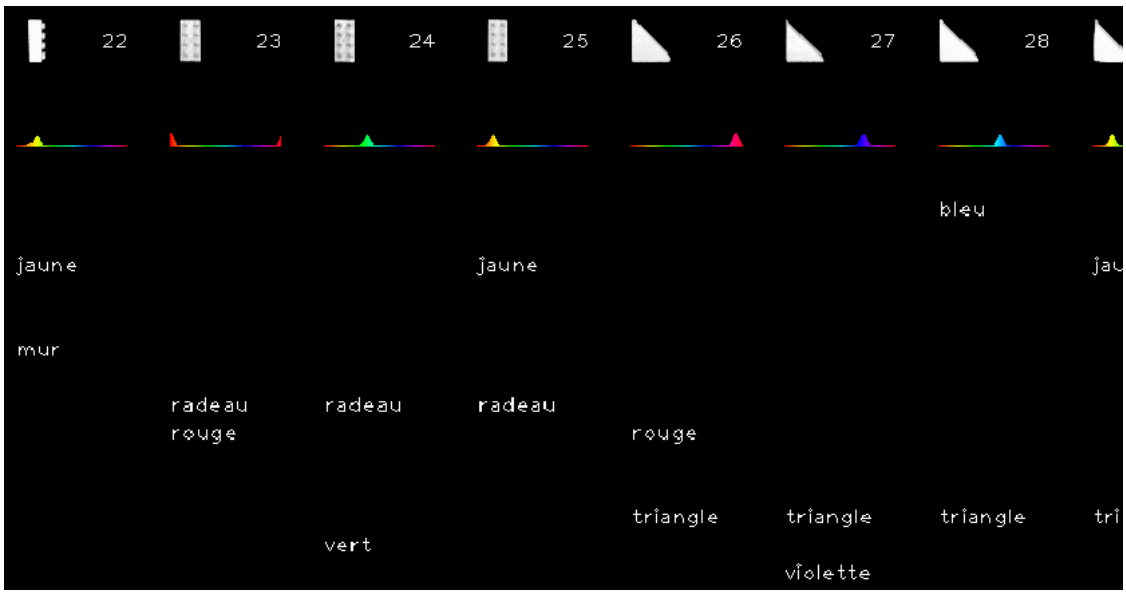
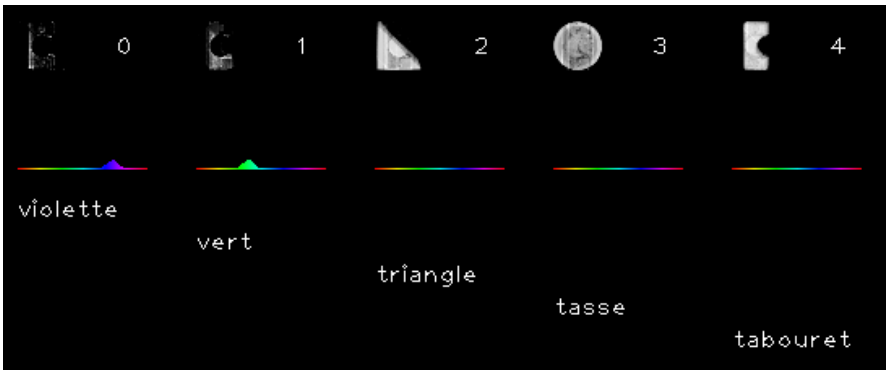


Figure 4.8: An excerpt of testing samples (which are 36 in total).

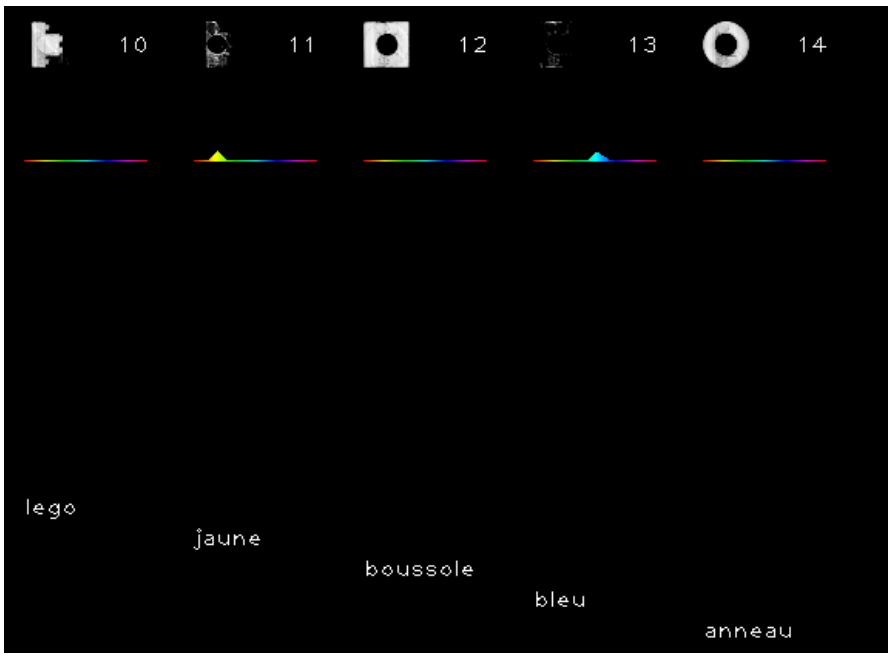
We notice small differences in the performance curves of "T2img" and "Img2T": "T2img" performs slightly better than "Img2T". If we take a look at some intermediate learning results, we find that when tested with the same object a learner would sometimes succeed in the task of "T2img", however fail in "Img2T", which finally causes the tiny gap between the two curves. For example, when given the Object 25 (*radeau jaune*, as shown in Figure 4.8) and based on the current (and obviously not perfect) learned results of components (as illustrated in Figure 4.9), a learning agent applying the NMF learning model reconstructs the tested object via the linear combination of the following principal components: *tabouret*, *radeau*, *lego* and *jaune*, with corresponding weights (whose value have been rounded) as 0.454, 0.307, 0.189 and 0.986. In "T2img", by applying $\chi^2(T_j, V^{test}_{i-word})$ as described in Section 4.2.1, it chooses the right object among all the objects on hearing the description "*radeau jaune*" because all other objects' textual descriptions are even farther from "*radeau jaune*". However in "Img2T", it

[1]In fact, if well learned, the hidden topics of LDA have several different types: most of them obey the format "one feature VQ symbol - one word label", but there are also possibilities where in a topic there might be several VQ symbols or word labels on some occasions and no symbols or words on others.



(a)

(b)



(c)

Figure 4.9: An example of a midterm result of learned components.

utters the names of components with largest weight values, that is “jaune”, “tabouret”, and as a result, leads to an error. A more intuitive way to explain this difference is to say that “T2img” chooses the best matching image in the test set, thus can give correct answers even with imperfect component dictionaries, while “Img2T” directly generates an answer, and thus is more sensitive to errors in the dictionary.

4.5 Learning with unambiguous noisy data

4.5.1 Validation of “NMF+TF-IDF” learning model by means of reference dictionary

Since using NMF to tackle directly the linguistic noisy data definitely does not lead to good performances, we proposed TF-IDF as a supplement to the learning via NMF (as described in Section 2.2). In order to testify this filtering effectiveness, we specifically conducted an incremental experiment with all 77 samples from *S-OBJ-B* (in which there are bad samples whose linguistic descriptions have partial or even no related words concerning shape and color, thus corresponding to the FS&S scenario of Section 2.1.4), and 10 learning experiments had been performed independently by processing all the samples in random order before the values of mean and variance concerning their performances were computed.

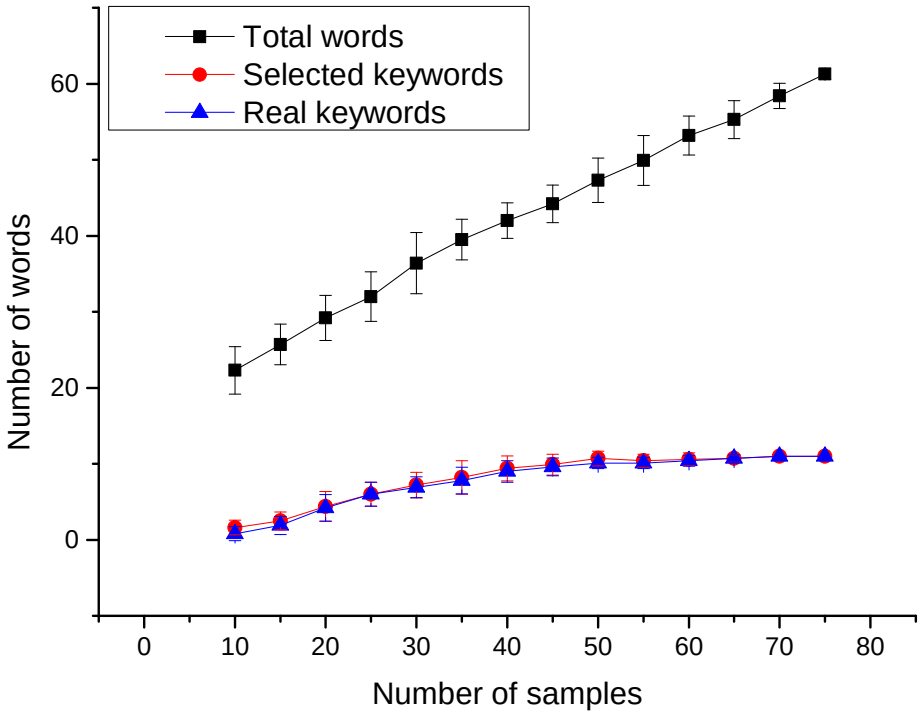


Figure 4.10: Performance of keywords’s filtering with *FS&S* data from *S-OBJ-B*.

Figure 4.10 plots the evolution of the total number of different words encountered in the samples (in black), the number of selected words by our filtering scheme (in red) and the real number of correct keywords in these selected words (in blue). We can see that our approach

selects an approximately correct number of words during the whole experiment, and converges to the correct total number of keywords after approximately 50 samples.

Two metrics are defined for the evaluation of the quality of word filtering and word-meaning dictionary. For the word filtering part, we compare the set of filtered words F with the set of reference words R using the equation:

$$s_{word} = 100 \times \frac{Card(F \cap R)^2}{Card(F) \times Card(R)} \tag{4.5}$$

where $Card(X)$ is the number of elements of X. This score is maximal when $F = R$ and decreases when the set of filtered words lacks some reference elements or when it contains additional erroneous words.

After word filtering, we performed dictionary learning with NMF, using only the words that are selected. We then compared the learned dictionary with the reference dictionary (using the same method as described in Section 3.2.1) applying the following formula:

$$s_{dict} = 100 \times \frac{\left(\sum_{i \in R} \sum_{j \in F} \delta(i,j) \cdot e^{-\chi^2(r_i, f_j)} \right) \cdot Card(F \cap R)}{Card(F) \times Card(R)} \tag{4.6}$$

where $\delta(i,j)$ is the Dirac function that equals 1 when the most activated word of the learned dictionary entry j is the same as in the reference word i, 0 otherwise. $\chi^2(r_i, f_j)$ is the χ^2 distance between the visual feature part of learned entry j and reference entry i. This measure is maximum when the learned dictionary is equal to the reference dictionary and decreases when the selected words are different from the reference or when the definitions of the selected words differ from their reference definitions.

Figure 4.11 shows the mean and variance of these values with incremental amount of training samples. We can see that the word filtering improves its performance another time and reaches the maximum score after 70 samples in all cases. The dictionary quality follows very closely the word filtering quality, showing that the dictionary learning using NMF is efficient and that the overall quality mainly depends on the word filtering. The difference with 100 is mostly due to remaining noise in the shape description.

As the results show, the "NMF+TF-IDF" cross-situational learning method manages to deal efficiently with noisy and ambiguous input taken from vision and speech recognition, where correct words represents only 17.74% of total words. Therefore, we would like to test the proposed method with more realistic and challenging tasks subsequently.

4.5.2 Comparison between "NMF+TF-IDF" and NMF only

Now that "NMF+TF-IDF" has been proved its power of word-meaning learning while effectively dealing with linguistic noise and ambiguities, we still wonder to what extent TF-IDF helps improving the learning process.

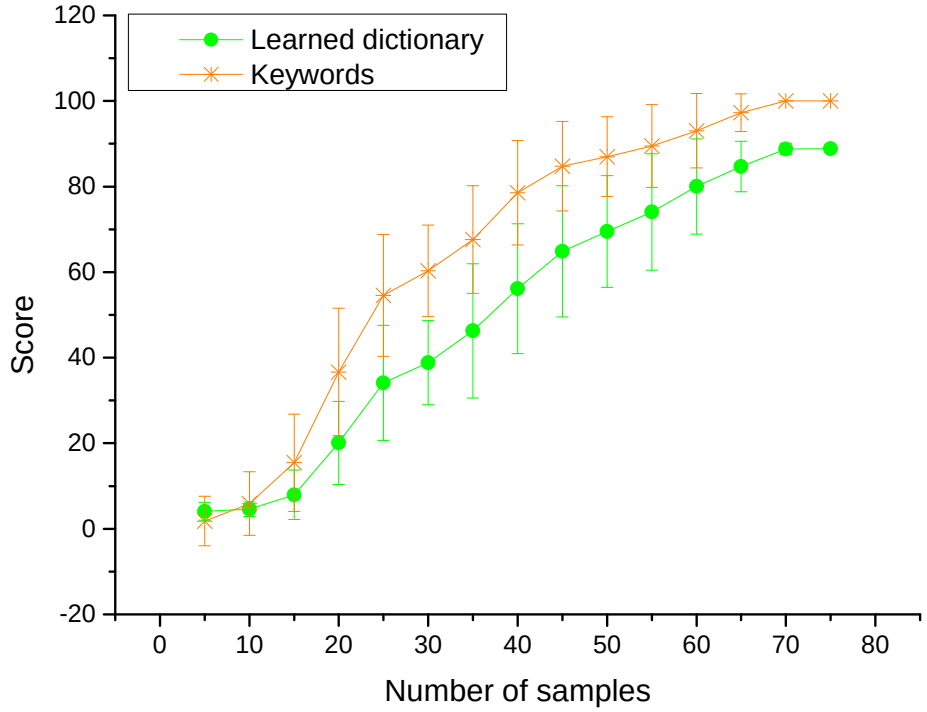


Figure 4.11: Quality of keywords and dictionary learning with data from *S-OBJ-B*.

We adopt the same experiment settings as described in Section 4.4 but with the samples whose descriptions are full sentences instead of mere keywords. We simulate the scenario of a teacher who randomly chooses an object (which is still not reusable) and describes it with a natural descriptive sentence.

The performance of two different learning agents, one of which uses TF-IDF to filter out unrelated words while the other one just accepts all it has received, are illustrated by "T2img" and "Img2T" curves respectively (Figures 4.12 and 4.13).

It is noticeable that the performances via NMF learning without the linguistic filtering function are more efficient at first in both cases. This is due to the fact that the statistical filtering performed by TF-IDF is not efficient with few samples and thus leads to erroneous keyword selections. However, the performance by applying NMF with TF-IDF continues to improve until the score of almost 100 could be reached, while the NMF learning without using TF-IDF stagnates or even reduces a little when it comes to the end of the incremental learning. This demonstrates that TF-IDF is efficient when enough samples (i.e., over 40) are available.

We now illustrate the performance of the automatic threshold selection strategy proposed in Equation 2.1 in Section 2.2. Figure 4.14 illustrates the evolution of the IDF values against the number of input training samples. We report the *mean*, *variance*, *maximum* as well as *minimum* of IDF values for the set of all words contained in the training samples. We also show the IDF thresholds idf_{low} and idf_{high} computed from Equation 2.1 (with legends of "Max-thresh-idf" and "Min-thresh-idf" respectively). In order to have a clear feedback

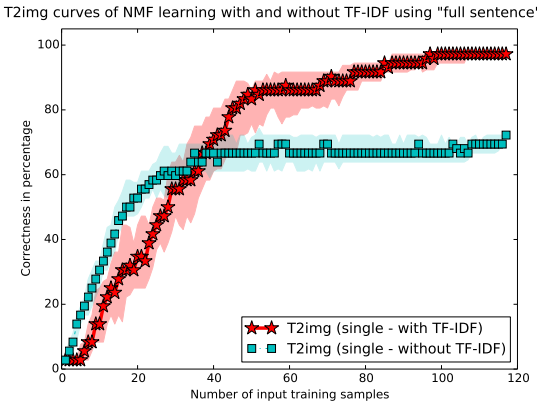


Figure 4.12: "T2img" performances of NMF model: with TF-IDF vs. without TF-IDF by using *FS&S* data from *S-OBJ-C*

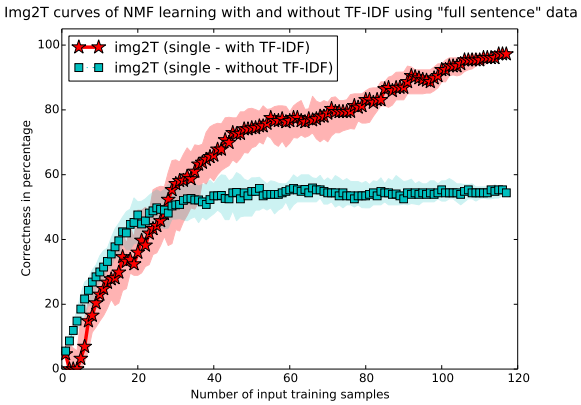


Figure 4.13: "Img2T" performance of NMF model: with TF-IDF vs. without TF-IDF by using *FS&S* data from *S-OBJ-C*

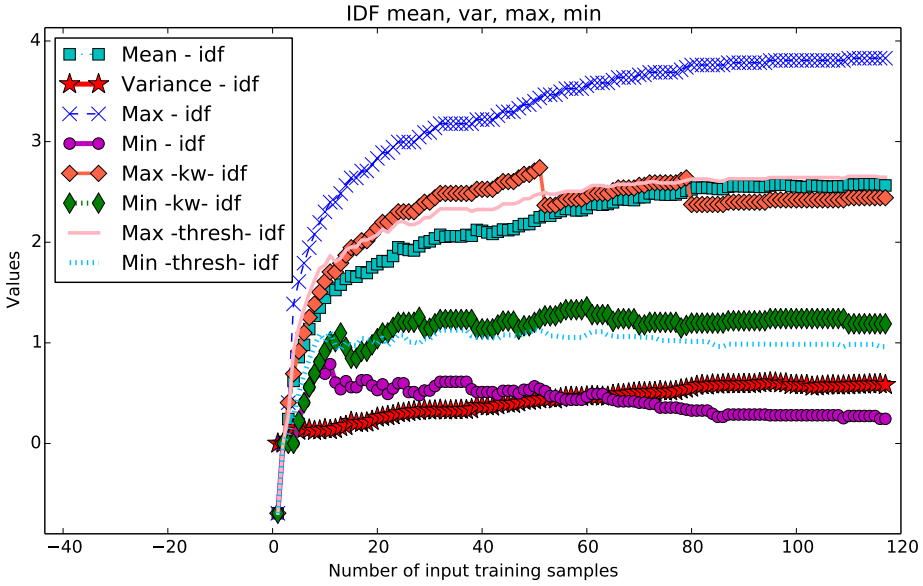


Figure 4.14: Evolution of the IDF values during learning. See text for details.

about whether the calculated thresholds are effective or not, we also draw the maximum and minimum IDF values for all keywords that have appeared up to the current learning step (marked by "Max-kw-idf" and "Min-kw-idf" respectively).

We used $\eta_{high} = 0.67$ and $\eta_{low} = 0.20$ in Equation 2.1 which lead to the fact that the curves of "Max-kw-idf" and "Min-kw-idf" converge in the area between "Max-thresh-idf" and "Min-thresh-idf" as expected. The initial behavior of the IDF threshold which incorrectly selects keywords before step 50 explains in a large part the better performances of learning without TF-IDF observed in Figures 4.13 and 4.12.

As for the TF threshold, which is based on the word occurrences in one cluster (serving as a document), we used the following adaptive policy:

a) when samples are few in one cluster, more specifically when the maximum value of word occurrence is not more than 1, all words will pass the TF filtering due to the fact that the TF as a criterion is unable to distinguish keywords from the rest in this situation;

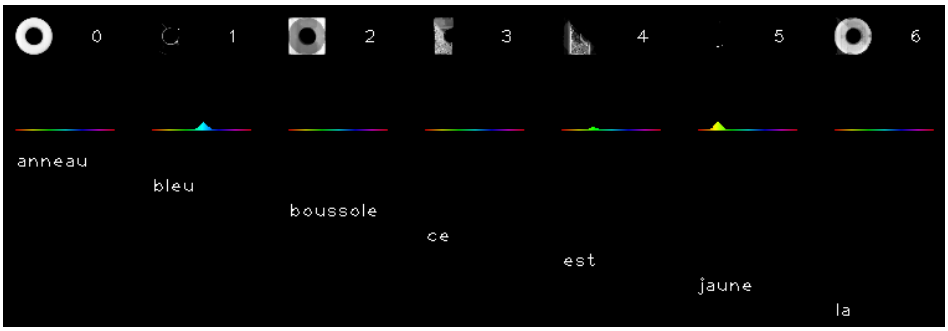
b) when samples become more and more, that is when the maximum value of word occurrence is over 1, we set the TF threshold as the proportion value of the number of samples belonging to a cluster against the sum of word occurrences regarding all samples in this cluster.

The general principle about devising the parameter setting is that it would be preferred to have extra unrelated words than loosing some real keywords, because the later case would strongly ruin the performance.

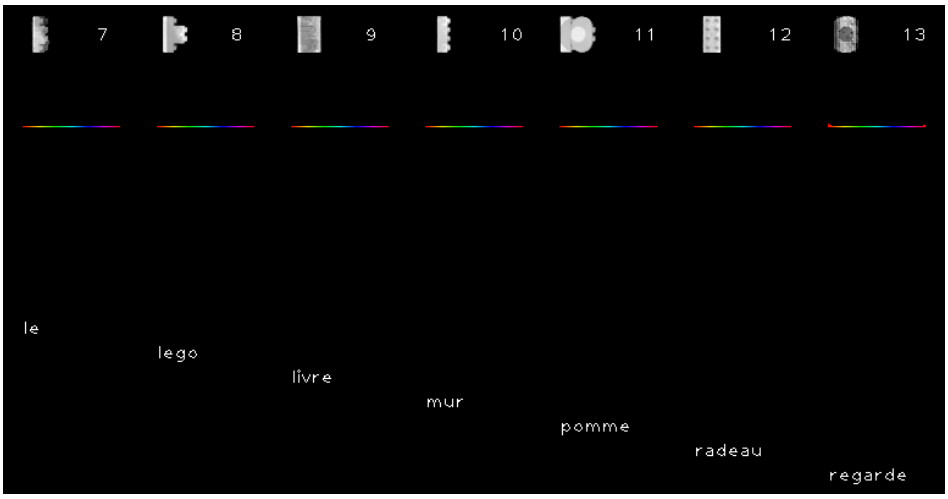
We finally visualize the results of NMF learning with or without using TF-IDF in Figure 4.15 and 4.16.

In both figures, thanks to the proposed NMF model (see Section 2.3), all keyword components as well as some unrelated word components manage to have a "one word-one modality" data description. In Figure 4.15, although there are five words like "ce", "est", "la", "le", "voila" with very common grammatical functions, which fail to be filtered out, the performances seem not to be influenced and the remaining words have correct definitions. In Figure 4.16, it is obvious first of all that due to the existence of too many unimportant components, the modal description of some principal components (eg. "boussole") are somehow damaged; besides, some unimportant components appears to have very similar or even the same modal description (eg. "abel", "avez-vous" and "bleu"; "aimes", "avec" and "belle") which coincidentally leads to the phenomenon of synonym, will have an direct misleading effect during testing by replacing the role of real keyword components; further more, some unimportant components are associated with mixed modal descriptions (eg. "a" and "as").

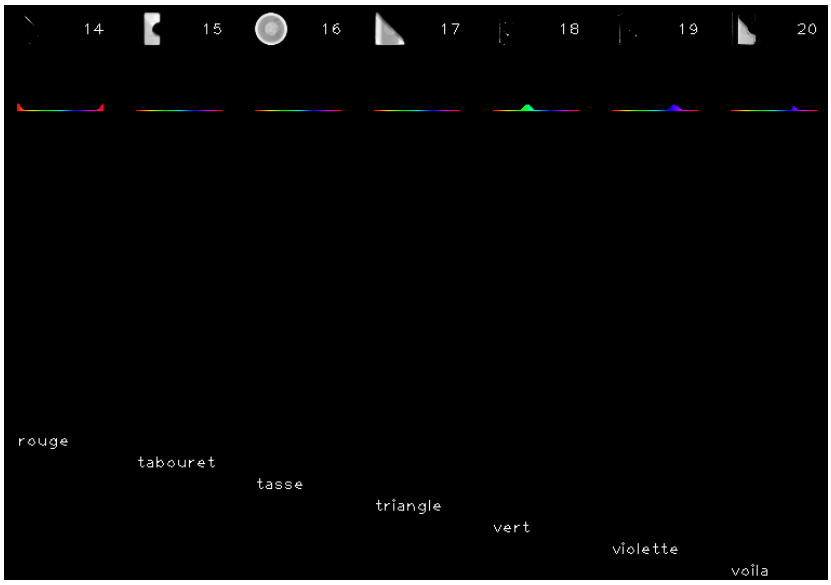
Obviously, there are stronger ambiguities in this experimental scenario which has already come closer to the real life communication scenario, however, by applying our proposed TF-IDF with well tuned parameter settings, learning via NMF still succeeds in making progress incrementally compared to the result when TF-IDF is not applied.



(a)

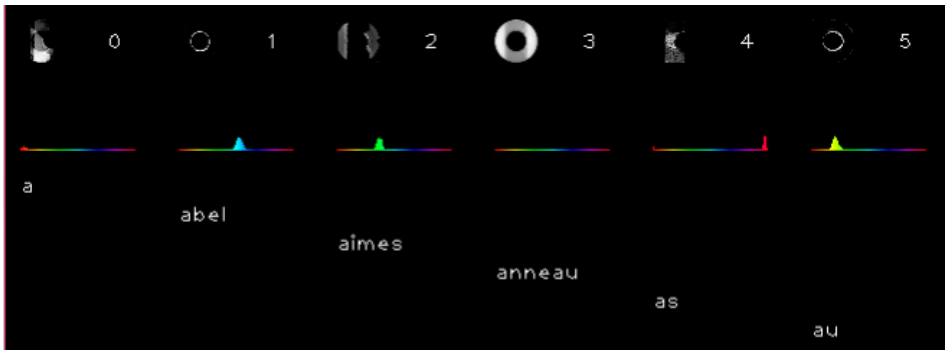


(b)

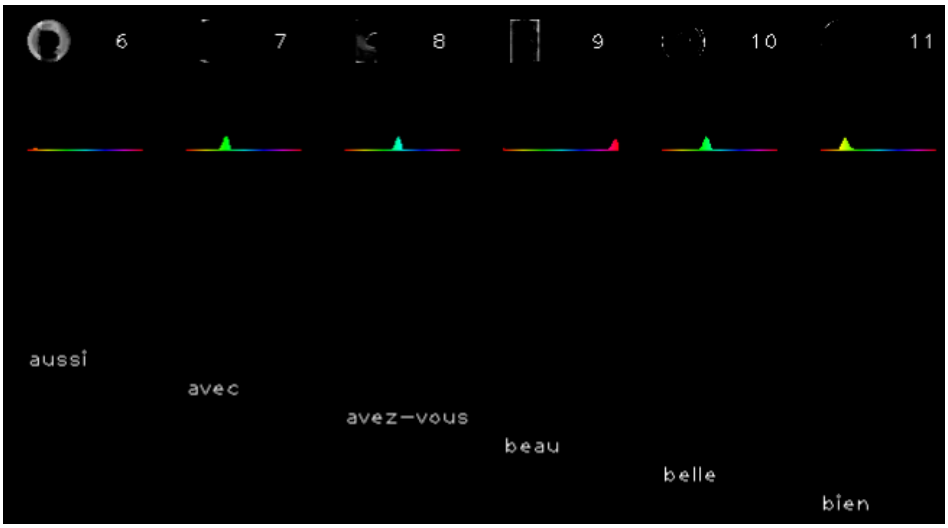


(c)

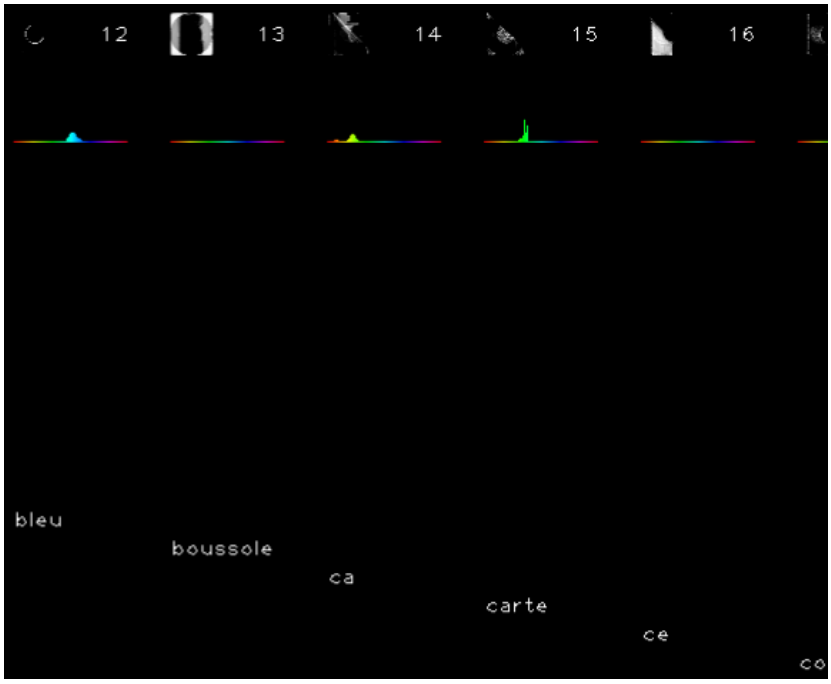
Figure 4.15: Example of the learned dictionary after NMF learning using TF-IDF.



(a)



(b)



(c)

Figure 4.16: Partial example (on a total of 75 elements) of the learned dictionary after NMF learning without using TF-IDF.

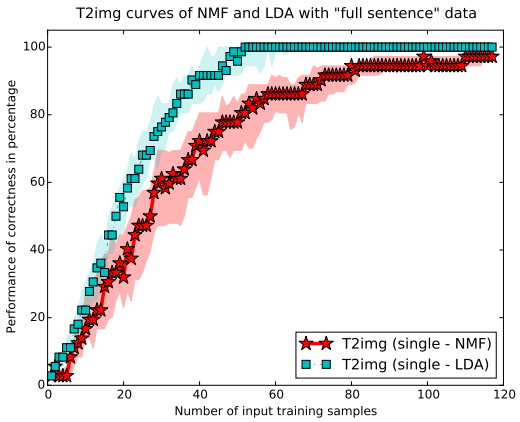


Figure 4.17: "T2img" performance of incremental learning with *FS&S* data from *S-OBJ-C*

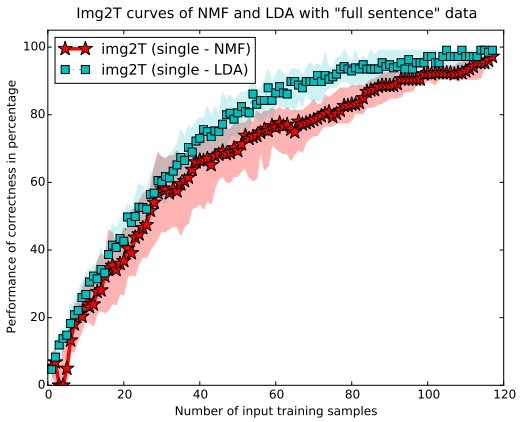


Figure 4.18: "Img2T" performance of incremental learning with *FS&S* data from *S-OBJ-C*

Now we come to the temporary conclusion that NMF with the help of TF-IDF is able to deal with linguistic ambiguity during incremental learning, and we now turn to compare its learning power with that of another model, LDA, in the same scenario, using the same data and fulfilling the same tasks.

4.5.3 Comparison of NMF with LDA

Similar to the settings of the previous experiment we use the same 117 training samples described by full sentences and 36 testing samples as before, the experiment has been launched 50 times and the results are illustrated below, which will be mainly compared with Figure 4.5 and 4.6 in the "keywords only" case.

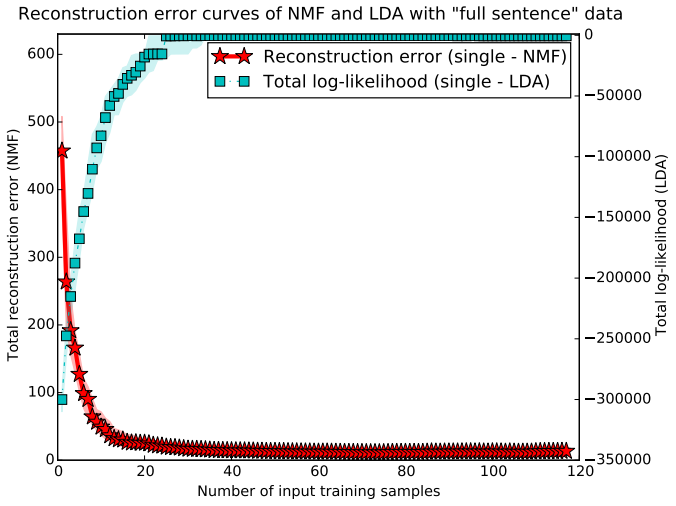


Figure 4.19: Reconstruction error curves of incremental learning with *FS&S* data from *S-OBJ-C*.

Due to the extra ambiguities in linguistic part compared to that of “keywords only", the performances are not guaranteed at the end of training to reach exactly 100%, especially in the curve of NMF learning. Contrary to the previous scenario using only keywords, LDA learns much faster than NMF coupled with the statistical TF-IDF filtering and achieves higher final performances. Besides, it is obvious that the fluctuation in the curves concerning LDA is lowered compared to NMF's performances, indicating that the NMF model seems more sensitive to the changes of corpus of descriptive words which will be filtered by TF-IDF. Lastly, in Figure 4.19, we also find that faster convergence in the criteria of reconstruction error is achieved by LDA than NMF.

In short, the results illustrate the better adaptation of the probabilistic model of LDA to this problem compared to NMF which requires a more complex pre-processing. We think this comes from the fact that some hidden topics capture the “noise” resulting from the words which are not considered as keywords. Indeed, we observe that, at the end of the training, some topics are linking a feature to a word, when some others just correspond to mere words (and no features). For NMF, on the contrary, it is almost impossible for TF-IDF to conduct a strict filtering of sentences with linguistic noise, hence unlike the case in Section 4.4 the NMF components are always redundant as well, containing noise items. What's more, in a noise item, whatever the feature description could be, it just corresponds to a single word, which makes it more difficult to absorb the extra noise.

4.6 Learning with ambiguous noisy data

Going further from the previous experiment with only a single object described each time, we simulate a scenario in which a teacher would present multiple (ie. 2 or 3) objects while describing them using full sentences in general without further referring intention (corresponding to the FS&D and FS&T scenario of Section 2.1.4). In this case, the learner will not only deal with linguistic ambiguity as demonstrated in the previous section but the referential ambiguity as well. Therefore, inheriting the same settings and data as in Section 4.5.3, the following curves add cases when “double” and “triple” data are utilized.

Due to further ambiguities in terms of referring, learning by presenting multiple objects each time does not lead to the full performance of 100% in the end, which is especially obvious in the “triple” case. Besides, as visualized in Figure 4.20 and 4.21, the impact of referential ambiguity reduces the learning speed and quality as well as the speed of convergence of reconstruction error with gradual effect from “single”, “double” to “triple”.

However, another perspective of interest to observe in Figure 4.20 and 4.21 is to pay attention to overall performances, that is to say to analyze the “single”, “double” to “triple” learning curves together, of NMF and LDA respectively. Obviously, the three learning curves of NMF appear mutually much closer than those of LDA, in other words NMF is less influenced by the changes of referential ambiguity thus rendering less variance concerning the incremental performances, which can be pointed out conspicuously by the fact that NMF performs even better than LDA in “Img2T” using triple data. Imagining that the learner is given a quadruple

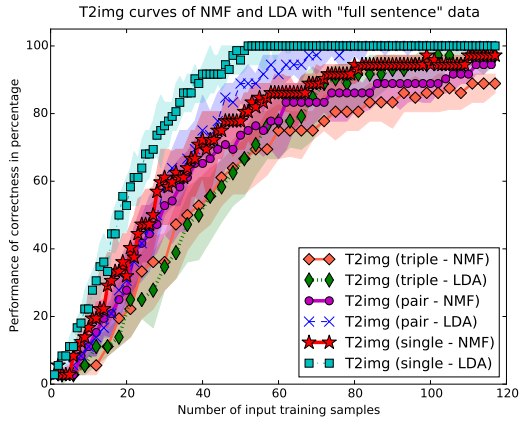


Figure 4.20: "T2img" performance of incremental learning with *FS&S*, *FS&D* and *FS&T* data from *S-OBJ-C*

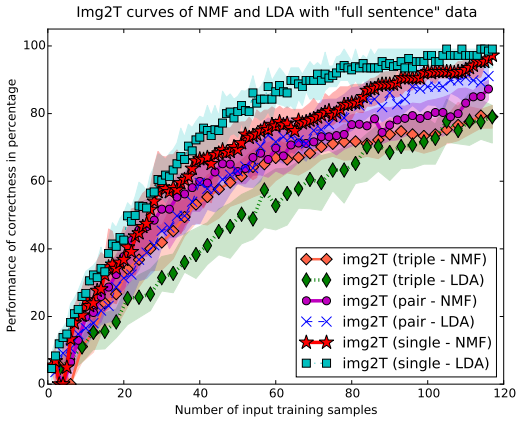


Figure 4.21: "Img2T" performance of incremental learning with *FS&S*, *FS&D* and *FS&T* data from *S-OBJ-C*

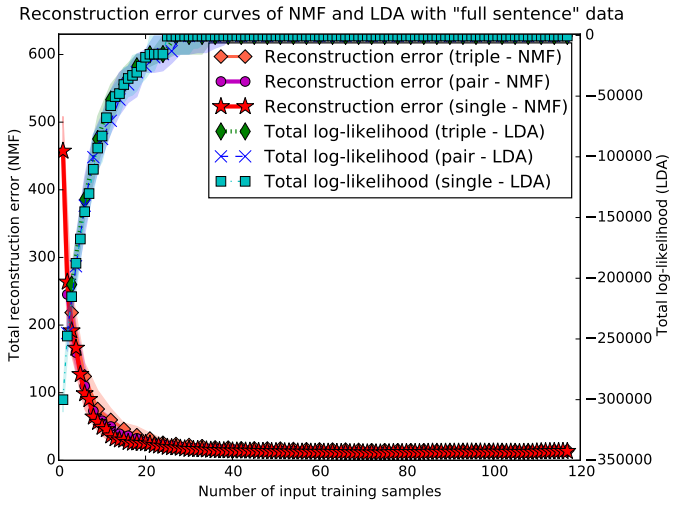


Figure 4.22: Reconstruction error curves of incremental learning with *FS&S*, *FS&D* and *FS&T* data from *S-OBJ-C*.

of (ie. four) objects each time, there possibly might be a reverse trend that NMF would surpass LDA in both "T2img" and "Img2T". Possible explanation could be from the quality of samples: if each time the samples given contain too much ambiguities, the statistical model of LDA would be shaped and refined less significantly, however, NMF exhibits comparatively better capability of extracting common factors directly from cross-situational raw data, therefore demanding less samples to have its model adequately trained.

It is also noticeable that the performance curves in "Img2T" are generally lower than those in "T2img". As pointed out in Section 4.4, this is once again linked to the fact that the "T2img" evaluation is more tolerant toward errors in the learned dictionary.

Up to now, we have studied how different kinds of ambiguities impact the performance in the incremental learning experiments with a random choice of training samples. In the next part we will study if an active sample selection strategy can be devised to accelerate the incremental learning of word-meaning association.

4.7 Learning with active sample selection

In this section, we simulate an incremental scenario in which a learner actively chooses object(s) to learn, which will be then described by a teacher either with related keywords only or with full sentences. Like previous experiments, the incremental learning has limited learning steps (eg. 117, 58 and 39 for "single", "double" and "triple" cases respectively) and the experiment settings and data remain the same as those in the previous section, however, the learner could reuse the training samples which have been chosen, in other words all the 117 training samples are considered for the choice of the next sample using either the random or active strategy. This is made to highlight the ability of active learning to efficiently ignore the already known samples.

As described in Section 2.5, two active learning strategies will be applied to help the learner accelerate its learning progress : MRES (maximum reconstruction error based selection) and CBE (Confidence base exploration). We will study their performances for both "T2img" and "Img2T" tasks, with NMF (by default, TF-IDF is applied in all situations in the rest of this thesis) and LDA respectively which will be compared with the strategy of random choice of objects. Similar to the ambiguity settings of the experimental data in previous sections, there are "single", "double" and "triple" cases using data of "keywords only" and "full sentence" respectively. In order to have a quantitative comparison of the different curves, we compute the area under the curves and use this as a global measure of learning performance (see Section 4.7.3 for details). We use $\boldsymbol{S}$, $\boldsymbol{D}$ and $\boldsymbol{T}$ in short for "single data", "double data" and "triple data" from *S-OBJ-C*, and *KW* and *FS* represent scenarios of "keywords only" and "full sentence". We use notation in the form:

$$Perf_{MRES-NMF}^{Img2T-KW-S}$$

in order to designate the performance for the MRES active strategy, applied with the NMF

algorithm, on the "keywords only" and single object data, evaluated on the "Image to Text" task, relative to the overall best performing approach.

Note that in order to enhance the comparability among all cases, we deliberately let initial samples (at the first learning step) be the same. For all curves, the simulation experiment was performed 50 times in total.

4.7.1 Active learning vs. random learning by applying NMF

4.7.1.1 Learning with "keywords only" data

In this first experiment, we compared the performances of NMF with "keywords only" data. From Figure 4.23, we observe that the "Img2T" performance (on the left half) is generally a little bit inferior to that of "T2img", whose possible reasons have been addressed in previous sections, yet they both indicate almost the same trend in terms of the progress of incremental learning by applying the different strategies.

In terms of the effects of active learning, first of all, the superiority seems quite conspicuous in terms of the speed of progress and the overall quality when "single" data are used, especially for MRES. Regarding the area under the curves (see full results in Table 4.1 and 4.2), this improvement can be quantified as:

$$Perf_{MRES-NMF}^{T2img-KW-S} - Perf_{random-NMF}^{T2img-KW-S} = 0.036$$

As for CBE, it still shows faster learning speed than that of random learning although not as good as that of MRES, with the overall quality as:

$$Perf_{CBE-NMF}^{T2img-KW-S} - Perf_{random-NMF}^{T2img-KW-S} = 0.01$$

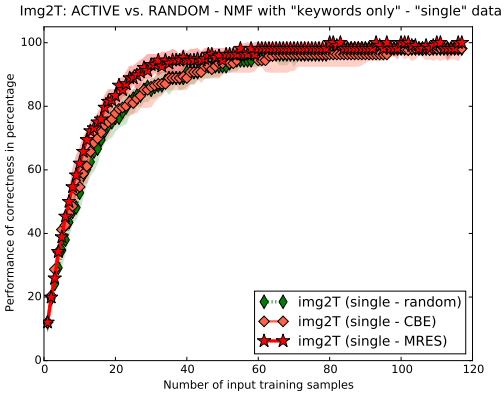
However, when referential ambiguities increase (ie. by using "double" and "triple" data), both active learning and random learning exert the performances almost the same level, where MRES and CBE appear slightly lower than random:

$$Perf_{MRES-NMF}^{T2img-KW-T} - Perf_{random-NMF}^{T2img-KW-T} = -0.009$$

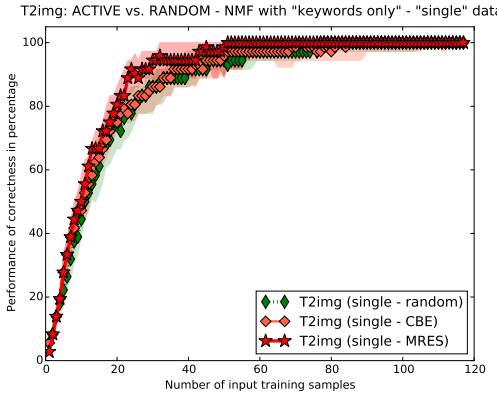
$$Perf_{CBE-NMF}^{T2img-KW-T} - Perf_{random-NMF}^{T2img-KW-T} = -0.009$$

4.7.1.2 Learning with "full sentence" data

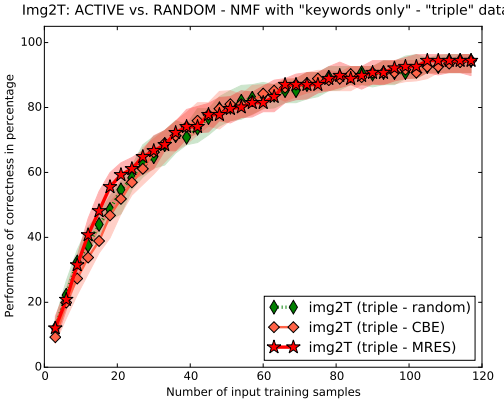
We now compare the performances of NMF with "full sentence" data. In Figure 4.24, we can see that the performances obtained with different strategies now consistently place MRES as the best method, above CBE, and that these two methods are above the random choice. This



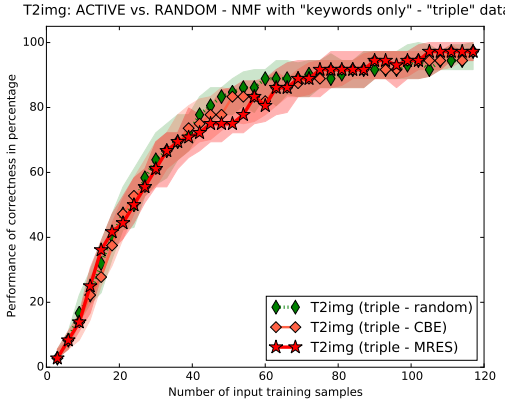
(a) "Img2T" results using $KW\&S$ from *S-OBJ-C*



(b) "T2img" results using $KW\&S$ from *S-OBJ-C*



(c) "Img2T" results using $KW\&T$ from *S-OBJ-C*



(d) "T2img" results using $KW\&T$ from *S-OBJ-C*

Figure 4.23: Active learning vs. random learning - comparisons in performances between MRES, CBE and random choice by applying NMF with "keywords only" data.

can be seen in the area under curves which show a significant difference:

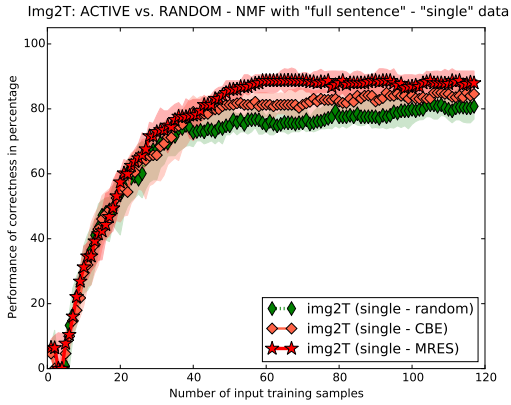
$$Perf^{T2img-FS-T}_{MRES-NMF} = 0.67$$

$$Perf^{T2img-FS-T}_{CBE-NMF} = 0.61$$

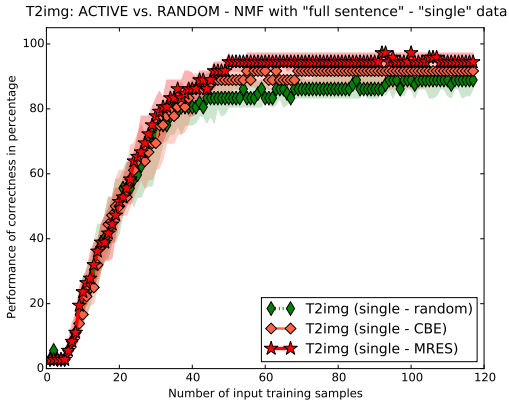
$$Perf^{T2img-FS-T}_{random-NMF} = 0.527$$

for instance.

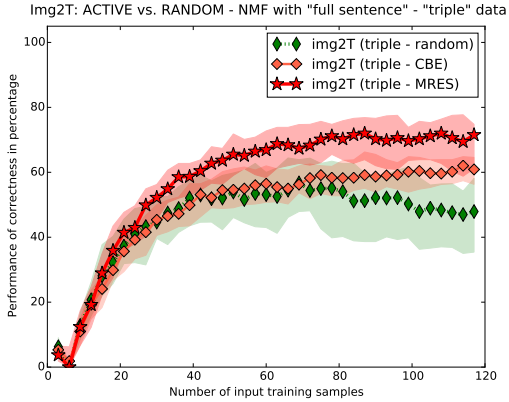
In these results, although active learning strategies play a role, we can not ignore the effect of TF-IDF. Indeed the optimized parameter setting (explained in Section 4.5.2) cannot accommodate all possible sequence of learning samples. This is particularly true for the random choice strategy where a lot of samples are used multiple times in this scenario (see further discussion in Section 5.3) while for MRES and CBE we could find comparatively the best parameters in terms of thresholds that lead to the results illustrated in the above images.



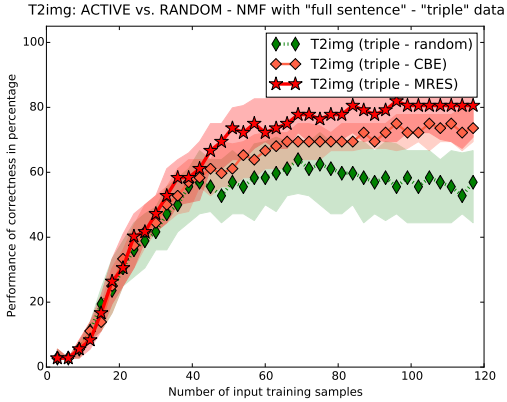
(a) "Img2T" results using $FS\&S$ from *S-OBJ-C*



(b) "T2img" results using $FS\&S$ from *S-OBJ-C*



(c) "Img2T" results using $FS\&T$ from *S-OBJ-C*



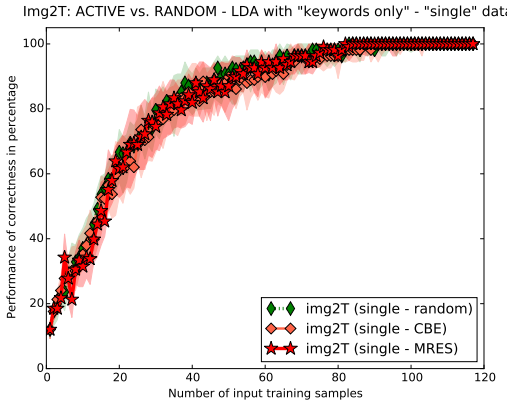
(d) "T2img" results using $FS\&T$ from *S-OBJ-C*

Figure 4.24: Active learning vs. random learning - comparisons in performances between MRES, CBE and random choice by applying NMF with "full sentence" data.

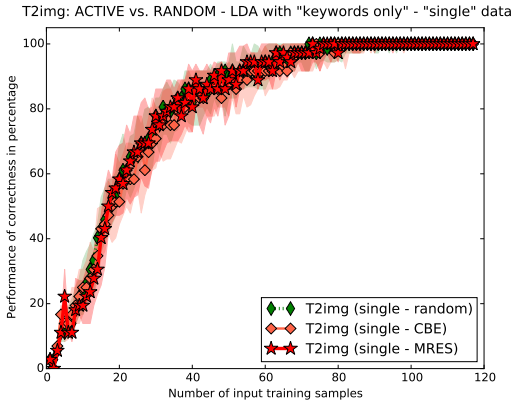
4.7.2 Active learning vs. random learning by applying LDA

4.7.2.1 Learning with "keywords only" data

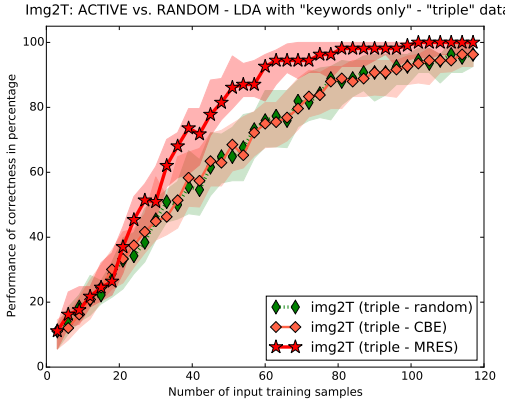
We now study the effect of active learning on LDA, with "keywords only" data. As shown in Figure 4.25, contrary to what has been observed for NMF (Figure 4.23 in Section 4.7.1.1), there is no marked difference in performances between the "Img2T" and "T2img" evaluation method.



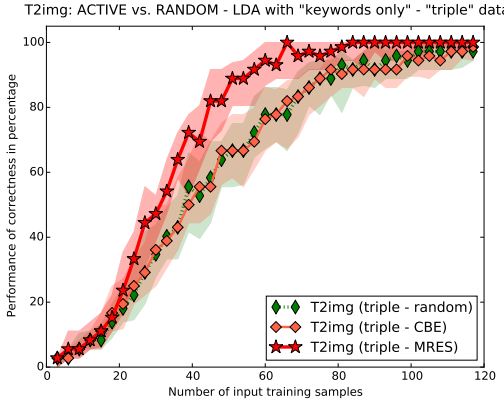
(a) "Img2T" results using $KW\&S$ from *S-OBJ-C*



(b) "T2img" results using $KW\&S$ from *S-OBJ-C*



(c) "Img2T" results using $KW\&T$ from *S-OBJ-C*



(d) "T2img" results using $KW\&T$ from *S-OBJ-C*

Figure 4.25: Active learning vs. random learning - comparisons in performances between MRES, CBE and random choice by applying LDA with "keywords only" data.

However, with LDA, the more ambiguous the referring situation is, the larger the difference between the active learning and random learning. This difference is particularly visible in the triple scenario, where MRES clearly outperforms CBE, which is very close to the random choice (see Table 4.1 and 4.2 for complementary data):

$$Perf^{T2img-KW-T}_{MRES-LDA} - Perf^{T2img-KW-T}_{random-LDA} = 0.108$$

$$Perf^{T2img-KW-T}_{CBE-LDA} - Perf^{T2img-KW-T}_{random-LDA} = -0.002$$

4.7.2.2 Learning with "full sentence" data

With the "full sentence" data, similar effects of superiority of active learning (especially for MRES) over random learning can be observed in Figure 4.26. Compared to what have already been observed in Figure 4.25, this gain is more evident with the increase of referential

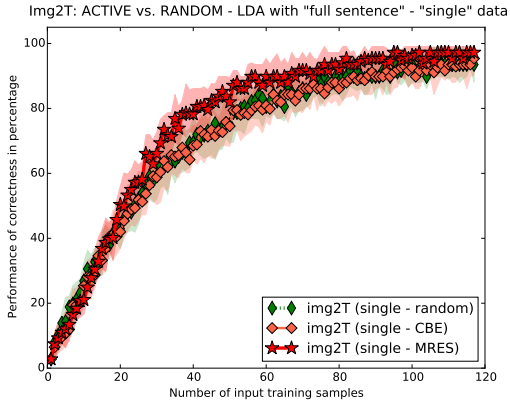
ambiguity as below:

$$Perf_{MRES-LDA}^{T2img-FS-T} = 0.801$$
$$Perf_{CBE-LDA}^{T2img-FS-T} = 0.672$$
$$Perf_{random-LDA}^{T2img-FS-T} = 0.674$$

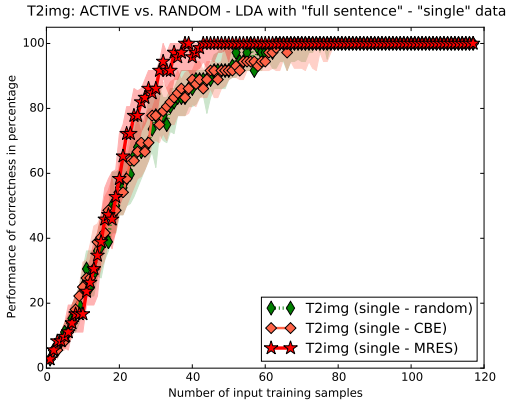
for instance.

Yet in the current situation where linguistic noise is added, this effect also happens in "single" case with the least referential ambiguity:

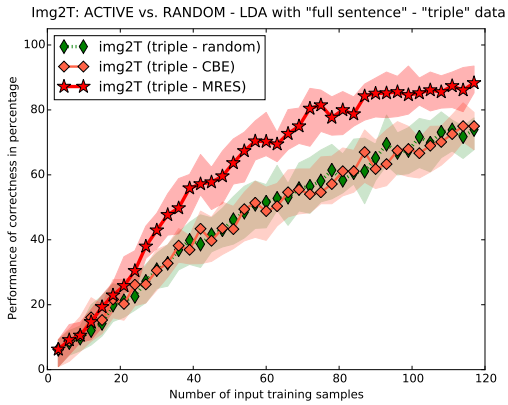
$$Perf_{MRES-LDA}^{T2img-FS-S} = 0.945$$
$$Perf_{CBE-LDA}^{T2img-FS-S} = 0.906$$
$$Perf_{random-LDA}^{T2img-FS-S} = 0.906$$



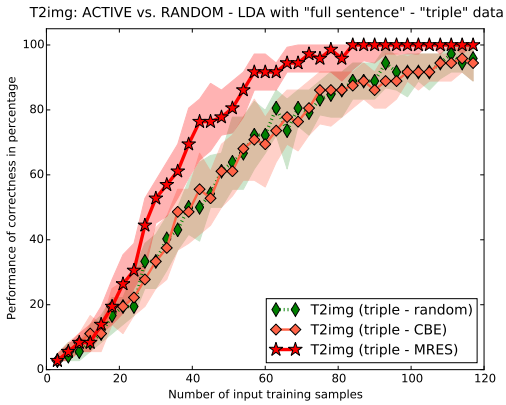
(a) "Img2T" results using *FS&S* from *S-OBJ-C*



(b) "T2img" results using *FS&S* from *S-OBJ-C*



(c) "Img2T" results using *FS&T* from *S-OBJ-C*



(d) "T2img" results using *FS&T* from *S-OBJ-C*

Figure 4.26: Active learning vs. random learning - comparisons in performances between MRES, CBE and random choice by applying LDA with "full sentence" data.

4.7.3 Discussion regarding the overall performances

In order to have a global comparison of performances of all case, we establish Table 4.1 and 4.2 that show the relative performance (area under the curves) in different cases by using different learning models and data. The reason why we call the performance value relative is because we have all values normalized, by means of being divided by the the maximum value of all cases.

Img2T / *Strategy* / *Case*		**MRES**			**CBE**			**random**		
		S	***D***	***T***	***S***	***D***	***T***	***S***	***D***	***T***
NMF	*KW*	1	0.901	0.829	0.966	0.872	0.816	0.964	0.882	0.82
LDA		0.915	0.873	0.828	0.91	0.838	0.723	0.926	0.834	0.723
NMF	*FS*	0.827	0.686	0.631	0.784	0.675	0.533	0.746	0.617	0.497
LDA		0.844	0.744	0.667	0.784	0.645	0.508	0.795	0.635	0.51

Table 4.1: Relative performance values of "Img2T" in all cases. See text for legend.

T2img / *Strategy* / Case		**MRES**			**CBE**			**random**		
		S	***D***	***T***	***S***	***D***	***T***	***S***	***D***	***T***
NMF	*KW*	0.996	0.895	0.802	0.97	0.866	0.802	0.96	0.884	0.811
LDA		0.8965	0.855	0.805	0.888	0.815	0.695	0.9018	0.813	0.697
NMF	*FS*	0.873	0.741	0.67	0.84	0.747	0.61	0.807	0.682	0.527
LDA		0.945	0.87	0.8007	0.906	0.798	0.672	0.906	0.801	0.674

Table 4.2: Relative performance values of "T2img" in all cases. See text for legend.

First of all, some simple conclusions can be drawn from direct observation of the above tables. For example, with the increase of referential ambiguity (ie. using data from ***S***, ***D*** to ***T***), the performance decreases under the same scenario as well as using the same strategy; Besides, NMF performs better almost always than LDA in *KW* scenario, however, LDA outperforms NMF in *FS* scenario, which corresponds to the conclusion about the different abilities of dealing with linguistic ambiguities between NMF and LDA as described in Section 4.5.3 and 4.6. In addition, using *KW* data, all experiments generally achieve better results in the test of "Img2T" than "T2img" while *FS* data would help the learning attain better performances in "T2img".

4.7.3.1 Comparing performances of *KW* and *FS*

It seems obvious that the performances using keywords only (*KW*) should surpass those of using full sentences (*FS*) which is more ambiguous in terms of linguistics, and in reality it is the case in Table 4.1. However, in Table 4.2, exceptions can be found in the performance of LDA learning, using "single" data while applying MRES, CBE and random choice respectively and using "double" data while applying MRES only. In fact, similar phenomena can be observed

if we trace back to compare Figure 4.5 and 4.20 even when random learning is conducted with "single" data.

The explanation is quite possibly related to the policy of determining the number of topics of LDA learning in Section 4.3: on one hand, there should be little difference concerning the evolution of topic numbers in cases of KW and FS respectively, because such numbers depend directly on the number of VQ clusters and the log likelihood of the feature part rather than the linguistic part; on the other hand, in practice, small amount of redundant topics are added to improve the results of LDA learning. In reality, by checking the midterm learned dictionary of topics by applying LDA, we found in both KW and FS cases that there are noise types in learned topics (especially the redundant ones) as follows: either multiple words corresponding to a vision feature, a word associating a vacant meaning or a wrong feature. The difference between KW and FS resides in the fact in FS it is almost always the noisy words that take part in the above ill-matching topics however in KW it is the keywords that play the role. In consequence, when performing "T2img" tests, the learner would be much more misled by the ill-matching topics in the KW case because a word label (as a keyword indeed) could just indicate a wrong feature, yet on the contrary in FS, if an ill-matching topic is labeled just by a noisy word, it won't be involved in "T2img" at all, which alleviates the misleading effect and thus helps to achieve better performances.

A short conclusion could be drawn from the above analysis: on one hand, in order to build a refined statistical model, it is necessary to have sufficient samples, otherwise the misleading effect would take place as we observe in KW case; on the other hand, the noisy words in FS case of LDA learning effectively helps to absorb the noise by replacing the role of keywords in the formation of ill-matching topics and in turn improves the learning performance of "T2img".

4.7.3.2 Comparing performances of MRES and CBE

Based on previous analyses of visualized results in Section 4.7.1 and 4.7.2 along with the data from the above two tables, MRES's performance is observed conspicuously better than that using random choice learning strategy in the majority cases. As for CBE, although its superiority over random seems not so strong as that of MRES, it could still distinguish itself from random based on the positive gap in terms of relative performance value, however small it is, in most cases (despite 9 exceptions among all 24 comparisons). Therefore, we come to the general conclusion that both models of MRES and CBE are proved valid in elevating the learning progress and quality, and even in some specific experimental settings such improvements are quite large. Concerning the comparison between the two active learning strategies, obviously MRES appears more effective than CBE.

Besides the fact that the two active learning strategies presented in Section 2.5.1 and 2.5.2 rely on different principles, in practice, we have to tune parameters carefully so as to exhibit as much its potential as possible. For example in MRES, the slacking index is adopted and this has been observed effective in improving the performance in active learning

via NMF using data of more ambiguities (ie. "D" or "T"). However, even without the slacking strategy, MRES can still work in the model of LDA, which remains effective in outperforming random choice strategy despite the increase of referential ambiguities. In CBE, however, finding the correct confidence threshold, whose optimized value differs according to different experimental scenarios, is essential to obtain good performances, showing a greater sensibility of this method to parameter settings.

4.8 Learning with more complex visual features

Due to the limitations of applying pixel as the shape descriptor, mainly in terms of shape reconstruction with unexpected combination of shape components (see more discussion in Section 5.4), we found interesting to extend the experiment settings by using more complex visual features (we choose the HOG descriptor) to further validate our proposed learning models.

To some extent, HOG could be regarded as pixel descriptor however with orientations, which is capable of recording more subtle information such as texture and interior pattern on the surface of an object. The following experiments, will make use of HOG data of "T1" and "T3" from *S-OBJ-D* as described in Section 4.1.0.3.

4.8.1 About the sensitivity to feature quantification

Compared to using pixel directly as the shape descriptor, HOG is more complex in its formation. In our applications, two vital procedures turn out to be quite sensitive to the performance of HOG as a way to describe shapes: *vector quantification* (*VQ*) for LDA and *VQ clustering* for NMF, which we have detailed in Section 2.1.3 about the feature quantification.

Initially, like what have been applied in previous experiments which use pixel to describe shape, we used the incremental clustering to group samples for TF-IDF as well as for the VQ for LDA. However, despite its described merits, it also has two main drawbacks: 1). the clustering result is directly influenced by the order in which the samples are presented, because a new sample will try to be compared one by one by using χ^2 distance with all existed clusters derived from all previous samples, and as a result, samples that appear first have the priority to form cluster centers; 2). the clustering result is usually not the global optimum for the reason that once a sample has been ascribed to a cluster, it will never belong to a new cluster, regardless that the latter one might show closer similarity.

Consequently, when HOG data were used, we found systematic loss of performances both via NMF and LDA because of the wrong clustering. As a solution, we resorted to the traditional K-Means clustering[2] whose performance is centroids-oriented rather than order-dependent. The difficulty of applying K-Means in an incremental learning scenario is that

[2]http://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

the number of centroids should be manually determined. It was resolved in our enclosed environment of *S-OBJ-D* because we knew the correct number of shape centroids (ie. 20), color centroids (ie. 6) as well as the clustering concerning both shape and color of all samples (ie. 120). We adopted this solution since our current objective is to analyze the best possible performance, knowing that it would not be feasible in an open environment in which new objects are unknown in terms of property and number, which is left for future work.

In order to illustrate the role the quality of feature quantification plays in the incremental learning using HOG data, the performances by adopting K-Means clustering are comparatively displayed below with those obtained by applying the original methods of incremental clustering. Note that, since the latter type of models is doomed to be sub-optimal, we just let it run 5 times so as to get the percentile curves without the pursuit of higher precision while the learning experiment using K-Means clustering, which possibly appears optimal, had been conducted 50 times before the final curves of higher quality concerning statistical representation were generated. In reality, such treatment of data will definitely not tamper with the essence of the following comparisons.

In the following figures the best performances of all cases are presented after the tuning of parameter settings of both incremental VQ (for LDA) and TF-IDF (for NMF) has been carried out. Note that these parameters are quite sensitive, especially for the use of TF-IDF (for NMF).

4.8.1.1 Learning with noise-free "T1" HOG data

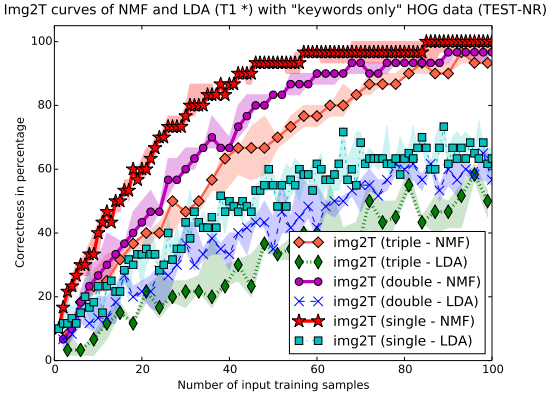
We first illustrate in Figure 4.27 the performances using "T1" data from *S-OBJ-D* in "keywords only" scenario with cases in terms of increasing referential ambiguities from "single", "double" to "triple" (marked by $KW\&S$, $KW\&D$ and $KW\&T$ respectively).

In Figure 4.27, clear at a glance is the difference of performances via LDA learning. Compared to the incremental VQ of HOG and color histograms on which the learning progress will stagnate at a level, approximately 80% for "T2img" and 60% for "Img2T", the k-means VQ could lead to higher performances, close to 100%. Correspondingly, it can be observed that the converging performance of LDA's total log-likelihood using K-Means (see Figure 4.27(f)) fully outperforms the other case (see Figure 4.27(e)) in terms of speed, stability and the final level. As for the performances of NMF, there seems no major differences because when dealing with noise-free data NMF can be applied directly.

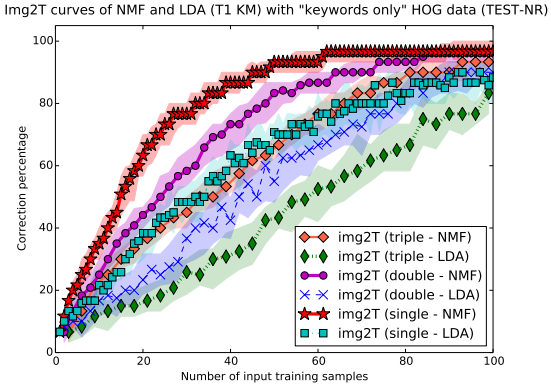
4.8.1.2 Learning with noisy "T1" HOG data

The comparison has also been made with noisy (i.e. Full sentences) HOG data, for which NMF is applied after TF-IDF filtering, and the results are illustrated in Figure 4.28 based on the same experimental settings as in the previous one.

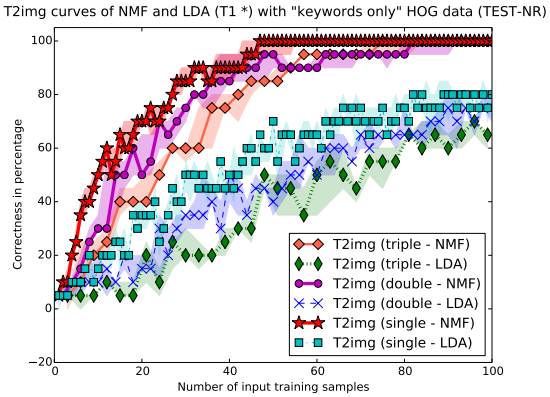
Regarding the performances of LDA learning, like before, when using the incremental



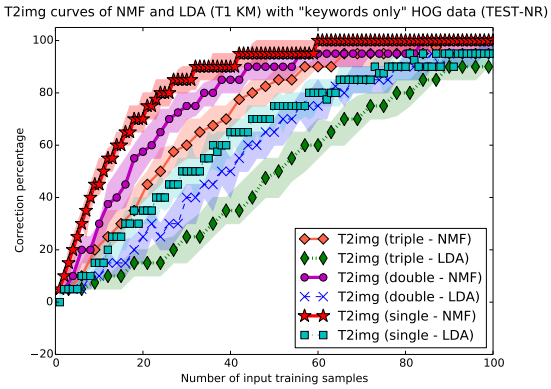
(a) "Img2T" performances by adopting the incremental clustering



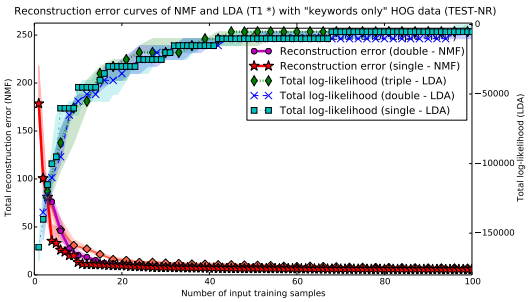
(b) "Img2T" performances by adopting K-Means clustering



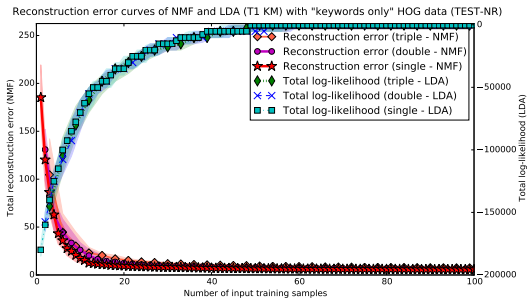
(c) "T2img" performances by adopting the incremental clustering



(d) "T2img" performances by adopting K-Means clustering

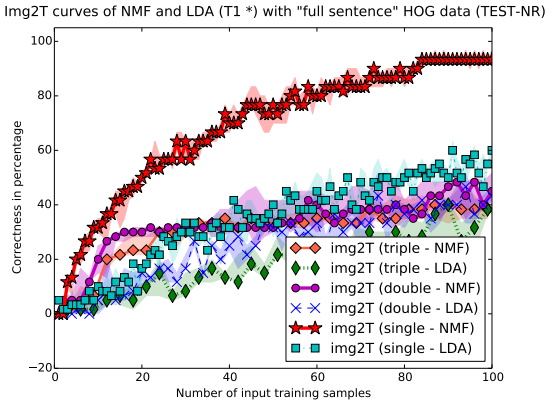


(e) Performances of the evolution of reconstruction errors by adopting the incremental clustering

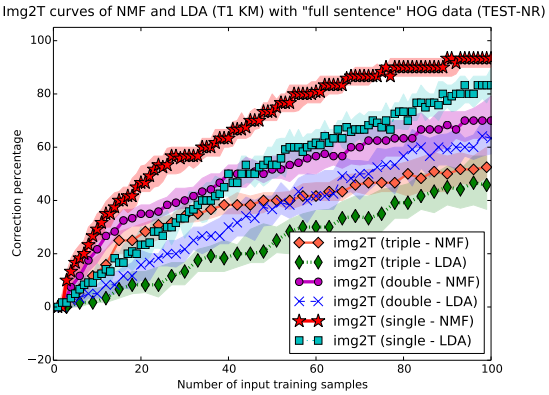


(f) Performances of the evolution of reconstruction errors by adopting K-Means clustering

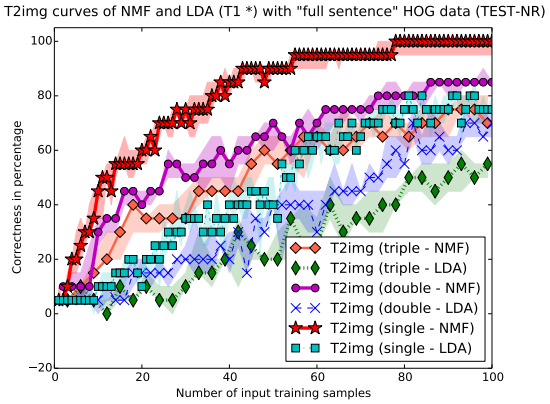
Figure 4.27: The comparison of performances during incremental learning by applying incremental clustering and K-Means clustering respectively in proposed models with "T1" data set in scenarios of $KW\&S$, $KW\&D$ and $KW\&T$ from *S-OBJ-D*.



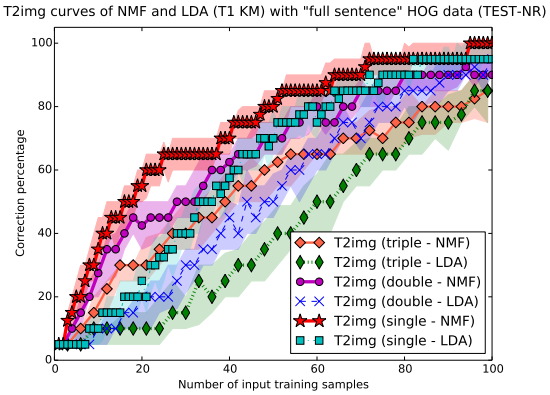
(a) "Img2T" performances by adopting the incremental clustering



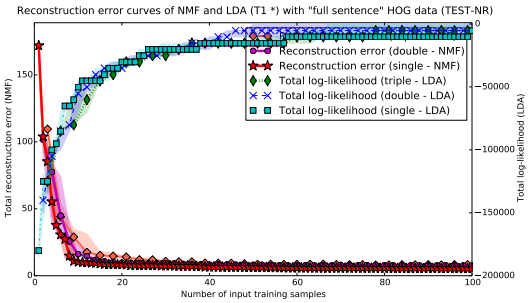
(b) "Img2T" performances by adopting K-Means clustering



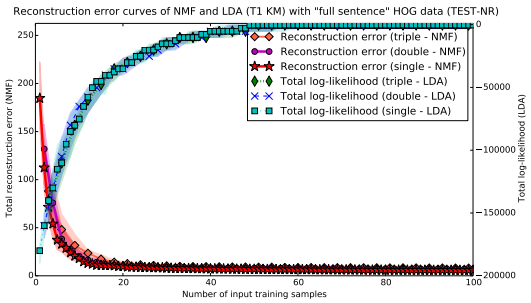
(c) "T2img" performances by adopting the incremental clustering



(d) "T2img" performances by adopting K-Means clustering



(e) Performances of the evolution of reconstruction errors by adopting the incremental clustering



(f) Performances of the evolution of reconstruction errors by adopting K-Means clustering

Figure 4.28: The comparison of performances during incremental learning by applying incremental clustering and K-Means clustering respectively in proposed models with "T1" data set in scenarios of *FS&S*, *FS&D* and *FS&T* from *S-OBJ-D*.

clustering based VQ, LDA performed at a lower level than in the case when the K-Means cluster is applied. However, in this case, the final performance with k-means still remains far from 100%.

As for NMF learning, contrary to what have been illustrated using only keywords in Figure 4.27 where there essentially seems to be no difference between cases, using k-means here improve performances, while not reaching high-levels. This is linked to the fact that a correct clustering is essential to the correct working of the TF-IDF filtering. We should emphasize that we have made special thresholds for TF-IDF using incremental clustering: besides the tuning of η_{low} and η_{high} in Equation 2.1, we used a TF threshold of 0 which means that all words would pass the TF filtering. Normally, when the clustering of samples is of good quality, every cluster encloses more or less the same amount of samples. However, with incorrect clustering, clusters may contain a very different number of samples, which prevents the TF criteria to be efficient. Thanks to the property of NMF that its performance will not decrease noticeably if some extra unrelated words are taken into account as components, removing the TF filtering makes it possible to maintain a limited, but coherent performance for the incremental clustering. K-means, by providing a better clustering allows to keep the TF filtering and reach much better performances.

By comparing the learning performances of our proposed models applying either the incremental clustering or K-Means clustering, we first reinforce the conclusion that the quality of feature quantification plays a vital role in the learning performance of our proposed models. The incremental vector quantification is based on the similarity measure with χ^2 kernel, which worked well for the shape described by pixels, but performs badly in the case of HOG. We attribute this bad performance to the fact that the shape description using HOG can cover more details and appears more sensitive to the condition changes in the experimental scenario than that directly of utilizing pixels. This is a good thing to have data able to represent more objects, but impose the use of a more efficient clustering algorithm.

Although K-Means in which the number of cluster centroids is known before the learning starts was applied, alternative solutions better than our incremental clustering using the χ^2 kernel, which proves not capable of fully presenting HOG information, should be proposed in future in order to exhibit the performances using HOG similar to the incremental learning performance reported in Section 4.4, 4.5, 4.6, and 4.7 using our simple shape descriptor.

4.8.2 About the necessary minimum size for training

We noticed that in all experiments using pixels as shape descriptor, the performances would always converge (either at the level of 100% or a little bit lower) before the learning comes to its end. However, in the experiments using HOG with "T1" data in the previous Section 4.8.1, while using similar number of samples, the learning seemingly fails to converge in some cases. It would be interesting to see their performances when more training samples (ie. "T3" data) are given, and analyze if the imperfection of learning caused by the ill feature quantification could be remedied by giving more training samples.

The results in this section are obtained by replacing "T1" data with those of "T3" and inheriting other experimental settings of the previous section.

4.8.2.1 Learning with noise-free "T3" HOG data

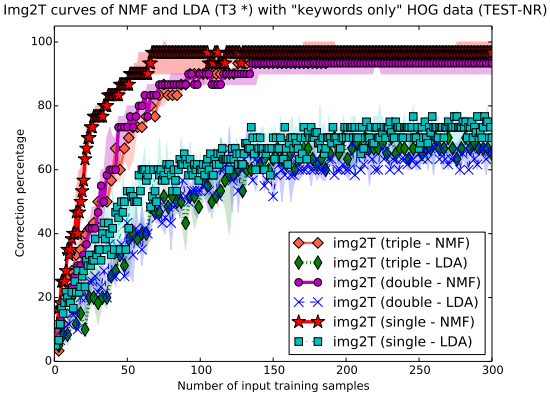
In this experiment, since no linguistic noise is involved, Figure 4.29 shows that NMF learning converges to the final state of 100% quite fast in all cases of "single" (with around 75 training samples given), "double" and "triple" (with around 125 training samples provided).

Regarding LDA, its learning in the previous section with "T1" data hardly converged when the size of training samples is less than 100 as shown in Figure 4.27. In fact, as illustrated in Figure 4.29, during the learning process by using incremental clustering, all cases of "single", "double" and "triple" converge approximately after 150 samples are given for training to their "bottle neck" levels respectively (70% (in "Img2T") and 85% (in "T2img") for "single", 65% (in "Img2T") and 80% (in "T2img") for "double" and "triple"). Moreover, the LDA learning by applying K-Means clustering is able to reach its maximum performance around 95% (in "Img2T") and 97.5% (in "T2img") and also converges when more or less 150 training samples are provided.

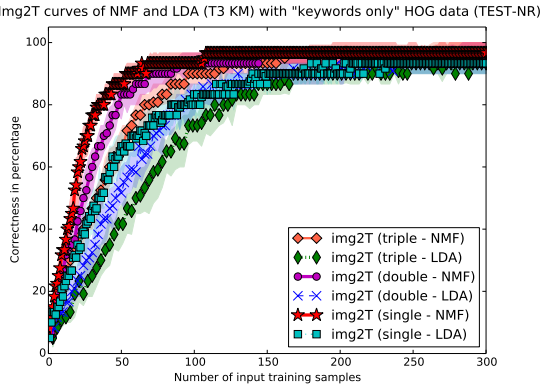
As a result, we can see that the imperfection of incremental learning process caused by the ill VQ could be made up to a limited extent by providing more training data, but reaching a good performance is still dependent on a better clustering approach.

4.8.2.2 Learning with noisy "T3" HOG data

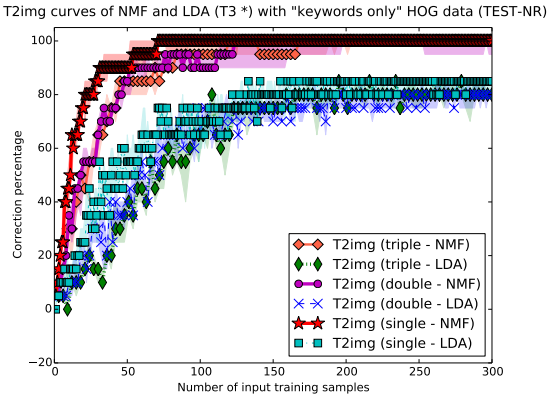
We now come to the case when noisy (Full Sentence) data are used. With carefully set TF-IDF thresholds, the difference for NMF between incremental VQ and k-means is strongly reduced. Regarding the "Img2T" performance, the learning with "single" data converges when approximately 150 training samples are used at the level around 95% and with "double" when about 225 samples are given at the level near 90%, however, with "triple" data, we can not see the trend of convergence even when all training samples from "T3" data are used. When it comes to "T2img" performance, learning in both cases of "single" and "double" could converge to 100% faster, with the minimum size of around 125 and 150 samples respectively and the convergence of learning with "triple" data is observable after around 170 samples are presented at the level of 90%. It is also noticeable that the learning performances with "triple" data for the different clustering solutions show obvious differences despite the fact that such differences seem imperceptible in "single" and "double" cases. In fact, the defect of miss-grouping of samples can not be avoided in this case, thus leading to the bad performance of filtering in which some keywords are wrongly filtered out. Therefore, for NMF with TF-IDF, using more samples makes it possible to compensate erroneous clustering to some extent, but there still exists an upper limit (eg. see Figure 4.30(a)) to the performance that is not overcome with new samples.



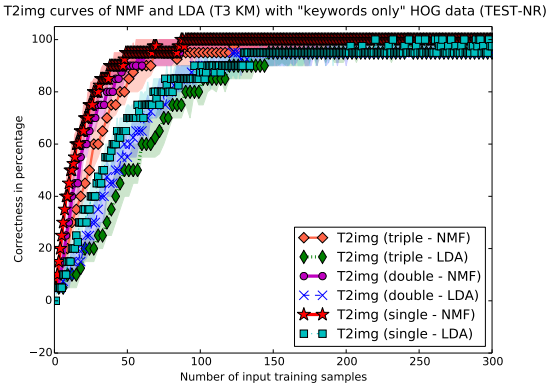
(a) "Img2T" performances by adopting the incremental clustering



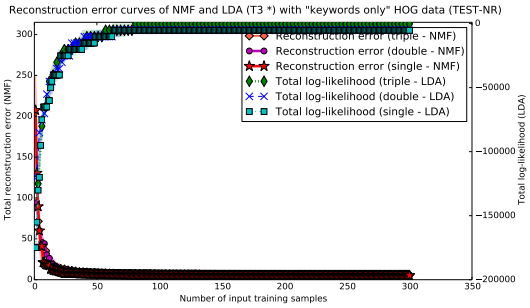
(b) "Img2T" performances by adopting K-Means clustering



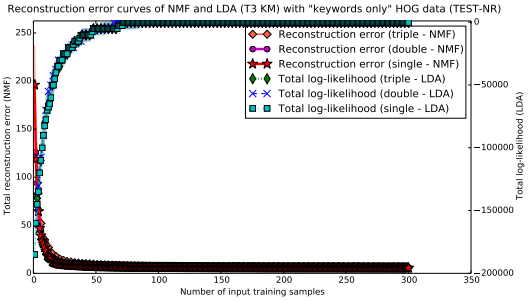
(c) "T2img" performances by adopting the incremental clustering



(d) "T2img" performances by adopting K-Means clustering

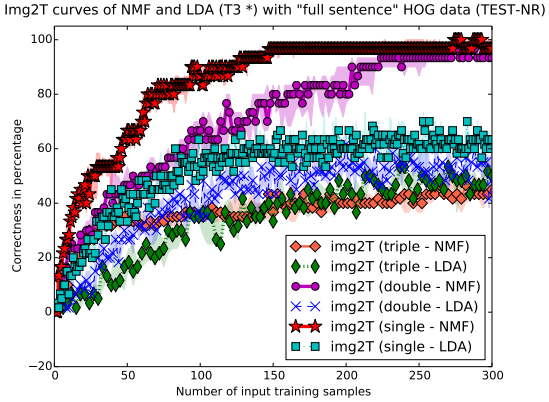


(e) Performances of the evolution of reconstruction errors by adopting the incremental clustering

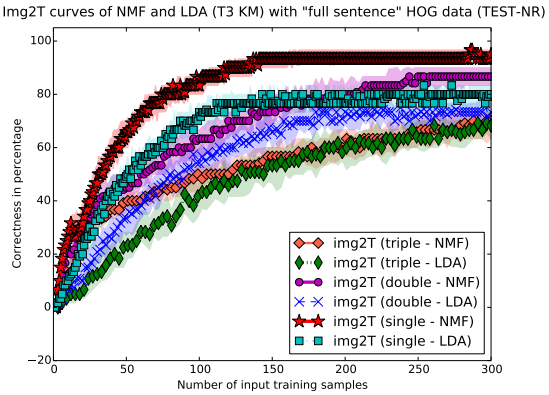


(f) Performances of the evolution of reconstruction errors by adopting K-Means clustering

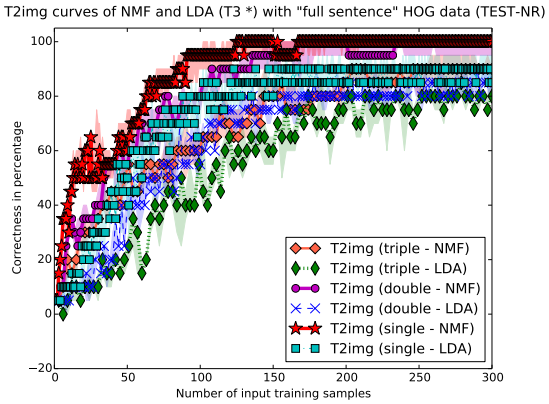
Figure 4.29: The comparison of performances during incremental learning by applying incremental clustering and K-Means clustering respectively in proposed models with "T3" data set in scenarios of $KW\&S$, $KW\&D$ and $KW\&T$ from *S-OBJ-D*.



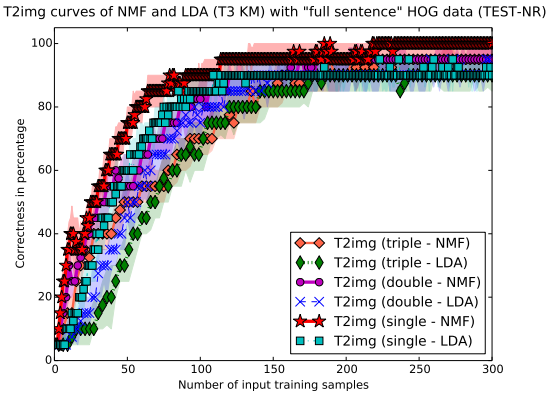
(a) "Img2T" performances by adopting the incremental clustering



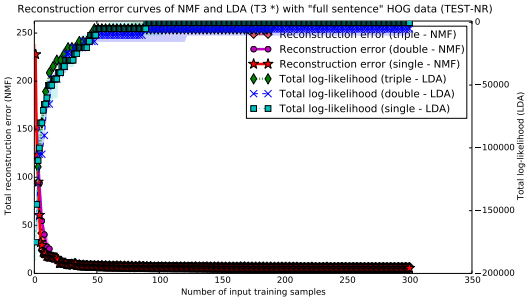
(b) "Img2T" performances by adopting K-Means clustering



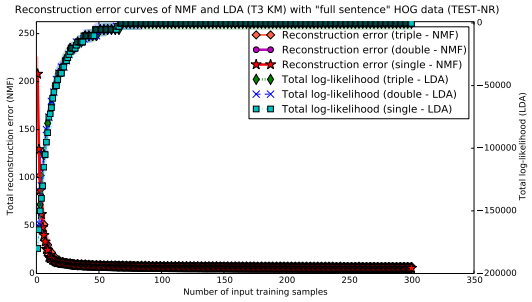
(c) "T2img" performances by adopting the incremental clustering



(d) "T2img" performances by adopting K-Means clustering



(e) Performances of the evolution of reconstruction errors by adopting the incremental clustering



(f) Performances of the evolution of reconstruction errors by adopting K-Means clustering

Figure 4.30: The comparison of performances during incremental learning by applying incremental clustering and K-Means clustering respectively in proposed models with "T3" data set in scenarios of $FS\&S$, $FS\&D$ and $FS\&T$ from *S-OBJ-D*.

As for the performance of LDA, Table 4.3 shows information concerning the convergence performances taken from the analysis of Figure 4.30. Like what has been described before,

Converging performance / *Task* / Case		**"Img2T"**		**"T2img"**	
		size of samples	final level	size of samples	final level
"single"	*K-Means*	110	80%	110	95%
	incremental clustering	110	60%	110	90%
"double"	*K-Means*	175	75%	125	95%
	incremental clustering	175	55%	125	80%
"triple"	*K-Means*	250	70%	170	95%
	incremental clustering	250	50%	170	70%

Table 4.3: Performances in terms of convergence of LDA learning in all cases.

the term "size of samples" indicates the minimum size of training samples necessary to arrive at the level of convergence which is termed as "final level" in the above table.

For LDA, compared to the training using "T1" data, using more data makes it possible to improve performance, but also reaches an upper limit at some point. It is also clear in Table 4.3 that the final convergence level via LDA learning by using a better clustering solution (ie. K-Means) would improve by around 20% compared to that by applying the sub-optimal one (ie. incremental clustering) in all cases for the task of "Img2T", and between 5% and 25% in the task of "T2img".

In this section, by conducting experiments of incremental learning with data in scenarios at different levels of referential and linguistic ambiguities, we have validated the feasibility of applying more complex vision features represented by HOG in our proposed learning models. In fact, compared to using pixel as the shape descriptor, HOG which manages to describe shape with more details is observed more sensitive and less stable in our proposed vector quantification in Section 2.1.3. For instance, the objects in *S-OBJ-D* have ground truth of 20 shape (using HOG) clusters and 6 clusters for color, however, when trying to tune parameters manually for the method of incremental clustering, we either get 19 or 21 shape clusters, never achieving 20; in addition, all 840 samples should be categorized into 120 groups, in fact, we've really found a parameter to do so, yet the categorization is only correct in the total number but unsatisfactory in the content of some groups because some samples of the same cubic side are divided into different groups while some other samples regarding different cubic sides are grouped into one. However, by using an alternative as the solution for feature quantification that relies on knowing the number of concepts in advance, we saw that our proposed models can still achieve high quality of word-meaning association learning in ambiguous and noisy environment, despite the fact that the demand of finding a good solution for a real open environment is yet to be satisfied.

4.9 Summary

As a summary of this chapter, we have extended our proposed models from batch learning experiments to the applications of incremental word-referent learning, mainly comprising data collections, performance evaluations derived from real interactive tests, policies of auto-determining the number of components/topics in incremental experimental settings and simulations of interactive protocols for experiments. The incremental learning experiments have been conducted in series, following an ascending order of data ambiguities from "unambiguous noise free" to "ambiguous noisy". As a solution to improve the effects and efficiency of incremental learning, we proposed two active learning strategies that are inspired from human intrinsic motivations and compared their performances with those by applying random choice. In addition, we also implemented complex new visual features to validate the proposed learning models and discussed two related factors which might contribute to the successful implementation.

According to the analyses of all our experimental results, both NMF and LDA computational models live up to our expectation on the cross-situational learning objective, dealing with both referential and linguistic ambiguities while making progress step by step. What have been validated also are the effective use of active strategies in our models as well as the models' compatibility of applying new vision features.

Chapter 5

Discussion

Contents

In each previous experiment, we evaluated the performance and analyzed the result respectively. In this chapter, we continue the discussion of some issues which appear more clearly if we take a transverse view across all experiment series. Besides, by reviewing all the processes of incremental learning performed by our models, it seems interesting to pay attention to the behaviors and some of their features might be or might be not in common with those possessed by human learners.

Therefore, the discussions in this chapter start from topics concerning learning algorithms, followed by the comparative studies of cognitive learning behaviors performed by computational models as well as humans, and ends by a comparison of their different types of capabilities.

5.1 Comparison between NMF and LDA approaches

As has been concluded many times in previous discussions, NMF and LDA are essentially a linear algebraic factorization approach in high dimension and a generative statistical model

respectively, both of which are capable of discovering the latent components/topics and giving explanation concerning the generation of observed data.

5.1.1 Overall behavior

For NMF, whose related values in matrix are non-negative thus giving better physical interpretations than other matrix decomposition based methods, it has first of all the striking advantage of dealing directly with raw data, the significance of which, as demonstrated in our work, resides in its ability to auto-discriminate the multimodal data presentation into different modalities, therefore giving rise to the basic concepts that are quite natural and convenient for linear combinations during reconstruction. Comparatively, LDA, which originates from the applications in the field of information retrieval, should first of all convert raw data features into vector-quantized feature symbols as adopted widely in previous works, before they are to be processed like textual documents. By avoiding the issue of processing feature data, LDA, in a sense, is only responsible for associating VQ symbols and textual words and putting them in a topic, yet whose performance is evidently relying on the quality in terms of accuracy of vector quantification. In short, in terms of factorizing raw data into basic components/topics, NMF has a sharper performance with less running time as observed in our simulations and appears less difficult to manipulate as a learning model, in the cases where no linguistic noise is involved.

However, when linguistic ambiguity exists in the interactive scenario, NMF, with its foundation in linear algebra, lacks the algorithmic mechanism (eg. statistics) to select out keywords that are really useful from a corpus of all recognized words. That's why we resort to a statistical measure (i.e. TF-IDF) to make up the defect of keywords selection as well as the determination of the number of components by providing pertinent restrictions in an effort to guide the decomposition via NMF towards the results that appear more precise and accurate. On the contrary, LDA on top of its foundation of statistics, has already been proved capable of discovering topic words from a corpus of documents. In our experiments, when all raw data are quantized to VQ symbols which are then concatenated with linguistic words (including unrelated descriptive words) to form one document representing one sample, LDA is able to find the right associations of VQ symbols with its possible corresponding topic words while withstanding the disturbance of noisy words based on the frequency analysis of those words and symbols. As shown by the results of our research, it seems that LDA is more adapted to the scenario with linguistic noise and thus performs better than NMF in experiments using pixels as the shape descriptor; yet in experiments using HOG, we observe that NMF still performs better than LDA, possibly because a perfect clustering of samples is not possible in this case with our approaches.

5.1.2 Determination of the number of topics

Another issue, difficult for both NMF and LDA, is the determination of the number of components/topics. Since there does not seen to exist an automatic method as a universal standard

for all applications, we proposed our own solutions that are both feasible to be applied on the algorithms and adaptable to our experimental scenarios.

Generally, for NMF, the number of keywords can be determined only after the linguistic ambiguity has been eradicated via keywords filtering; on the contrary, for LDA whose topic number is calculated based on the sum of the quantized feature clusters concerning both shape and color, only when the number of topics are determined can we be aware of the possible keywords according to the resultant probability distribution of words per topic. As demonstrated in experiments, proposed solutions are capable of not only figuring out the number of topics but adapting to the whole process of incremental learning as well, and in addition, a minor amount of redundancy in the determination of this number has been proved to optimize the learning performances compared to the adoption of the exact number of keywords or VQ symbol summation in related experiments.

However, as for the difficulty of manipulating them, TF-IDF (for NMF) performs well only when pertinent parameter settings are provided, which proves to be laborious not only because the threshold values should be carefully tuned but also due to the fact that if the incremental learning pattern changes (in accordance to the changes of the experimental scenario) parameter settings have to be redefined, otherwise fluctuations occur as response to the unstable results of keywords filtering. LDA's performance is first of all dependent on the quality of vector quantification of features in which the tuning of parameter settings is also involved and its limitation has been described in HOG related experiments where K-means and incremental clustering methods are compared. Besides, the current diagram of determining the number of topics for LDA figures out only a roughly correct number which is able to be increased but never decreased (cf. Section 4.3) thus resulting in the possible over-redundancy of topics.

An additional, yet interesting, remark about this issue is that, for both models, the weak point (ie. the inability of NMF to deal with linguistics without the use of TF-IDF, and of LDA to process raw feature data without resorting to VQ) ought to be overcomed before the strong point (ie. the capability of NMF to process directly the raw feature data, and of LDA to perform conduct textual/linguistic analysis) exhibits its power, and the determination of the number of components is in both cases a side effect of the solutions to the models' weak point.

5.1.3 Compatibility with active learning strategies

In active learning experiments, relevant measures of error have to be devised for both NMF and LDA. It turns out that there is no problem of applying the active learning strategies in both NMF and LDA, however, since TF-IDF is used in NMF when dealing with "full sentence" data, different strategies might collide in terms of the priority of selecting new samples (to be discussed in Section 5.3), which tremendously adds to the complexity as well as the difficulty of building active learning models.

As for the effectiveness, the superiority of active choice can be observed in almost every

experiment if LDA is applied as the learning model, however, when it comes to NMF, there are evidently some experimental results where active learning with "double" and "triple" data have approximately the same or even lower performance than random learning, probably because TF-IDF would exert its own influence (apart from that of NMF) on the learning performance on one hand, and because the current shape descriptor still has limitation of perfectly describing shapes causing errors on the other hand. That's the main reason why LDA, thanks to its better property in terms of compatibility and effectiveness in cooperation with active choice strategy, is adopted in the final for the extensive study in comparison with human behaviors.

5.1.4 About synonym and polysemy

Lastly, we can project the comparison between NMF and LDA one step further beyond our presented work and discuss what would happen if we try to model synonyms (several words having the same meaning) and polysemy (one word having several meaning).

In the framework of linear algebra, NMF is a way of finding a set of bases of lower dimension on which any input sample is able to be reconstructed through linear combinations. These bases represent particular feature values and serve as the latent components. New samples will therefore be decomposed as a weighted sum of these deterministic components. Differently, in the framework of probability theory, LDA discovers a set of latent topics which are in the form of probabilities concerning feature distributions given a certain topic and therefore there is a probabilistic link between word and features.

As seen evidently from the literal meanings of synonym and polysemy, the "Img2T" test will naturally be afflicted with synonym the same way "T2img" will have troubles with polysemy. Taking "Img2T" in a synonymous scenario for example, the problem will be that for one particular visual feature, two different words can be generated. Imagine one vision feature (noted as F) that corresponds to multiple words (noted as X, Y for instance), according to the current framework of NMF, there appears two components represented as F-X and F-Y during learning and it remains uncertain whether the former, the latter or even the both would be used in the linear combination for reconstruction when an F resembling visual feature is given (in fact, in practice both F-X and F-Y would have vision errors which probably might dominate the decision), thus the result would appear quite unmanageable. In LDA, the synonym could be represented by one topic F-$\{X, Y\}$, where X and Y could have close probabilities. From this, it is possible to generate either X or Y, and not a combination of the two as NMF would produce.

In summary, both models in the current version seem not able to deal with synonym and polysemy satisfactorily, however, with respect to the mathematical nature of the models, the probabilistic nature endows LDA with comparatively better presentations and explanations concerning the two ambiguities while NMF can not due to its deterministic nature.

5.2 Role of purity of concepts in the performance during testing

One of highlights of this thesis is the proposal that the devised learning model (especially with NMF) could help to learn pure concepts exclusively in one modality based on the multi-modal input even with noise.

In fact, cognition through the combinational use of pure concepts is a basic capacity of humans, for instance, a *black horse* as a concrete object could be identified as the combination of two abstract concepts: *black* from the color category and *horse* from the biological category. In the history of the study of cognition, the discussion about the term *concept*, concerning the combination of name and properties in relation to the existence of an object (either abstract or concrete), could also date back to the early age of Plato and Aristotle when many great thinkers from the east and the west shed light upon it.

Regarding the performances of our proposed models, taking the human capability as inspiration, it seems appealing to think that the purer the concepts are (in terms of the correctness in number and the quality of feature description in single modality only), the better the learning performances achieve. This was the case in the experiments of chapter 3 with simple datasets. However, from our results in incremental learning scenarios with more complex data, on the one hand, it is not feasible to achieve situations where the learned concepts always have a "one to one" mapping to the ground truth concepts; and on the other hand, it seems not necessary to pursue such an ideal situation because a redundant number of learned components might even help to augment the performances. Indeed, all experiments about LDA as well as NMF using "full sentence" data in the previous chapter have, compared to the ground truth, a redundancy of the number of topics/components which in most cases correspond to noisy contents and in turn make the data description of the real important topics/components even purer. Therefore, the slightly impure learned concepts which contain multi-modal data presentations will not decrease the performance all the time.

Another issue of interest, already partly discussed in Section 4.4, is the fact that the scores of "T2img" could regularly outperform "Img2T", especially when learning progress has already come to the matured phase of learning (ie. with performance score at a high level). As explained in section 4.4, this is linked to the fact that concepts are not pure enough, because if all learned topics/components would be pure, both "T2img" and "Img2T" would achieve the full performance of 100%. This can also be linked to the general property that the "T2img" task is a discriminative task (one has to choose among images from a text description), while the "Img2T" task is generative (one has to generate a text description of the image). And it is a general property in machine learning that discriminative algorithms often outperform generative ones.

This can be contrasted with the fact that in almost all experiments, the performances of "Img2T" is higher than "T2img" in the beginning of the learning. The explanation here resides in the fact that if topics/components are not well learned, the learner is supposed to fail in both "Img2T" and "T2img", however, according to the scoring rules described in

Section 4.2.1 and 4.2.2, for the same testing object, if "T2img" fails it leads to a loss of 1/36 ($\approx 2.78\%$) score, while failing on one keyword causes only a loss of $(1/2) \times 1/36$ ($\approx 1.39\%$).

5.3 Applying TF-IDF in different experimental scenarios

In a simple incremental learning experiment in Section 4.5, we managed to apply a strict filtering via TF-IDF in which keywords were correctly picked out, however, we gave it up and resorted to a soft filtering in the other experiments using NMF with noisy data not only because it becomes unmanageable as the noise level increases but also due to the findings that a small amount of noisy words wrongly recognized as keywords does not seem to decrease the testing performances too much.

Special concerns have been given to cases where active choice strategies are applied to incremental learning. Indeed, TF-IDF works better with an unbiased sampling of training samples, which is antagonistic with the use of active sampling that has the goal of biasing sampling in order to improve learning speed:

1) Knowing that TF-IDF depends on statistics, for which the corpus of linguistic samples should be given properly to ensure its effectiveness, we implemented an extra measure to let samples belonging to different categories (as the results of feature clustering, described in Section 2.1.3) be chosen preferentially until in every category there is at least one sample for training so as to avoid the observed case of extreme inequality of sample allocation where samples of some categories have overwhelming more probabilities to be selected thus leaving less or no chance for some other types of samples;

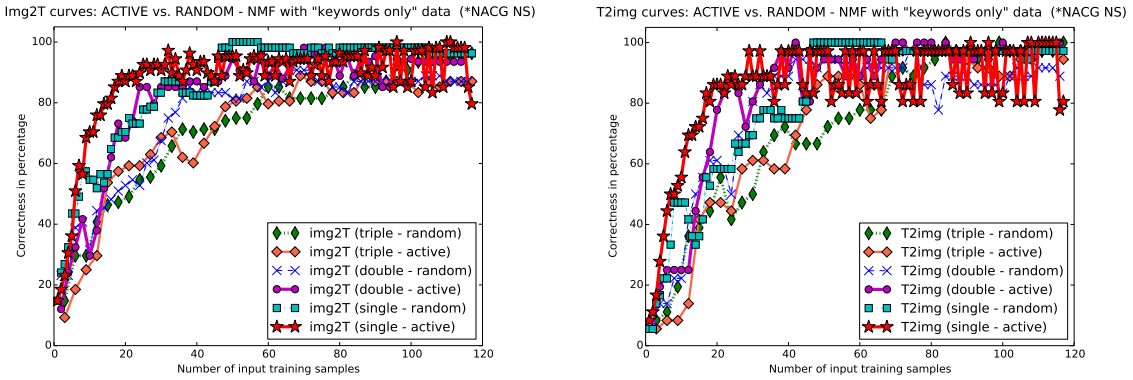
2) Subsequently, the priority issue appears during the above special period of preferential selection when active choice (ie. MRES or CBE) is applied. The solution obeys the principle that if the active choice's preference is in common with that of the preferential selection, the learner's decision follows the shared preference; otherwise when contradiction occurs, the learner's preference goes to the preferential selection in prior.

More generally, TF-IDF helps to simulate the humans' learning in an ambiguous linguistic scenario simply relying on statistics, however in contrast, many other effective measures (eg. joint attention, grammar, intonation, etc.) are taken by humans for linguistic disambiguation. As a common feature, the resilience to errors exists in both human cognitive activities and plenty of applications of modern intelligent systems, yet admittedly, while humans have more powerful solutions (eg. fuzzy inference and association) to deal with imperfect knowledge containing some uncertain or even wrong information, TF-IDF primarily works on statistical calculations. The consequence is that the effective running of TF-IDF imposes some complexity on the logic of object selection. At last, the true potentiality of the TF-IDF model is expected to be fully released when a large amount of samples is involved in that the statistics itself works for massive sample data.

5.4 About the slack strategy applied in MRES

We may notice, as introduced at the end of Section 2.5.1, that a slack strategy has been applied in MRES to benefit the active learning performance via NMF. Apart from its positive effect on the performance at the technical level, we would like to further discuss it at the theoretical level.

In order to formulate this issue clearly, inheriting the same experimental settings as in Section 4.7.1.1, we illustrate in Figure 5.1 the result of an active learning experiment (using "keywords only" data) via NMF applying MRES yet without the use of the slack strategy.



(a) "Img2T" results using $KW\&S$ from *S-OBJ-C* (b) "T2img" results using $KW\&S$ from *S-OBJ-C*

Figure 5.1: Active learning (MRES) vs. random learning - a demo of active incremental learning by applying NMF with "keywords only" data yet without the use of the slack strategy. As a result, fluctuations are observed in the performance of active learning, especially in "single" case.

Unlike the steady progress of learning performances in Figure 4.23, in Figure 5.1, there are strong fluctuations, especially evident in "single" scenario, in which a case study has been conducted to clarify some important characteristics related to the active learning progress:

A. Objects of some certain features are over selected. Among all 117 samples used for training, as high as 60.68% of them are concerning "pomme" (apple) and 59.83% related to "vert" (green). For example, from the 59_{th} learning step to the 66_{th}, every choice prefers the sample with the shape of "pomme", and in particular from Step *59* to *61* and Step *63* to *65*, all chosen samples are consecutively "pomme vert". In consequence, while samples of some certain features constitute the majority of training samples, there must be some other sort of samples which are totally ignored by the learner and fail to contribute to the variety of learned components, leading to the underperformance in testing tasks;

B. Repeated choosing samples of identical features does not improve the quality of learning via NMF, which seems contradictory to our common human experience

of focused study or engaged learning so as to crack a hard nut. In fact, the goal of NMF is to acquire a certain number of factors, which are not supposed to be further decomposed, from original raw sample data. It would be valuable that a new sample is equipped with new features that are to be factorized by NMF. However, if features in a new sample are identical to those of some previous samples, then the factorized components that are capable of reconstructing the previous samples are supposed to be able to reconstruct this new one already. In the end, this new sample of identical features would give no push to the changes of decomposed factors, resulting in no improvement of learning;

C. There exists a kind of "dead loop" as a specific phenomenon in the learning process: given a certain state of knowledge represented by learned components, the learner would choose sample(s) of certain features according to the rule of maximum reconstruction error, however, the chosen new sample would not change too much the knowledge state (mainly due to Characteristic B. as one of the reasonable explanations) then the learner would still favor this type of sample over and over again. Or in some cases, a circular transition between several states emerges when the leaner is keen on a limited number of samples, each of which helps to let the state jump from one to another but only within this circle;

D. There might be disagreement concerning results between the reconstruction error as a measure in MRES (only for candidate training samples) and "Img2T" as well as "T2img" (exclusively for testing samples). Still focusing on the performance at Step *59*, *60* and *61* where the candidate sample chosen according to MRES is "pomme vert" as a result of maximum reconstruction error it possesses among all candidates, we also check the "Img2T" and "T2img" performance about the 11_{th} object for testing, which is also "pomme vert": "T2img" task succeeds in all three steps while "T2img" succeeds totally at Step *59* and partially at Step *60* and *61* with the wrong choice of "anneau" (ring) instead of "pomme". The primary explanation is that there is too much noise in the learned components which are not pure enough to achieve perfect performances in all sorts of evaluations. However, the reconstruction error as a measure in MRES cares about the optimal linear combination that leads to the minimum reconstruction error regardless of how many components are utilized as well as how much weight each of them is endowed with; on the contrary, "Img2T" and "T2img" focus more on the principal components of highest weights (especially for "Img2T" which favors only the first two), that's why a sample of a certain sorts of features would be well reconstructed through linear combination of many seemingly not-well-learned components, but might fail in our cognition tasks where only their principal components of high weights take effect.

The fundamental reason which gives rise to the above issue from the algorithmic point of view is that the shape descriptor using pixels is not so precise enough that some shapes, like "pomme" and "bousole"(compass), "mur"(wall) and "radeau"(raft), etc, can be misrecognized. Besides, the samples themselves contain a certain level of noise (for instance, the color histogram of "vert" appears a little bit different in different samples) which might

exacerbate the phenomenon of mis-recognition.

Yet from the cognitive point of view, the proposed demo, to some extent, simulates behaviors out of the mental disease of mild paranoid which gets a person into a dead end by narrowing his perspective of mind and thus hinders the progress in terms of cognition. As a solution, the effective treatment should follow the direction of alleviating the state of stubbornness by helping him get rid of the controlling of negative mental feelings such as anxiety or fear, thus more freedom could be released for him to regain a full perspective of cognition. In fact, the slack strategy in active choice is proposed as the embodiment of this thinking in our proposed learning model of NMF: if MRES is still believed to be effective at the theoretical level, why not enlarge the size of the pool of candidates, which creates more opportunities to jump out of the "dead loop" yet still maintains the influence of MRES. As a consequence, which has been displayed in related sections in this chapter, the fluctuations disappear based on a more reasonable coverage of different categories of samples meanwhile the performance out of active choice is observed better than that out of random choice.

In the end, as an extension, we also tried MRES without using slack strategy to deal with HOG data via NMF learning. In fact, similar phenomenon as illustrated in Figure 5.1 still appears but less serious with the most frequent keyword takes around one third of all occurrences (comparatively less than the proportion of approximately 60% in the above case study), possibly due to the fact that HOG is more precise in describing shape features yet not fully capable of dealing with some specific shapes. As for LDA, which avoids the direct processing of raw shape data by applying vector quantification in advance, MRES has been observed very well to be integrated with the learning model even without using the slack strategy, yet for the convenience of comparing equally the effects of MRES on NMF and LDA, the slack strategy has been adopted as well in related active learning experiments in Section 4.7.2 and 5.5.2.

5.5 About the repetition behavior

5.5.1 Analysis of the experiments

In [96], where comparative studies of active learning and random learning had been tested on human participants, Kachergis et al first proposed that most learners use immediate repetition to disambiguate pairings, and then further confirmed that those who repeat multiple pairs per trial outperform those who repeat only one pair per trial, given the experimental rule that in every trial four artificial object-word pairs are provided.

In order to evaluate this repetition behavior on our proposed model, we choose the learning model of LDA which appears more robust and effective in the application of active learning strategies and remains less sensitive to the parameter tuning, in cases of "keywords only (KW)" and "full sentence (FS)" with data of "single ($\boldsymbol{S}$)", "double ($\boldsymbol{D}$)" and "triple ($\boldsymbol{T}$)" respectively, using both MRES and CBE. We therefore took the experiments of Section 4.7.2

and computed the statistical results presented in Table 5.1. Here, "R-NXT" is the mean of

Repetition / *Strategy* / Case		**MRES**		**CBE**		**random**	
		R-NXT	R-WHT	R-NXT	R-WHT	R-NXT	R-WHT
	S	0.29	*0.00*	0.31	*0.00*	0.31	*0.00*
KW	***D***	0.84	*1.00*	1.15	*0.34*	1.17	*0.30*
	T	1.74	*2.04*	2.38	*0.89*	2.41	*0.90*
	S	0.34	*0.00*	0.32	*0.00*	0.32	*0.00*
FS	***D***	0.98	*1.00*	1.16	*0.34*	1.17	*0.31*
	T	1.92	*2.02*	2.38	*0.87*	2.41	*0.88*

Table 5.1: Mean value of the word repetitions in successive trial and the feature repetition within trial in active learning experiments using LDA learning model.

the repetition of descriptive keywords for samples in the successive trial, equivalent to the term "immediate repetition" in [96] and "R-WHT" for "within-trial feature repetition", the mean of the repetition of same features from sample(s) within a trial, which will be discussed later on.

According to [96], active learners mainly rely on immediate sample repetition to facilitate learning, yet from our implementation, the resulting strategy seems different: in the "triple" scenario, for instance, random sample choices (for both *KW* and *FS* cases) led to a mean repetition of 2.41 words in successive steps, however, concerning active choices, both MRES and CBE lead to less word repetitions, especially for MRES, by applying which the mean repetition of words is 1.74 in *KW* and 1.92 in *FS*. In fact, this phenomenon fits almost all cases (despite the fact that the exception occurs when MRES is applied in *FS* with "single" data), particularly in "double" and "triple" scenarios in which referential ambiguities prominently appear.

Two basic reasons could be used to explain such a difference in applying the repetition strategy. On one hand, in [96], each trial consists of four mutually different objects thus no "within-trial repetition of objects" is allowed, however in our experiment scenarios, especially in "double" and "triple" cases, the same features (shape or color) from different objects could appear in a double or a triple and this gives rise to a "within-double feature repetition" or "within-triple feature repetition" (both particular cases of "R-WHT") which can simply reduce the complexity of each trial. In fact, for instance in the "triple" scenario, the number of repeated features inside a triple is 0.88 (in *FS*) or 0.90 (in *KW*) with the random strategy and 2.02 (in *FS*) or 2.04 (in *KW*) with the MRES. On the other hand, unlike computational models, humans are less efficient at keeping a long-term memory of the past co-occurring records and hence the successive repetition facilitates learning for humans but not for our model.

It is also interesting to go back to the figures in Section 4.7.2 to observe that, in situations where the performances in terms of "R-NXT" and "R-WHT" are similar, the learning progress exhibits less difference between active and random strategies (cf. Figure 4.25(a) and 4.25(b)). When "R-WHT" is allowed in the experimental scenarios, the more it is utilized the better

learning quality is achieved (cf. figures 4.25(c), 4.25(d), 4.26(c) and 4.26(d)). On the contrary, when "R-WHT" is infeasible (ie. in "single" scenario), it seems that "R-NXT" still plays an important role to distinguish the performance of active learning from that by applying random choice (cf. Figure 4.26(a) and 4.26(b)). As a conclusion, "R-WHT" is adopted as a measure in priority by an effective active learning strategy to reduce ambiguity, followed by "R-NXT" whose power seems secondary yet still visible. This combination use of "R-WHT" and "R-NXT" quite conforms to the way humans would behave if repetition in a trial is allowed.

5.5.2 Active learning without within trial repetition

Although a comparison of learning behaviors has been conducted in the previous section between the performances of our proposed models and humans in reality, the feasibility of using "within-trial feature repetition" in our experiment was strongly exploited by our model. Hence, we set restrictions on all active learning experiments in Section 4.7.2 such that "within-trial feature repetition" is inhibited in every trial, either "double" or "triple". For example, in a triple, if the first object is "lego rouge", then any object which is red ("rouge") in color or has the shape of "lego" will never be second of the third sample in this triple. Based on the statistics of 50 times repetitions of experiments, the learning progress using *FS* data is illustrated in Figure 5.2, and the performance gains for active learning versus random learning (using both *KW* and *FS* data) measured by the area under the curve are shown in Table 5.2 and 5.3 and repetition statistics are shown in Table 5.4.

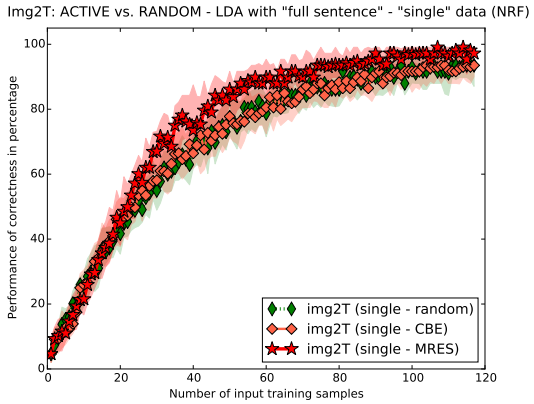
Img2T / Case	*Strategy*	**MRES** over **random**		**CBE** over **random**	
		D	***T***	***D***	***T***
R-WHT	*KW*	0.039	0.107	0.004	0
allowed	*FS*	0.109	0.157	0.01	-0.002
R-WHT	*KW*	0.003	0.013	-0.009	-0.004
inhibited	*FS*	0.036	0.026	0	0.005

Table 5.2: Gain of performance value of active learning over random learning in "Img2T".

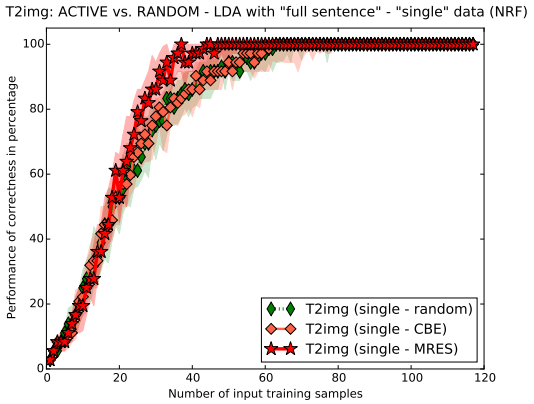
T2img / Case	*Strategy*	**MRES** over **random**		**CBE** over **random**	
		D	***T***	***D***	***T***
R-WHT	*KW*	0.042	0.108	0.002	-0.002
allowed	*FS*	0.069	0.1267	-0.003	-0.002
R-WHT	*KW*	0.003	0.022	-0.006	0.005
inhibited	*FS*	0.033	0.022	0.005	-0.009

Table 5.3: Gain of performance value of active learning over random learning in "T2img".

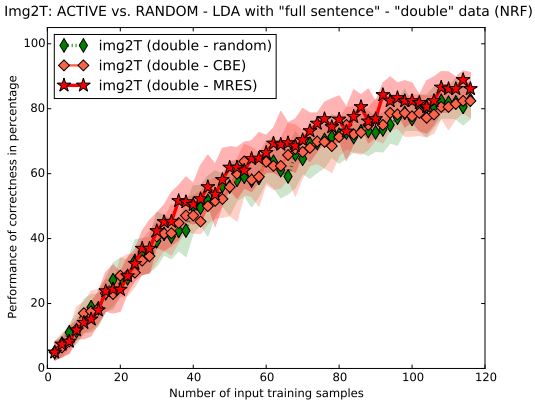
From these experiments, we can observe first a sudden fall of performance in terms of both the learning speed illustrated by Figure 5.2 and the relative performance with a loss of up to 40% compared to the experiments where the within trial repetition is possible.



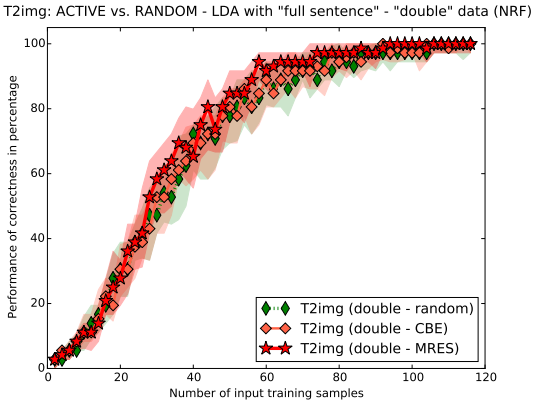
(a) "Img2T" results using $FS\&S$ from *S-OBJ-C*



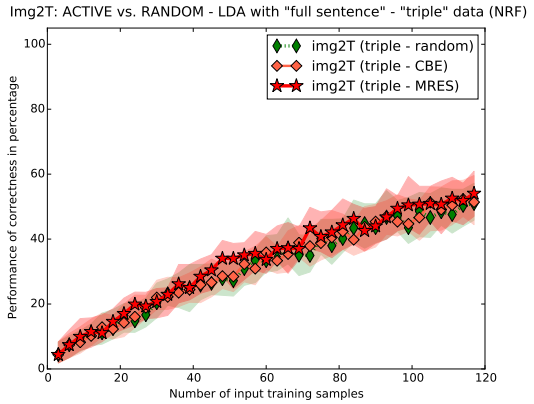
(b) "T2img" results using $FS\&S$ from *S-OBJ-C*



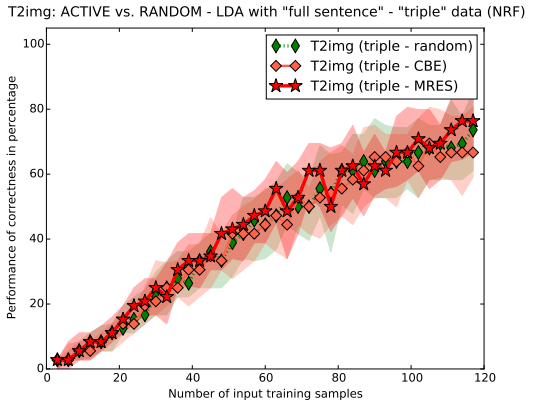
(c) "Img2T" results using $FS\&D$ from *S-OBJ-C*



(d) "T2img" results using $FS\&D$ from *S-OBJ-C*



(e) "Img2T" results using $FS\&T$ from *S-OBJ-C*



(f) "T2img" results using $FS\&T$ from *S-OBJ-C*

Figure 5.2: Active learning vs. random learning - comparisons in performances between MRES, CBE and random choice by applying LDA with "full sentence" data when the feature repetition within trial is inhibited.

Repetition / *Strategy* / Case		**MRES**	**CBE**	**random**
		R-NXT	R-NXT	R-NXT
	S	0.30	0.32	0.32
KW	***D***	1.18	1.26	1.27
	T	2.74	2.84	2.80
	S	0.33	0.31	0.32
FS	***D***	1.27	1.25	1.25
	T	2.85	2.88	2.84

Table 5.4: The mean value of the word repetitions in successive trial when the feature repetition within trial is inhibited in active learning experiments using LDA learning model.

Similarly, the gain between random and active learning representing the learning superiority of the later, shrinks as well in "double" and "triple" scenarios, where "within-trial feature repetition" used to be allowed, as shown in Table 5.2 and 5.3 (in spite of a few exceptions concerning the performance of CBE).

However, even if the global performances are lower when "within-trial feature repetition" is not allowed, active choice will resort more to immediate repetition so as to exhibit its superiority of performance over random choice, which is especially noticeable by observing the increase of values concerning the repetitions of MRES in terms of "R-NXT" in Table 5.1 and 5.4.

As for the conclusion in [96] that active human learners who repeat multiple pairs (of word and object) outperform those repeating only one pair on average, data from Table 5.4 partially agree with it in "double" case (which has the most equivalent learning scenario to [96] with four feature-word pairs) with the mean repetition value of around 1.25 words and in "triple" case with the mean repetition value of around 2.8 words. Evidently, computational learning models use comparatively less immediate repetitions.

Finally, we go back to the summary at behavioral level about the learning performances. First of all, "within-trial feature repetition" plays a predominant role in disambiguation and acts as the main power to make active learning effective for computational algorithms, which is also consistent with human learning experience. Besides, immediate repetition, utilized almost at the same level for both active and random learning when "within-trial feature repetition" is inhibited, also proves to be another factor which helps the learning be less ambiguous and renders active learning strategy useful (observed in the *FS* case). However, the computational learning model, unlike humans which rely more on short-term memory for this task, relies less on the immediate repetition because the disambiguation via the repetition of features used long ago still appears possible.

5.6 About the relative use of *exploration* and *exploitation*

Based on our experimental data, we have conducted comparative analyses at the behavioral level, for instance, about how the instant repetition behaviors help to reduce referential ambiguities as stated previously in Section 5.5. Another issue of interest is the behavior of *exploration* and *exploitation* when applying CBE in incremental learning. Using the same legends as in tables in Section 5.5, we show in Table 5.5 the proportion of EXP (*exploration*) as well as EPT (*exploitation*) in all active choices. As can be seen, *exploration* is adopted with strikingly high frequency by the learner in all experimental cases while *exploitation* plays a limited supplementary role. It is also noticeable that the more ambiguity (in terms of both referential and linguistic) exists, the higher proportion of EPT appears, which is consistent with the policy of CBE that larger reconstruction error (equivalent to lower confidence) leads to more conservative actions (ie. *exploitation*) so as to consolidate what has been learned. As a matter of fact, we have tried confidence thresholds with many ranges of values and found that if the proportion of *exploitation* becomes larger, the training samples would just be chosen from a limited categories of samples, leading to the deficient training with other samples and finally resulting in the stagnation or even the decrease of performances. Therefore, we tried to lower *exploitation*'s proportion until superior performances of CBE can be observed over random learning.

Proportion / *Data* / Case		***S***		***D***		***T***	
		EXP	EPT	EXP	EPT	EXP	EPT
R-WHT	*KW*	92.56%	7.44%	90%	10%	89.85%	10.15%
allowed	*FS*	90.67%	9.33%	83.14%	16.86%	89.13%	10.87%
R-WHT	*KW*	93.95%	6.05%	87.79%	12.21%	89.33%	10.67%
inhibited	*FS*	95.62%	4.38%	95.28%	4.72%	73.13%	26.87%

Table 5.5: Proportion of *exploration* and *exploitation* behaviors in active learning experiments (conducted 50 times) applying CBE.

One could wonder why the CBE active learning model strongly prefers *exploration*. There are the following factors which might possibly be the explanations on the part of algorithm's behavior: 1). first of all, there is a bias towards the variety of samples in the evaluation. Because it is based on the global testing of different samples, the covering of as many features as possible of training samples plays a critical role. Since, as just mentioned above, *exploitation* in CBE has the tendency of restricting the choice of training candidates to a limited range, thus *exploration* is much more applied to ensure the variety coverage; 2). in some cases, the proposed learning algorithms are proved capable of learning object features even if related samples are just presented very few times. Considering the property of interchangeability (possessed by both LDA and NMF) and the "long-term memory" effect caused by the perfect memorization of past samples, there is less demand of repeating a used sample because no matter when it has appeared before it makes no difference to the training. Yet despite the above differences, CBE shares in common with human behaviors the fact that instead of completely repeating one of the used samples, new samples whose features remain partially

the same as before while bringing in some differences contribute to improve the learning.

5.7 Comparison with human capabilities

Compared to the real word-referent learning system of humans, what has been proposed in this thesis could be regarded as models with massive simplifications, which we can trace in the following three major points:

1. *Visual system*

   The biological visual system (ie. mainly the eye and the visual processing area in the brain) of human is a complex system endowed with a combination of critical visual functions, natural to humans but difficult for the computer vision. First of all, humans eyes are less illuminance-sensitive when dealing with colors, for example in experiments using data from cubes (ie. *S-OBJ-D*) whose background is seen as pure white by human eyes, the original color histogram reports that there are always some red pixels recognized as systematic errors. As a solution, we had to use a filtering on the intensity to remove the above noisy influence. Besides, the human visual system is quite efficient with the transformation of coordinates, eg. image rotation, deformation due to the angle of observations, etc, which in the case of computer vision needs laborious efforts. In addition, saliency detection is effectively performed by humans, therefore leading to efficient object segmentation and tracking against the background;

2. *Cognitive ability*

   Though many cognitive abilities in humans probably contribute to the word-referent learning ability, the memory system structure has been shown to produce a clear difference between the human behavior and our models: as described in the previous section, human is comparatively weaker in memorizing details, especially in instant memory with large quantity of data as well as in the long term; on the contrary, efficient memory through data indexing helps computer take advantage of the whole data for learning;

3. *Social skills*

   Human beings are social animals. For a child, its incremental acquisition of social skills accompanies the growth of ages. Therefore, it is hard to imagine that an infant's learning would happen strictly without the use of of social means like joint attention which could be further defined as responding to joint attention (RJA) and initiating joint attention (IJA) [135]. This capacity would clearly take an important part in reducing ambiguity in real learning scenarios. For the same reason, the growing infant at least knows some basic language skills, hence the keywords would be much easier to be referred to when interactions take place. However, this does not exclude the possibility that sometimes infants also primarily use statistics to figure out the correct references across various situations. Furthermore, compared to the proposed models, the underlying cause of human intrinsic motivation definitely appears more complex, and the phenomena of

deliberate selection rules as introduced in Section 5.3 which is just for the effectiveness of models would probably be very different.

Nevertheless, even if constrained by the limited implementation of component systems, the proposed learning models still manage to demonstrate how simple concepts can emerge from the visual and audio data, by applying specific strategies in a series of learning tasks and achieve good performances. As indicated at the beginning, the research of this thesis is just anchoring a few principles concerning cognitive learning while harnessing less complex data as well as interactive actions so that the potentiality of computational learning algorithms could be observed and exploited possibly at its maximum, despite the fact that making use of more complex strategies and signal processing would further improve the learning performances in future versions of our proposed models.

Chapter 6

Conclusion and future work

Contents

6.1 Conclusion

This thesis describes the building of complete computational models for the interactive learning of objects in terms of word-feature associations through simple communications between a human teacher and a computer as a learner. Inspired by the infants' impressive ability to learn to recognize and name objects, which often takes place in a parent-child interaction, our study mainly focuses on two issues in the domain of machine learning and developmental robotics: *cross-situational learning* and *active learning*.

We developed two *cross-situational learning* models based on two different topic models: Non-Negative Matrix Factorization (NMF) and Latent Dirichlet Association (LDA), each of which, associated with complementary processing (TF-IDF statistical word filtering for NMF and vector quantification of raw data for LDA) has been proved capable of dealing with two sorts of ambiguities: the *referential ambiguity* and *linguistic ambiguity* in simple interactive scenarios.

The results of learning were evaluated with two approaches 1). measuring the similarity of learned results and their corresponding ground truth, which might not be available in practical use, and 2). performing two tasks (called "Img2T" and "T2img") that simulate the everyday testing of a learner's achievements by offering objects' information in one modality (textual words or vision feature) and asking for the other (vision feature or textual words).

From these evaluations, we found that our models are able to learn representations that make it possible to solve the two tasks and that NMF, in simple scenarios, is able to produce pure concepts in the format of "one modality-one word" close to the ground truth. Furthermore, we proposed specific methods for the auto-determination of the number of components/topics for NMF and LDA, and proved them to be pertinent in all experimental scenarios.

Besides, two *active learning* strategies: Maximum Reconstruction Error Based Selection (MRES) and Confidence Base Exploration (CBE) were compared and proved compatible with proposed *cross-situational learning* models while effective when comparing the incremental learning performances in between with the strategies of MRES, CBE and random choice. We also made a high-level comparison of the behaviors of computational models and humans. As a result, in executing some behaviors, like slack strategy and instant repetition, both algorithms and humans have similar behaviors, while in other cases such as the relative use of exploration and exploitation when applying CBE, behaviors appear widely divergent due to different types of capabilities.

Finally, all these results have been obtained with an experimental setup that includes all the necessary data processing modules. These processings include the necessary computer vision processes and pre-processing specific to the NMF and LDA algorithms. While simplified, this setup makes it possible to record data for exhaustive simulation experiments and also perform preliminary experiments in real human-robot interactions (see Appendix C) where a human has to teach object and feature names to a robot.

6.2 Future work

As the continuation of this thesis, future work can be foreseen on several aspects:

6.2.1 Relative size of modality representations

A first technical point concerns the NMF algorithm. As has been mentioned in Section 2.1.1.2, one reason why we set the histogram length of HOG descriptor to be 900, the same as in the case of utilizing pixel (compared to a histogram of 80 values for color), is that we want to make use of the effect concerning the auto-separation of different modal data by applying NMF. Experimentally, we observed that the longer the modality length is, the more value this modality will be allocated by following the current updating rules (cf. Section 1.4.2.3 and 2.3) during NMF decomposition. As a result when we review the decomposed components, the one indicating a simple concept in terms of the modality of the shorter length would have in its histogram noisy values corresponding to the other modality of the longer length, and in the extreme case when one modality has the length far longer than that of the other, the noise would even overwhelm the needed data value in some components. An idea emerges quite naturally that compensation weight, with negative correlation to the modality length, could be introduced during the normalization of NMF's iterations, but this would break the

assumption that the true modality distributions in the histogram is supposed to be unknown a priori. It would therefore be of interest to study how NMF could keep such a separation power of pertinent modalities with modalities of very different length.

6.2.2 Synonyms and polysemy

As briefly discussed in Section 5.1 about the possibility of our proposed models of NMF and LDA to deal with synonymous and polysemous learning situations, it would be interesting to design experiments in which synonymy and polysemy exist. Based on the current expectation on the models' performances, solving these problems would be more easily performed through LDA, and should include more information, for instance by introducing the conditional probability derived from the contextual environment so as to calculate a more reliable posterior probability distributions concerning word-meaning mappings to well interpret the decision of why adopt one of the polysemous meanings while discarding others.

6.2.3 Hierarchical layout of knowledge

Compared to the very simple concepts and words we have focused on, human concepts are much more complex. In particular, they often use hierarchies to describe different nested categories of concepts (e.g. Animals > Mammals > Dogs). It would be interesting to study if hierarchical extension of NMF [102] or LDA [21] could be applied to our problem and generate topics that resemble the ones used by humans. In particular, would it be possible to find a solution to precisely control the effects, in terms of how many levels in the hierarchy and how many concepts in each layer ?

6.2.4 Wider range of multi-modal data

Proposed works only implement shape (using either pixel or HOG) and color for the object-word learning, however, in [5] for example, multi-modal data are applied, containing visual, audio and haptic information, to help robot learn more object concepts. In a similar way, it would be promising to extend the current models to adapt more complicated multi-modal data including for instance the touch to learn words related to object texture, or the action performed by the robot in order to learn action names, i.e. verbs. Moreover, word learning could be extended to other domains like object size or spatial relationships among objects or between an object and the robot, which are expected to reach the limitations of the current framework of bag of words (BoW) where the order of the words are not taken into account and the non-symetric spatial relationships (eg. "the glass is over the book") can not be described. A more complex words representation using for example n-grams could be used.

In a shorter term, even in the implementation of vision features, more complex vision descriptors like those learned by deep learning could be implemented to avoid the manual

specifications of the two visual features we used and to apply the models in more realistic scenarios.

6.2.5 More sophisticated language model

Natural language ability contributes a lot to help humans do well in tackling linguistic references. It is quite promising that future development of our learning models could also make use of language models, for example concerning the grammar. In fact, one possible improvement can be made by exploiting existing resources such as WordNet [61] and VerbNet [165] where some semantics on words exist and could be used. This would provide a much more efficient replacement than the simple statistical TF-IDF filtering used in this PhD. On a longer term, it could be linked to the idea mentioned previously of learning the grounding of hierarchies of concepts and could lead to the more realistic real robot interactive applications.

6.2.6 From individual learning to group learning

Finally, Schueller's study on naming games [164] demonstrate the active learning in a social environment in which a group of agents ought to come to a consensus about the word-object associations. It simulates the evolutionary process of the formation of a local language and studies the influence of active learning in this setup. However, the word-object learning rule is comparatively much simpler than ours. An interesting perspective would then be to consider the combination of the proposed models in this thesis and the framework of naming games so as to let the mechanism of word-object learning be analyzed at a social scale, studying for example if stable concepts (such as the individual colors) could appear and how to design robust enough visual features to deal with viewpoint variations among agents.

Appendix A

LDA algorithms

A.1 Variational algorithms

Variational methods (see detailed formulation in [20]), also known as variational expectation maximization (EM) algorithm, is based on using a variational distribution $q(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\phi} \,|\eta, \gamma, \lambda)$, which can be fully factorized, as a good approximation to the original coupled system $p(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\phi} \,|\mathbf{w}\,; \alpha, \beta)$. The difference between the two distributions is noted as $KL(q(\mathbf{z}, \theta, \phi \,|\eta, \gamma, \lambda) \,\|p(\mathbf{z}, \theta, \phi \,|\mathbf{w}\,; \alpha, \beta))$ where $KL(q \,\|p)$ is the Kullback-Leibler divergence of two distributions of q and p. The Jensen inequality provides a lower bound on $\log p(\mathbf{w}; \alpha, \beta)$, formulated as :

$$\begin{aligned}
\log p(\mathbf{w}; \alpha, \beta) &= \log \iint \textstyle\sum_z p(\mathbf{w}, \mathbf{z}, \theta, \phi; \alpha, \beta) d\theta d\phi \\
&= \log \iint \textstyle\sum_z \frac{p(\mathbf{w}, \mathbf{z}, \theta, \phi; \alpha, \beta) q(\mathbf{z}, \theta, \phi | \eta, \gamma, \lambda)}{q(\mathbf{z}, \theta, \phi | \eta, \gamma, \lambda)} d\theta d\phi \\
&\geq \iint \textstyle\sum_z q(\mathbf{z}, \theta, \phi \,|\eta, \gamma, \lambda) \log \frac{p(\mathbf{w}, \mathbf{z}, \theta, \phi; \alpha, \beta)}{q(\mathbf{z}, \theta, \phi | \eta, \gamma, \lambda)} d\theta d\phi \\
&= \iint \textstyle\sum_z q(\mathbf{z}, \theta, \phi \,|\eta, \gamma, \lambda) \log p(\mathbf{w}, \mathbf{z}, \theta, \phi; \alpha, \beta) d\theta d\phi \\
&\quad - \iint \textstyle\sum_z q(\mathbf{z}, \theta, \phi \,|\eta, \gamma, \lambda) \log q(\mathbf{z}, \theta, \phi \,|\eta, \gamma, \lambda) d\theta d\phi \\
&= E_q[\log p(\mathbf{w}, \mathbf{z}, \theta, \phi; \alpha, \beta)] - E_q[\log q(\mathbf{z}, \theta, \phi \,|\eta, \gamma, \lambda)] \\
&\triangleq L(\eta, \gamma, \lambda; \alpha, \beta)
\end{aligned} \tag{A.1}$$

It could be easily verified, as stated in [20], that the difference between $\log p(\mathbf{w}; \alpha, \beta)$ and $L(\eta, \gamma, \lambda; \alpha, \beta)$ is just the KL divergence between the variational posterior probability and the true posterior probability, written as

$$\log p(\mathbf{w}; \alpha, \beta) - L(\eta, \gamma, \lambda; \alpha, \beta) = KL(q(\mathbf{z}, \theta, \phi \,|\eta, \gamma, \lambda) \,\|p(\mathbf{z}, \theta, \phi \,|\mathbf{w}\,; \alpha, \beta)) \tag{A.2}$$

where $KL(q(\mathbf{z}, \theta, \phi \,|\eta, \gamma, \lambda))$ serve as the lower bound of $\log p(\mathbf{w}; \alpha, \beta)$.

Therefore, the problem of finding a variational distribution $q(\mathbf{z}, \theta, \phi \,|\eta, \gamma, \lambda)$ which minimizes $KL(q(\mathbf{z}, \theta, \phi \,|\eta, \gamma, \lambda) \,\|p(\mathbf{z}, \theta, \phi \,|\mathbf{w}\,; \alpha, \beta))$ can be converted to the problem of maximizing $L(\eta, \gamma, \lambda; \alpha, \beta)$ by adjusting parameters of η, γ and λ.

This optimization problem, generally speaking, is solved by using Lagrange Multipliers, then executed by applying fixed point iteration to update the latent variables. And the whole process can be described in two phases as **E** step and **M** step.

In **E** step, $L(\eta, \gamma, \lambda; \alpha, \beta)$, which is created as a function for the expectation of the log-

likelihood as stated in Equation A.1, is to be maximized by using Lagrange Multipliers with regard to η, γ and λ respectively while assuming that α and β are fixed. As a result, we get for each variational paramter its iterative formula with respect to other variables, and the updating work is done by the use of fixed point iteration.

In **M** step, Lagrange Multipliers and fixed point iteration are also applied for $L(\eta, \gamma, \lambda; \alpha, \beta)$, with regard to estimated parameters of α and β, however, fixing all variational parameters of η, γ and λ for the purpose of further maximizing the lower bound by improving the model parameters. We could get the iterative equation for β by following the above procedure and also for α with additional using of Newton-Raphson algorithm, as described in [20].

A.2 Sampling-based algorithms

Apart from the variational algorithms, which is highly dependent on its initialization of parameters and the trap of local optimum, sampling-based algorithms, especially Gibbs Sampling, are proposed and more often used for inferring posterior variables, despite the fact that variational methods are shown to be faster in terms of convergence.

Gibbs Sampling [131, 183, 74] is one member of a family of algorithms from the Markov Chain Monte Carlo (MCMC) framework [69], aiming to build a Markov chain that has the target posterior distribution as its stationary distribution. In effect, Gibbs Sampling is based on sampling from conditional distributions of the variables of the posterior. Since we can not directly calculate the joint posterior probability in Equation 1.37, then a posterior conditional distribution should be found first before Gibbs Sampling is to be applied.

We notice from Figure 1.19 that the Dirichlet-Multinomial conjugate structure happens in $\alpha \rightarrow \theta_d \rightarrow z_{d,n}$ and $\beta \rightarrow \phi_k \rightarrow w_{d,n}$, therefore the posterior distribution for θ_d and ϕ_k are

$$\begin{aligned} \theta_d &\sim Dir(\alpha + \mathbf{n}_d) \\ \phi_k &\sim Dir(\beta + \mathbf{n}_k) \end{aligned} \tag{A.3}$$

where $\mathbf{n}_d = [n_d^{(1)}, ..., n_d^{(K)}]$ represents the number of observed words belonging to every topic in the d_{th} document and $\mathbf{n}_k = [n_k^{(1)}, ..., n_k^{(V)}]$ the number of words generated by the k_{th} topic for all documents (with V referring to the size of the vocabulary for the total words of all documents).

It is by evident that given the probability distribution of $\mathbf{z}$, it would be available for the distributions of $\mathbf{n}_d$, $\mathbf{n}_k$ and thus the posterior probabilities of θ_d, ϕ_k in Eq.A.3.

Right now, we can focus on finding a simpler algorithm, known as the collapsed Gibbs sampler, where the multinomial parameters are integrated out and only $z_{d,n}$ is sampled, expressed as $p(z_{d,n} \,|\mathbf{z}_{-d,n}, \mathbf{w}; \alpha, \beta)$ where $\mathbf{z}_{-d,n}$ is the topic allocations for all words from all

documents except for $w_{d,n}$, and come to

$$\begin{aligned} p(z_{d,n} \,|\mathbf{z}_{-d,n}, \mathbf{w}; \alpha, \beta) &= \frac{p(z_{d,n}, \mathbf{z}_{-d,n}, \mathbf{w}; \alpha, \beta)}{p(\mathbf{z}_{-d,n}, \mathbf{w}; \alpha, \beta)} \\ &= \frac{p(\mathbf{z}, \mathbf{w}; \alpha, \beta)}{p(\mathbf{z}_{-d,n}, \mathbf{w}_{-d,n}; \alpha, \beta).p(w_{d,n}; \alpha, \beta)} \\ &\propto \frac{p(\mathbf{z}, \mathbf{w}; \alpha, \beta)}{p(\mathbf{z}_{-d,n}, \mathbf{w}_{-d,n}; \alpha, \beta)} \end{aligned} \tag{A.4}$$

Note that $w_{d,n}$ is independent of $\mathbf{z}_{-d,n}$ and $p(w_{d,n}; \alpha, \beta)$ is available from direct observations.

For $p(\mathbf{z}, \mathbf{w}; \alpha, \beta)$, which in fact indicates the whole process of generating a word in a document, we have

$$\begin{aligned} p(\mathbf{z}, \mathbf{w}; \alpha, \beta) &= \iint p(\mathbf{z}, \mathbf{w}, \theta, \phi; \alpha, \beta) d\theta d\phi \\ &= \int p(\mathbf{z} \,|\theta; \alpha) p(\theta; \alpha) d\theta \int p(\mathbf{w} \,|\phi; \beta) p(\phi; \beta) d\phi \end{aligned} \tag{A.5}$$

where we notice that the two integral items just correspond to the generative processes of $\alpha \to \theta_d \to z_{d,n}$ and $\beta \to \phi_k \to w_{d,n}$ as Dirichlet-Multinomial conjugate structures shown in Figure 1.19. Hence,

$$\int p(\mathbf{z} \,|\theta; \alpha) p(\theta; \alpha) d\theta = \prod\nolimits_d \frac{B(\mathbf{n}_d + \alpha)}{B(\alpha)} \tag{A.6}$$

$$\int p(\mathbf{w} \,|\phi; \beta) p(\phi; \beta) d\phi = \prod\nolimits_k \frac{B(\mathbf{n}_k + \beta)}{B(\beta)} \tag{A.7}$$

where the multinomial beta function $B(\alpha) = \frac{\prod_k \Gamma(\alpha_k)}{\Gamma(\sum_k \alpha_k)}$ and $\Gamma()$ represents the gamma function.

By combining Eq.A.6 and Eq.A.7, we have

$$p(\mathbf{z}, \mathbf{w}; \alpha, \beta) = \prod\nolimits_d \frac{B(\mathbf{n}_d + \alpha)}{B(\alpha)} \prod\nolimits_k \frac{B(\mathbf{n}_k + \beta)}{B(\beta)} \tag{A.8}$$

Finally, subsituting equations Eq.A.8 into Eq.A.4, the chain rule equation of Gibbs sampling equation for LDA can then be derived [1]

$$\left[p(z_{d',n'} \,|\mathbf{z}_{-d',n'}, \mathbf{w}; \alpha, \beta) \propto \prod\nolimits_d \frac{B(\mathbf{n}_d + \alpha)}{B(\mathbf{n}_d^{-(d',n')} + \alpha)} \prod\nolimits_k \frac{B(\mathbf{n}_k + \beta)}{B(\mathbf{n}_k^{-(d',n')} + \beta)} \right] \tag{A.9}$$

where the superscript $^{-(d',n')}$ signifies leaving the $(d', n')_{th}$ token out of the calculation.

[1] All the expanded formulae can be referred to in literature [131, 183, 74]

A.3 Online LDA

In many applications of LDA based models, the collection of documents grows over time, making it infeasible to run batch algorithms repeatedly. Therefore, online LDA (o-LDA) models[74, 87, 31], which update the estimates of the topics once each document is observed, are proposed.

For example, Griffiths[74] proposed an on-line version of the Gibbs sampler using Eq.A.9 to assign words to topics, but with counts only from the subset of the words rather than the full data. Hoffman[87] devised an online variational Bayes for LDA, where the main structure of the Expectation Maximization algorithm (annex A) is kept, however, with modifications that 1). the training only takes place in a special corpus containing only a randomly chosen document which is occurring repeatedly and 2). the updating of λ (the variational parameter for β) is the weighted average of the old value and the newly iterated one. "o-LDA" model is proposed in [176, 10] where paremeters are firstly initialized with the old words (eg. the first $\sigma - 1$ words from a corpus of total N words) by applying the batch Gipps sampling, and then updated by only sampling the newly observed words (eg. the words from σ_{th} and on) meanwhile a temporary posterior (which might as well be termed in this thesis) $p(z_i \,|\mathbf{z}_{i-1}, \mathbf{w}_i)$ (with $i = \{\sigma + 1, ..., N\}$, $\mathbf{z}_{i-1} = [z_1, ..., z_{i-1}]$ and $\mathbf{w}_i = [w_1, ..., w_i]$) replaces the true posterior $p(z_j \,\big|\mathbf{z}_{N\setminus j}, \mathbf{w}_N)$ (with $j = \{1, ..., N\}$). Since the performance of "o-LDA" is critically dependent on the accuracy of the initial batch sampling, incremental Gibbs sampler and particle filter are proposed in [31]. Incremental Gibbs sampler discards the batch initialization of "o-LDA" while maintaining its idea of temporary posterior sampling, however, with two changes at every iteration: 1). the topic allocations are sampled for all words (with $i = \{1, ..., N\}$) instead of only the newly observed ones; 2). after the sampling of the i_{th} word, some previous words in $\mathcal{R}_\rangle$ (known as "rejuvenation sequence" in the literature, with $\mathcal{R}_\rangle \subseteq \{1, ..., i\}$) are also to be resampled. Based on the framework of the incremental Gibbs sampler, particle filter for LDA mainly solves the problem of how often the "rejuvenating" resampling actions should happen by introducing the theory of particle filter[55, 54] in which the importance weight vector $\omega_i = [\omega_i^{(1)}, ..., \omega_i^{(p)}, ..., \omega_i^{(P)}]$ and the proposal probability for every topic in terms of each particle p are proposed as $z_i^{(p)}$ ($p = \{1, ..., P\}$). During every iteration, both $z_i^{(p)}$ and $\omega_i^{(p)}$ are to be sampled similar to the procedure of the temporary posterior sampling (see detailed formulations in [31]) and ω_i will be normalized before the checking of whether $\|\omega_i\|^{-2} \leq ESS$ (effective sample size) threshold: if so, the "rejuvenating" resampling takes place; otherwise, it skipps to a new iteration.

As for the application of LDA in the domain of human-robot word-referent learning, a representative example can be found as *Online MLDA* described in [6], where the procedure $\beta \rightarrow \phi_k \rightarrow w_{d,n}$ illustrated in Figure 1.19 is extented to four set of parameters in parallel as $\beta^m \rightarrow \phi_k^m \rightarrow w_{d,n}^m$ ($m \in \{visual, auditory, haptic, word\}$) in order to represent the word allocation per topic of four modalities. Every observed object, which contains four collections of vector quantized features of repespetive modalities, is regarded equivalent to the term "document" and the categorization of an object serves as a topic. The inference algorithm by

using Gipps sampling for batch LDA learining is still in the form as in Equation A.9 except that ϕ_k^m and θ_d should be calculated, considering the information of four different modalities. Adaptations for the online use are summarized as follows:

1) parameter updatings only take place by the sequential sampling of the newly observed input information of an object;

2) a forgetting factor λ is introduced to regulate the influence of previous parameter results (as initial settings) on the learning of new object;

3) particle filter theory is integrated in the topic allocation (z_{mk}) process for a given modality feature word (w_{mk}) during sampling and

4) a model selection method is proposed to find the optimized setting of α, λ, targeting the objective of achieving the best word prediction performance given modal information of visual, acoustic and haptic (ie. maximizing $p(w^w \,|w_{obs}^v, w_{obs}^a, w_{obs}^h)$).

The proposed *Online MLDA*, which models the possible category allocation for every observed object as well as the modal feature distribution per category, is not only used for a refined analysis of the observed object concerning categorization, but also for inferring the category of the unseen object according to $\hat{z} = \arg\max_z p(z \,|w_{obs}^m)$ and predicting descriptive words for unseen objects by solving $\hat{w} = \arg\max_w p(w^w \,|w_{obs}^m)$.

Appendix B

Complementary results for chapter 3

B.1 Determination of k for NMF with reference

We report here the results on the partial *S-OBJ-A* dataset.

1. *2c-2s* data

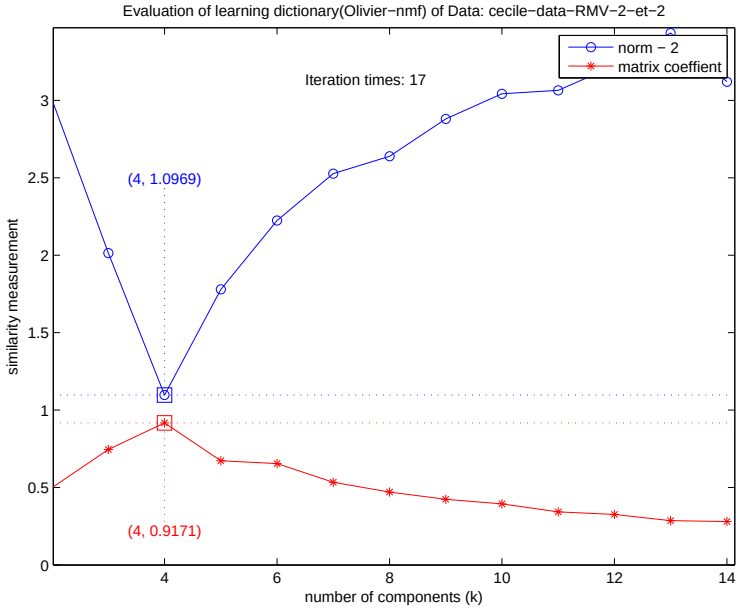


Figure B.1: Evaluation of NMF learned dictionary (2c-2s).

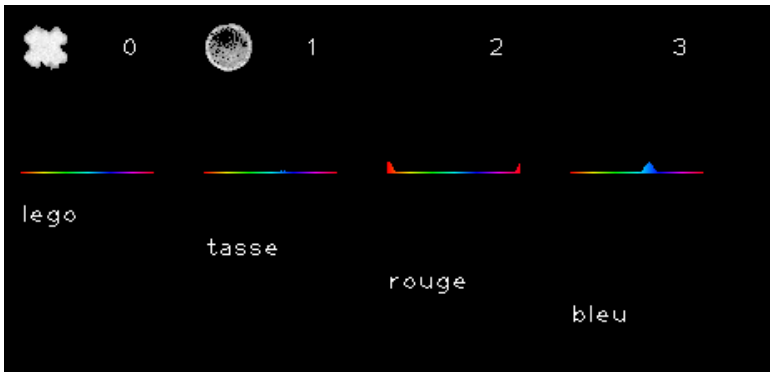


Figure B.2: Visualization of the optimal NMF learned dictionary (2c-2s) in norm-2 criterion.

Figure B.3: Visualization of the optimal NMF learned dictionary (2c-2s) in correlation coefficient criterion.

As similar to the previous case, both the optimal learned dictionaries possess the right number of components and approximate the reference.

2. *3c-3s* data

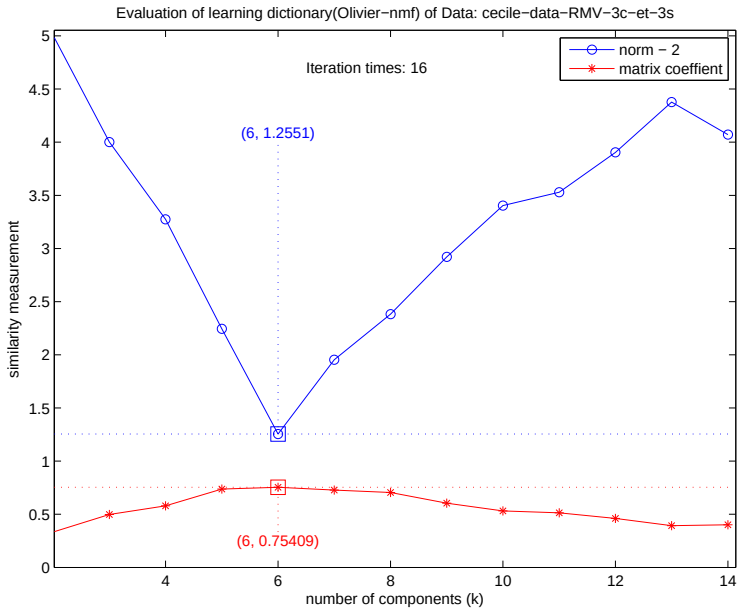


Figure B.4: Evaluation of NMF learned dictionary (3c-3s).

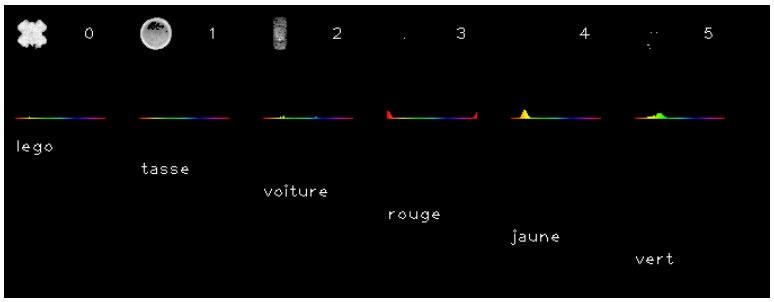


Figure B.5: Visualization of the optimal NMF learned dictionary (3c-3s) in norm-2 criterion.

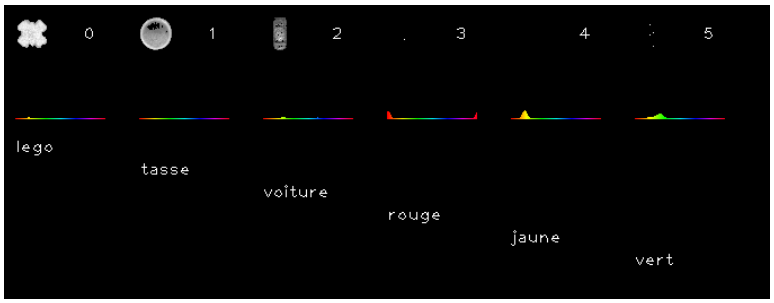


Figure B.6: Visualization of the optimal NMF learned dictionary (3c-3s) in correlation coefficient criterion.

3. $4c$ data

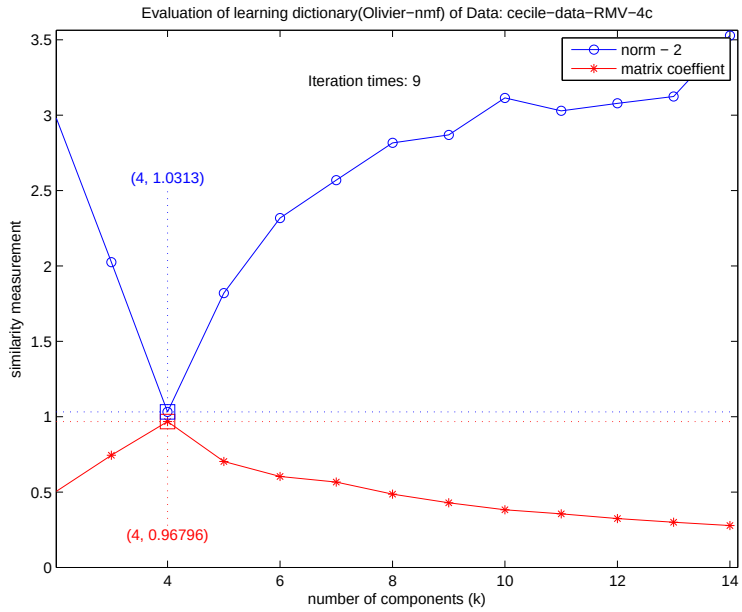


Figure B.7: Evaluation of NMF learned dictionary (4c).

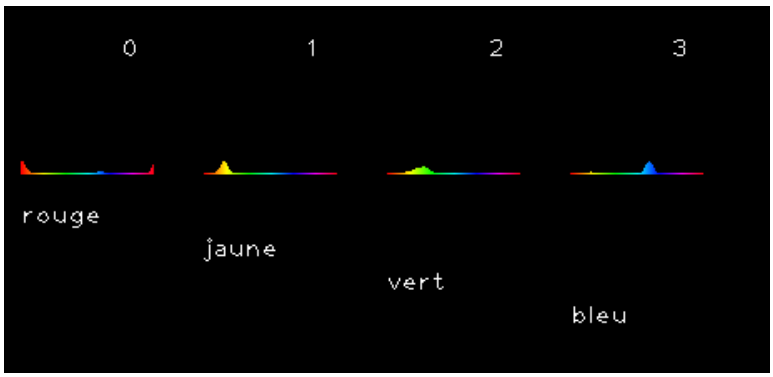


Figure B.8: Visualization of the optimal NMF learned dictionary (4c) in norm-2 criterion.

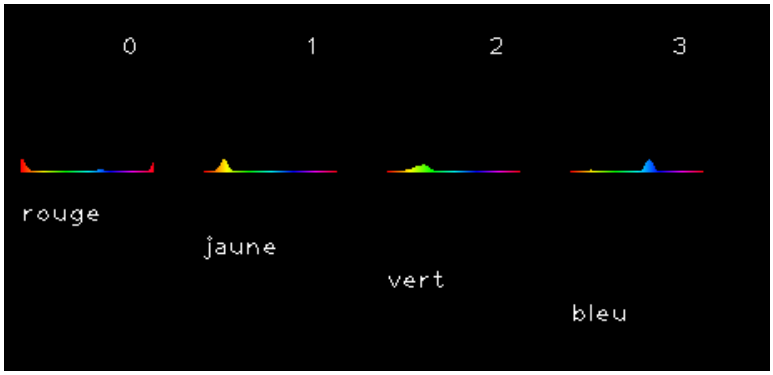


Figure B.9: Visualization of the optimal NMF learned dictionary (4c) in correlation coefficient criterion.

4. *4c-2s* data

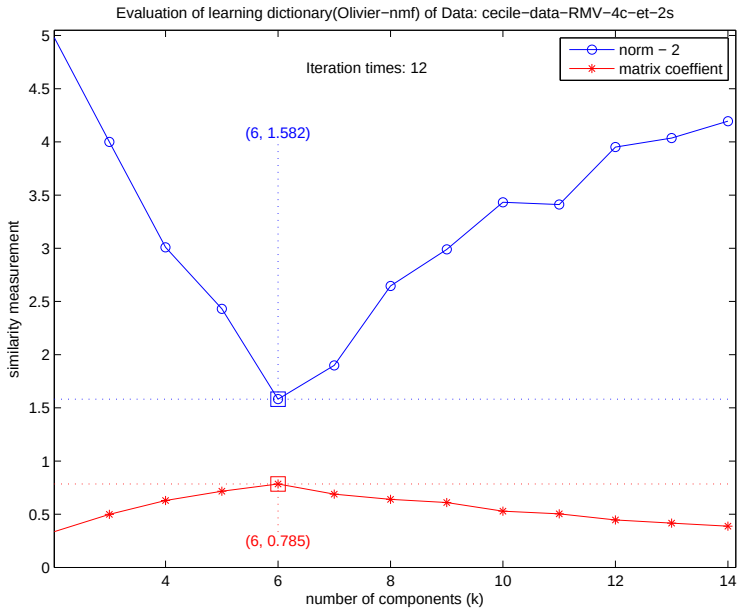


Figure B.10: Evaluation of NMF learned dictionary (4c-2s).

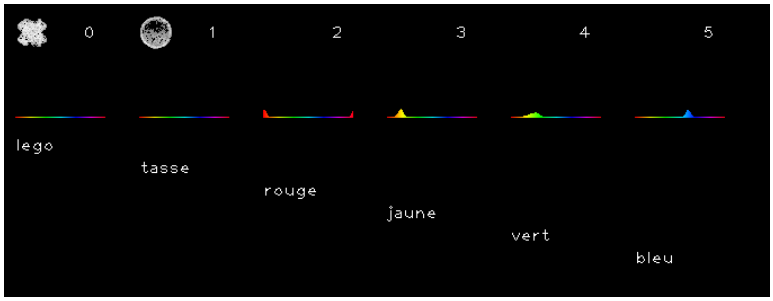


Figure B.11: Visualization of the optimal NMF learned dictionary (4c-2s) in norm-2 criterion.

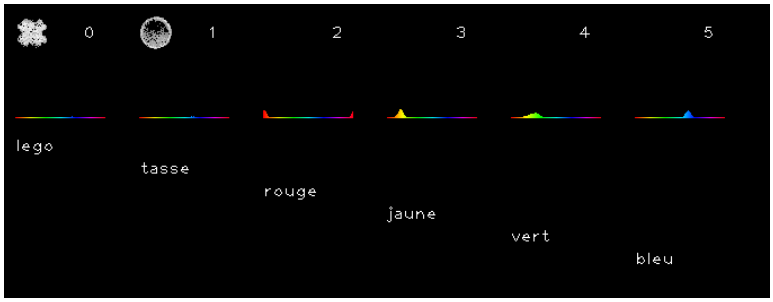


Figure B.12: Visualization of the optimal NMF learned dictionary (4c-2s) in correlation coefficient criterion.

5. *4s* data

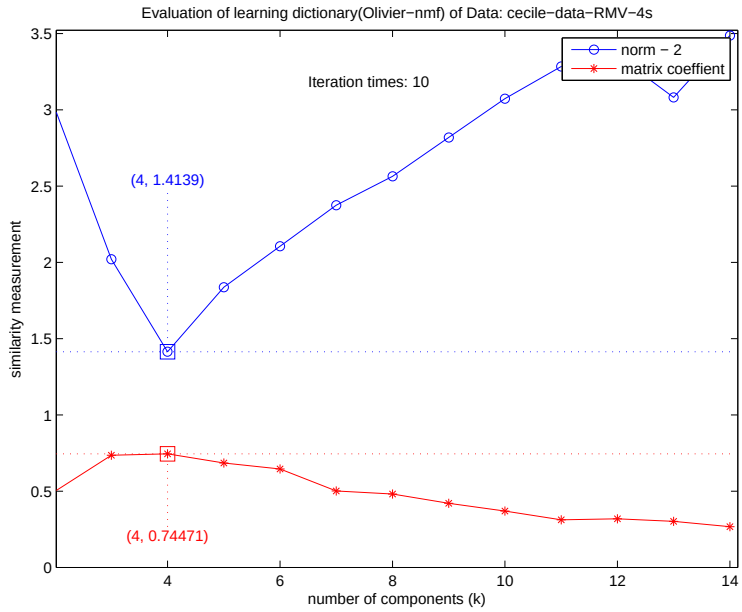


Figure B.13: Evaluation of NMF learned dictionary (4s).

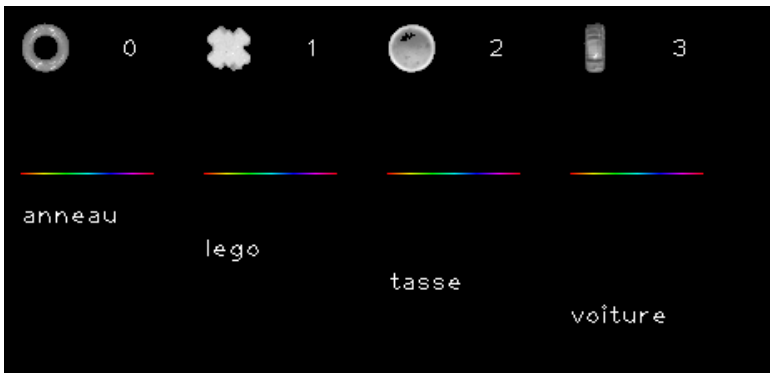


Figure B.14: Visualization of the optimal NMF learned dictionary (4s) in norm-2 criterion.

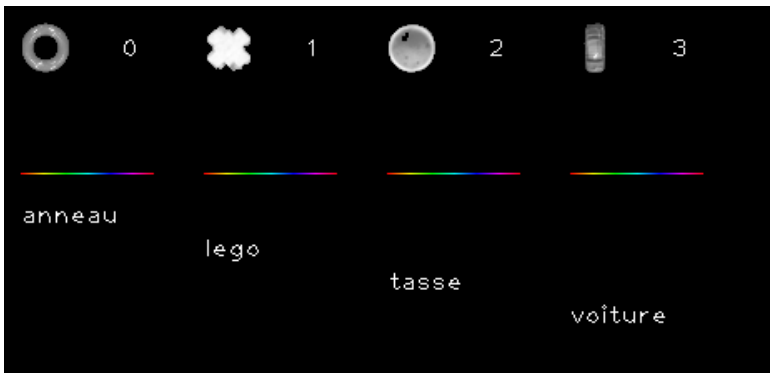


Figure B.15: Visualization of the optimal NMF learned dictionary (4s) in correlation coefficient criterion.

6. *4s-2c* data

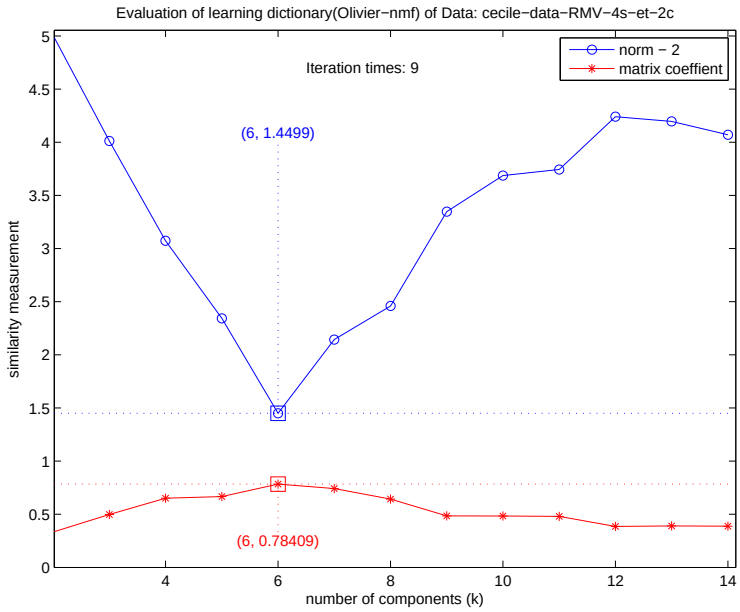


Figure B.16: Evaluation of NMF learned dictionary (4s-2c).

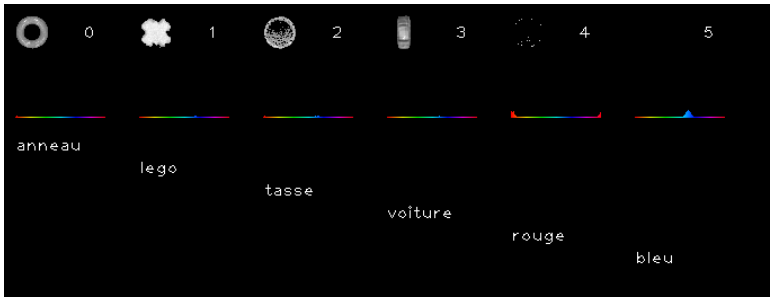


Figure B.17: Visualization of the optimal NMF learned dictionary (4s-2c) in norm-2 criterion.

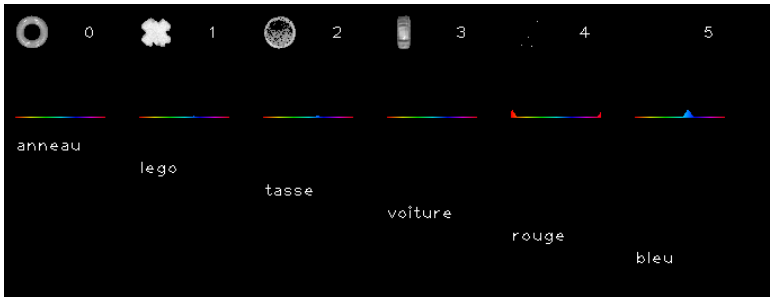


Figure B.18: Visualization of the optimal NMF learned dictionary (4s-2c) in correlation coefficient criterion.

B.2 Evaluation of SV-NMF

We report here the results on the partial *S-OBJ-A* dataset.

1. *2c-2s* data

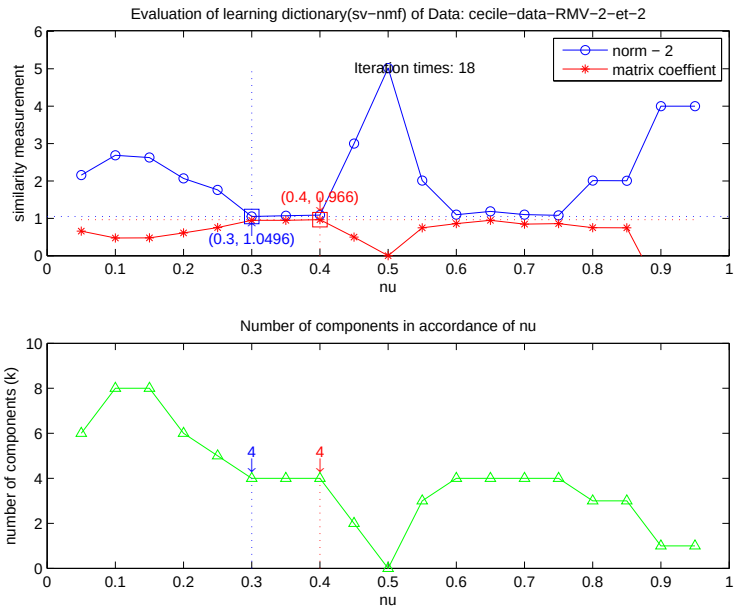


Figure B.19: Evaluation of SV-NMF learned dictionary (2c-2s).

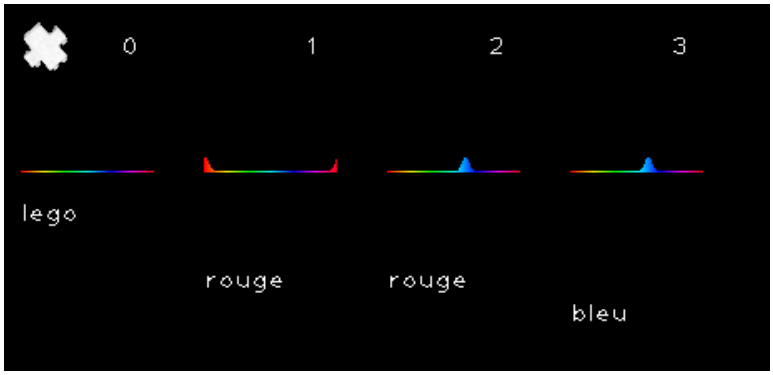


Figure B.20: Visualization of the optimal SV-NMF learned dictionary (2c-2s) in norm-2 criterion.

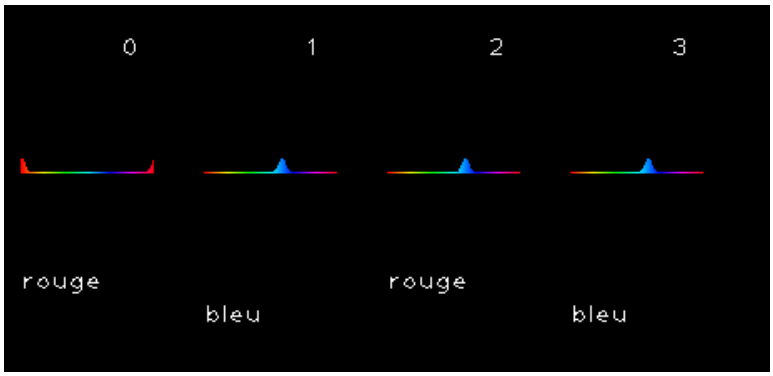


Figure B.21: Visualization of the optimal SV-NMF learned dictionary (2c-2s) in correlation coefficient criterion.

Once again, both criteria curves indicate the right k via two respective nu values as optimals, yet the model fails to produce all the necessary entries with some components repeated and some contradictory, for example *rouge* and *bleu*.

2. *3c-3s* data

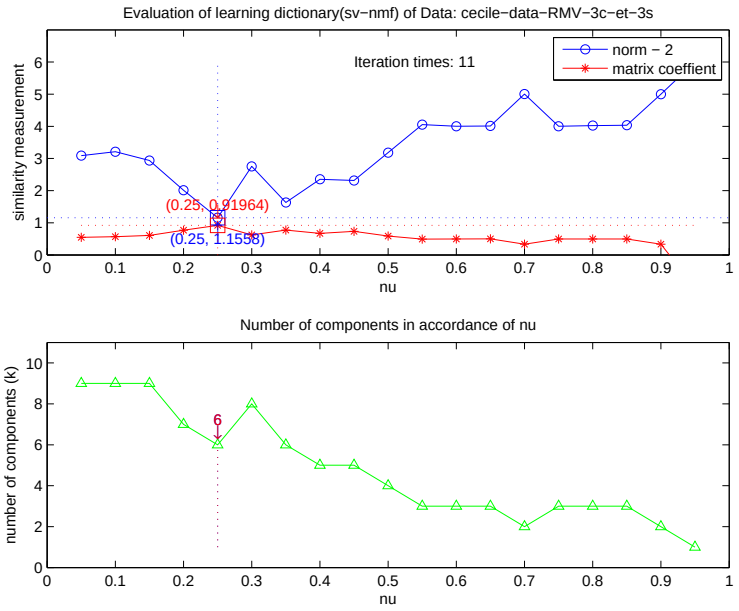


Figure B.22: Evaluation of SV-NMF learned dictionary (3c-3s).

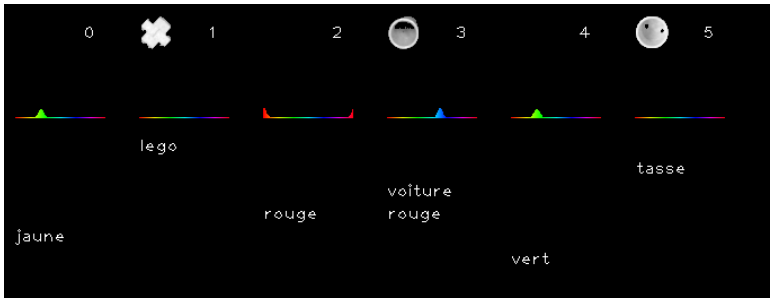


Figure B.23: Visualization of the optimal SV-NMF learned dictionary (3c-3s) in norm-2 criterion.

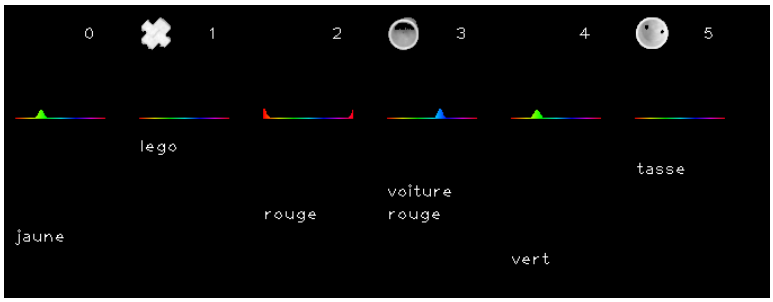


Figure B.24: Visualization of the optimal SV-NMF learned dictionary (3c-3s) in correlation coefficient criterion.

This time both criteria suggest the same nu, thus the same k and the same learned result, in which four compoents just match the referents while two are totally wrong.

3. *4c* data

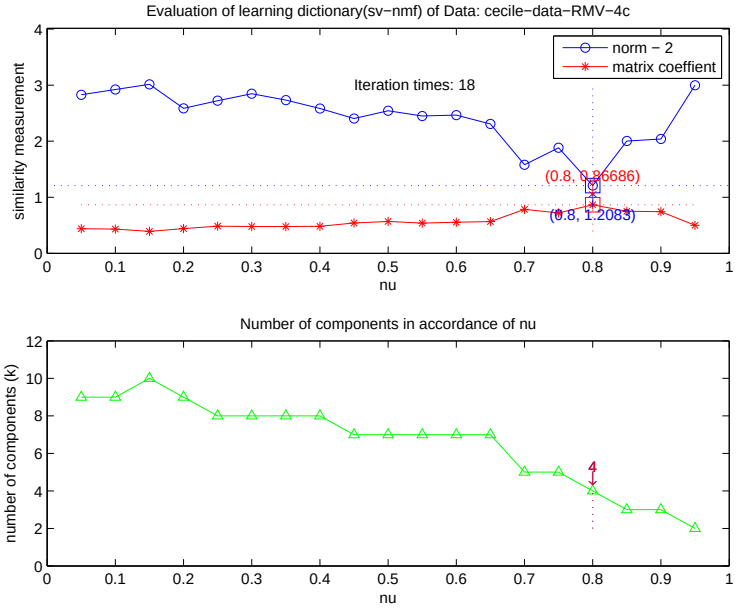


Figure B.25: Evaluation of SV-NMF learned dictionary (4c).

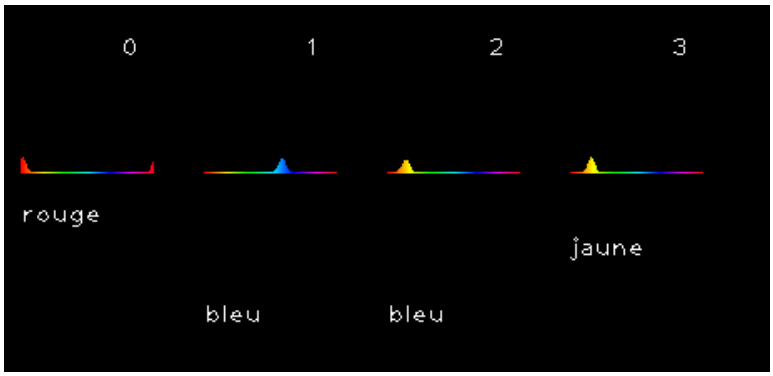


Figure B.26: Visualization of the optimal SV-NMF learned dictionary (4c) in norm-2 criterion.

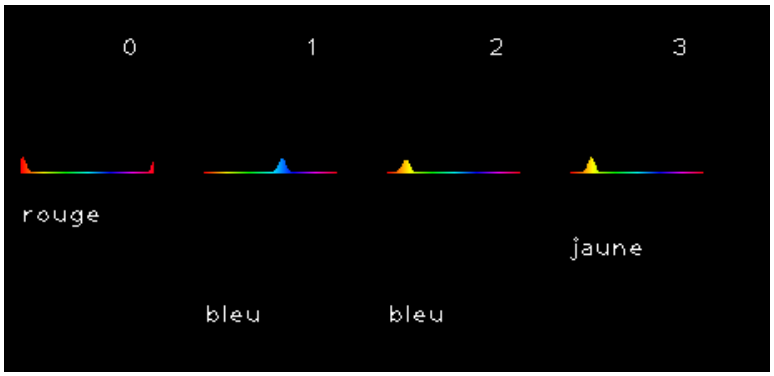


Figure B.27: Visualization of the optimal SV-NMF learned dictionary (4c) in correlation coefficient criterion.

Resembling the previous case, k is correctly estimated but there is one component totally wrong in the learned dictionary.

4. *4c-2s* data

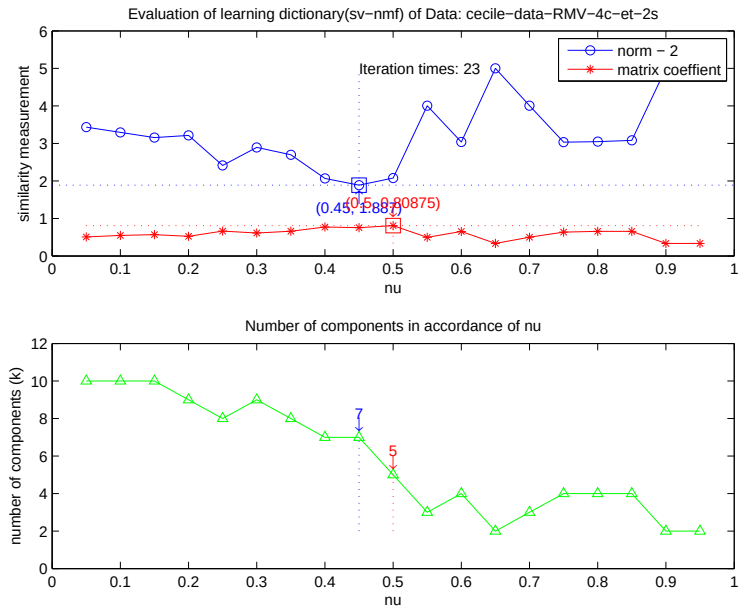


Figure B.28: Evaluation of SV-NMF learned dictionary (4c-2s).

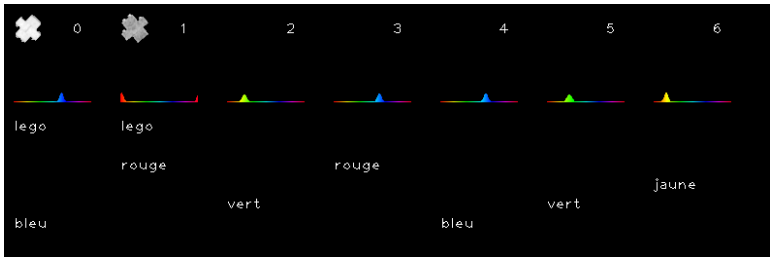


Figure B.29: Visualization of the optimal SV-NMF learned dictionary (4c-2s) in norm-2 criterion.

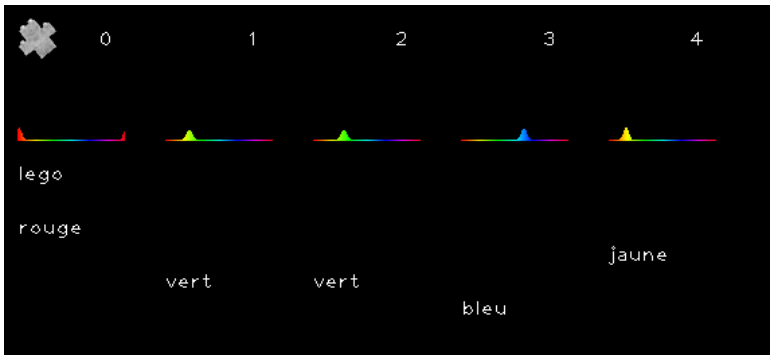


Figure B.30: Visualization of the optimal SV-NMF learned dictionary (4c-2s) in correlation coefficient criterion.

In this case, SV-NMF method gives two different predicted numbers of components according to *norm-2* and *correlation criteria* respectivelly. We can notice at the bottom of Figure B.28 that no parameter of $nu(\nu)$ gives indication to 6, so the algorithm chooses its two neighbours of 5 and 7 as the optimals. The qualities of the resultant dictionaries are not as ideal as before.

5. *4s* data

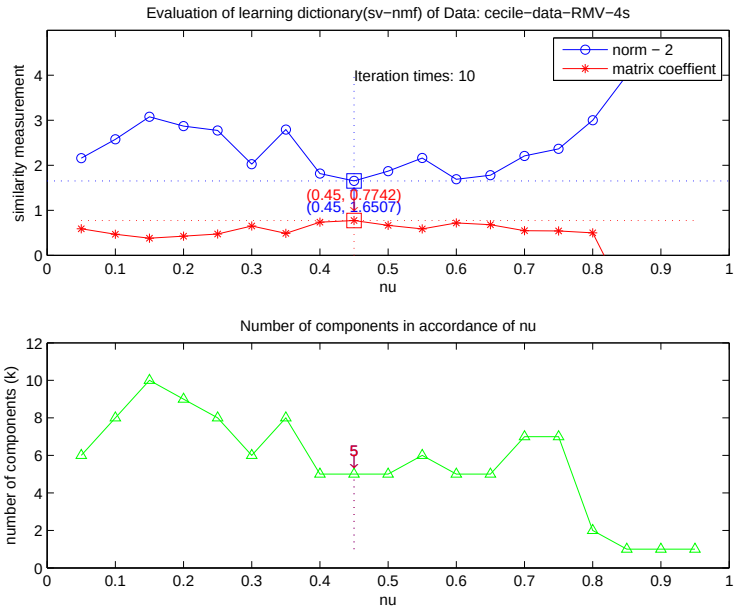


Figure B.31: Evaluation of SV-NMF learned dictionary (4s).

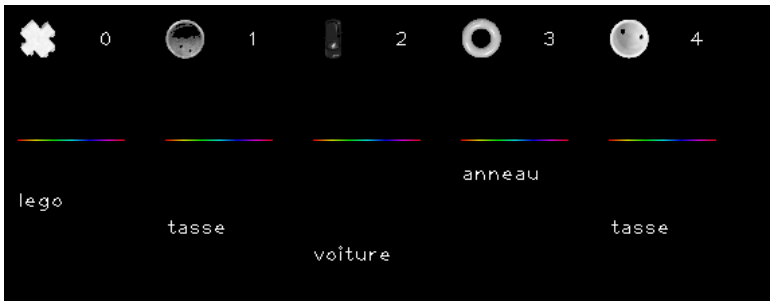


Figure B.32: Visualization of the optimal SV-NMF learned dictionary (4s) in norm-2 criterion.

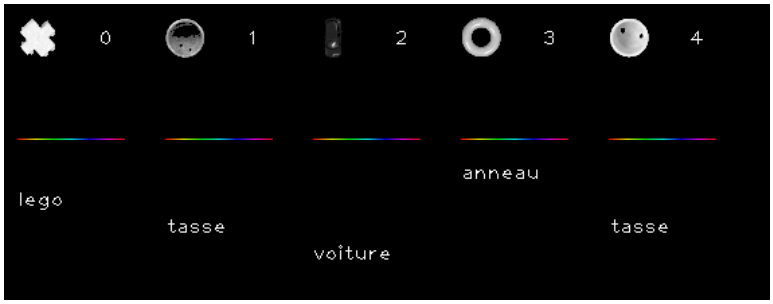


Figure B.33: Visualization of the optimal SV-NMF learned dictionary (4s) in correlation coefficient criterion.

We can observe that since no $nu(\nu)$ corresponds to $k = 4$, the SV-NMF learned dictionary finally has five components, where one concept ("tasse (cup)") is repeated twice.

6. *4s-2c* data

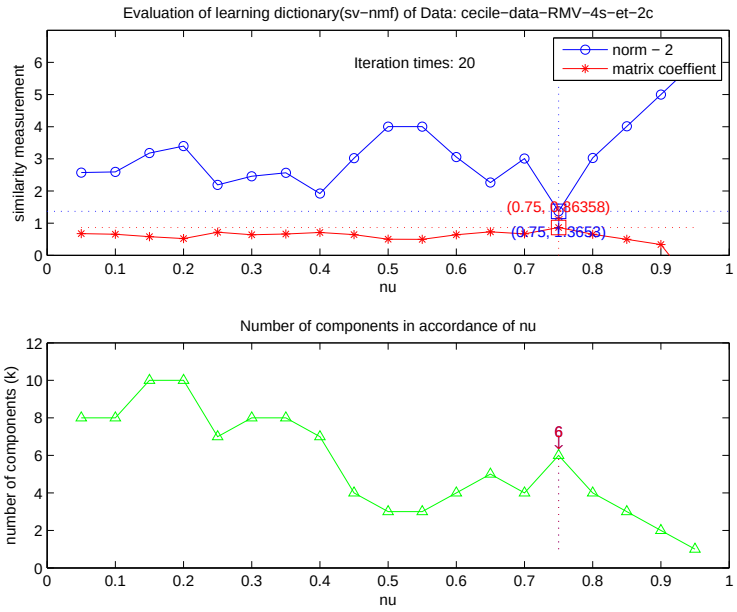


Figure B.34: Evaluation of SV-NMF learned dictionary (4s-2c).

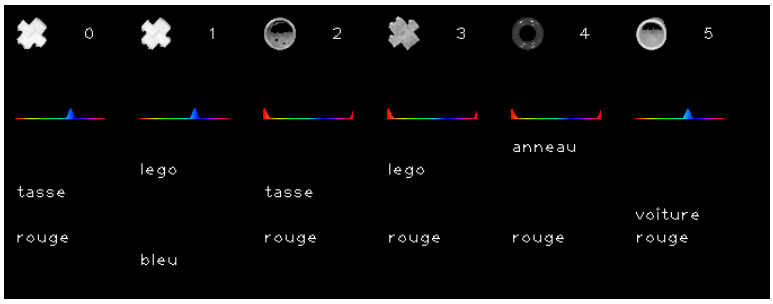


Figure B.35: Visualization of the optimal SV-NMF learned dictionary (4s-2c) in norm-2 criterion.

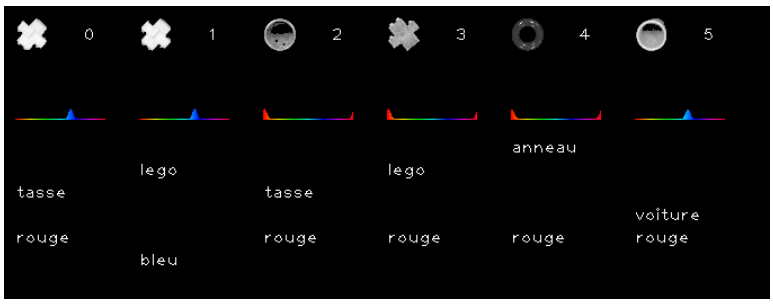


Figure B.36: Visualization of the optimal SV-NMF learned dictionary (4s-2c) in correlation coefficient criterion.

In spite of the number of k correctly estimated, the learned dictionaries possess no pure concepts where two of them are totally wrong.

B.3 Determination of k for NMF without reference

We report here the results on the partial *S-OBJ-A* dataset.

1. *2c-2s* data

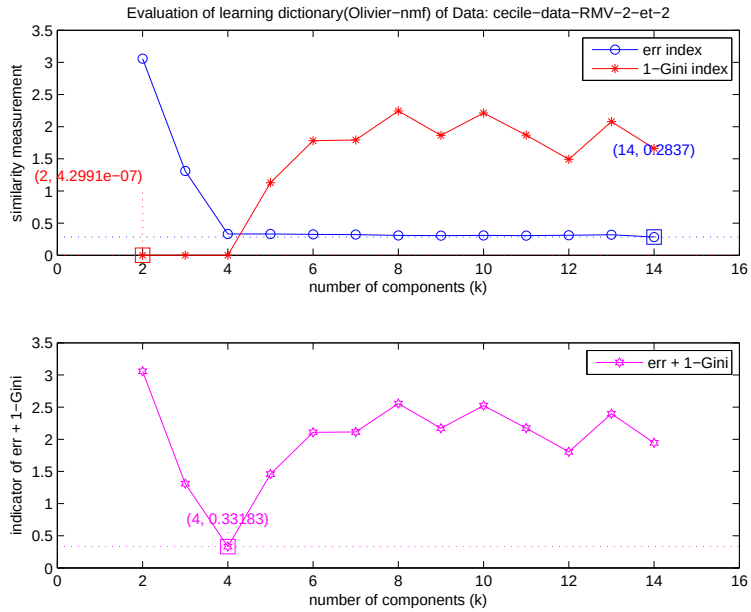


Figure B.37: Evaluation of NMF learned dictionary (2c-2s) in criteria of error and Gini index.

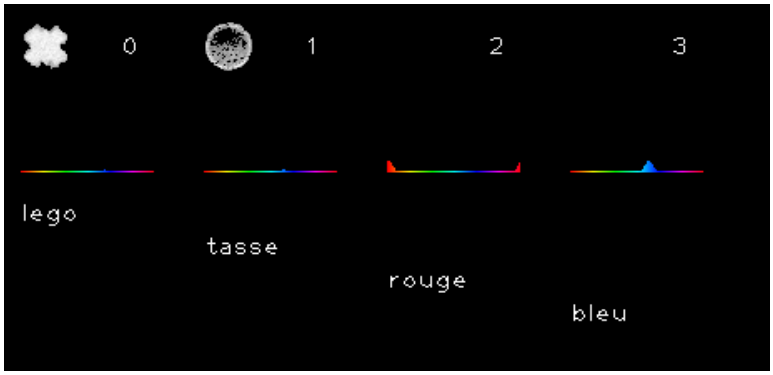


Figure B.38: Visualization of the optimal NMF learned dictionary (2c-2s) in criteria of error and Gini index.

Here we arrive at as similar results as in the previous case.

2. *3c-3s* data

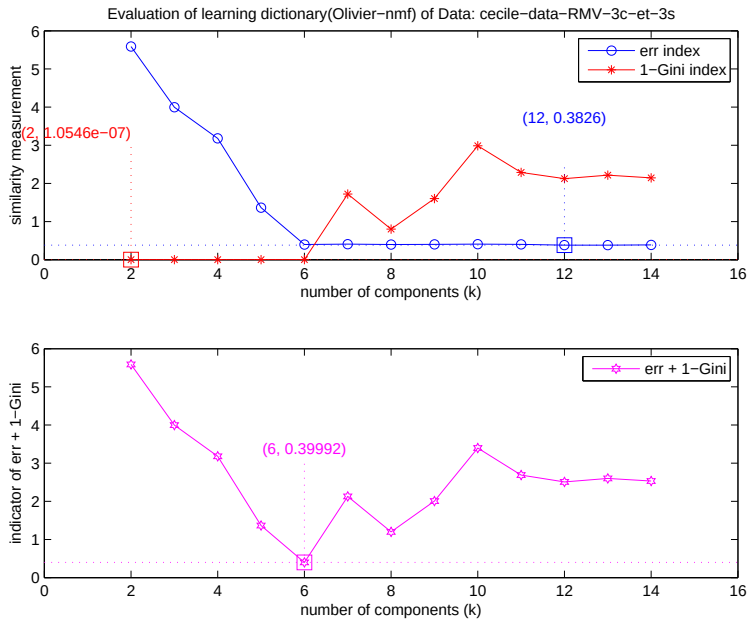


Figure B.39: Evaluation of NMF learned dictionary (3c-3s) in criteria of error and Gini index.

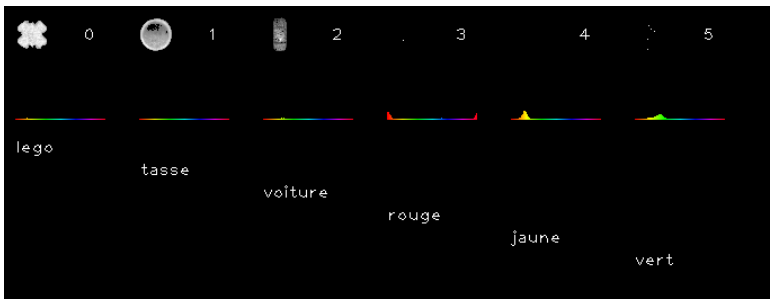


Figure B.40: Visualization of the optimal NMF learned dictionary (3c-3s) in criteria of error and Gini index.

3. *4c* data

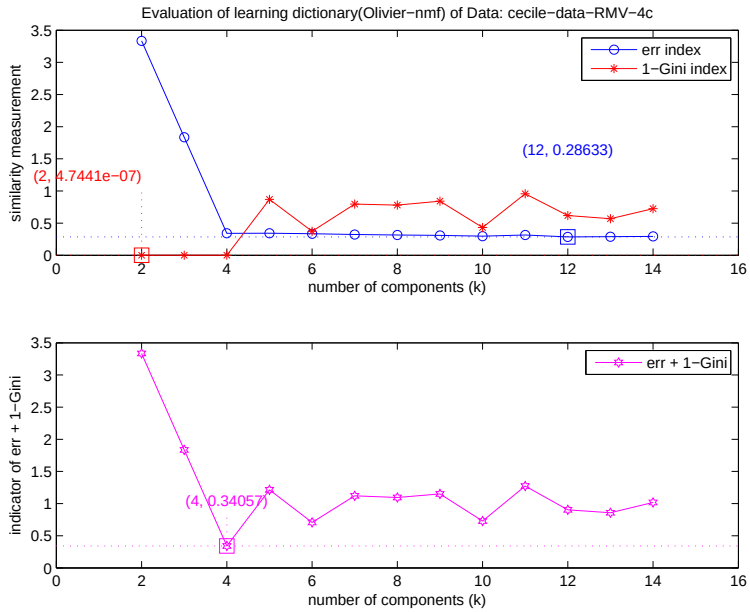


Figure B.41: Evaluation of NMF learned dictionary (4c) in criteria of error and Gini index.

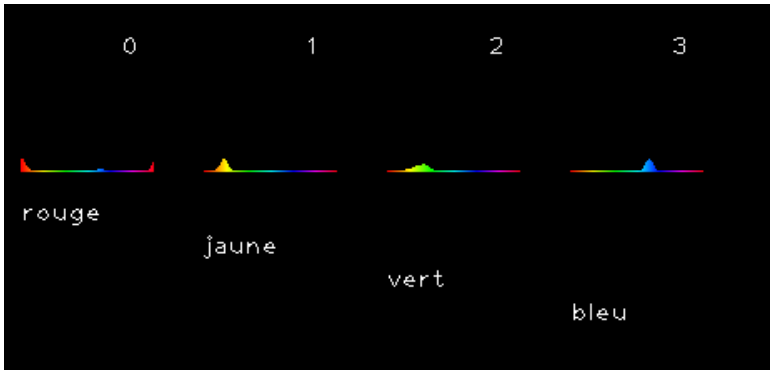


Figure B.42: Visualization of the optimal NMF learned dictionary (4c) in criteria of error and Gini index.

4. *4c-2s* data

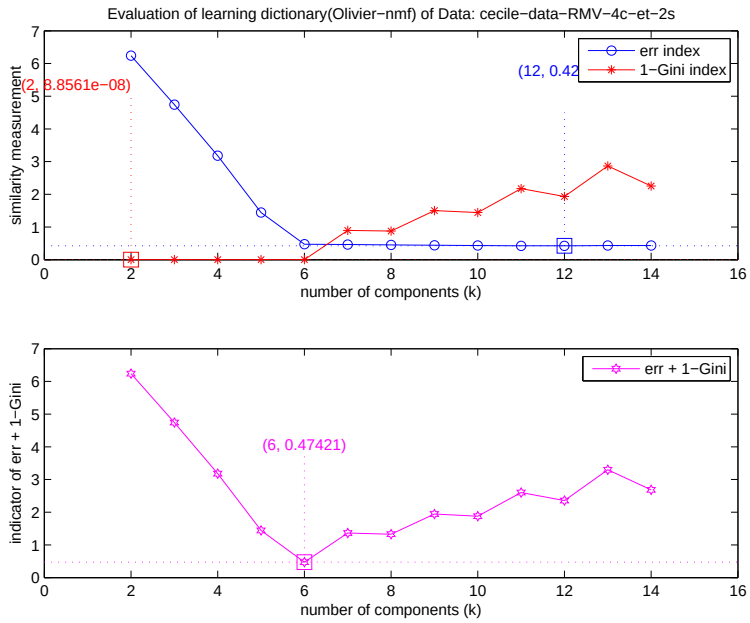


Figure B.43: Evaluation of NMF learned dictionary (4c-2s) in criteria of error and Gini index.

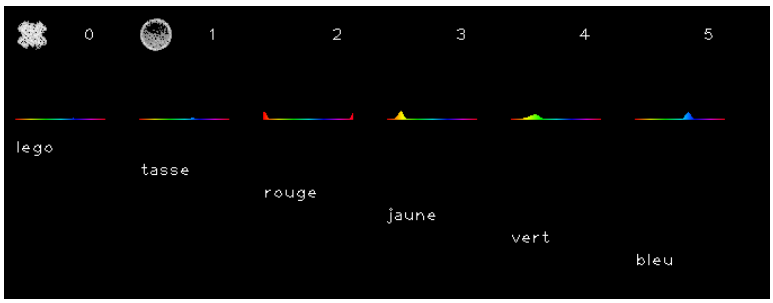


Figure B.44: Visualization of the optimal NMF learned dictionary (4c-2s) in criteria of error and Gini index.

5. *4s* data

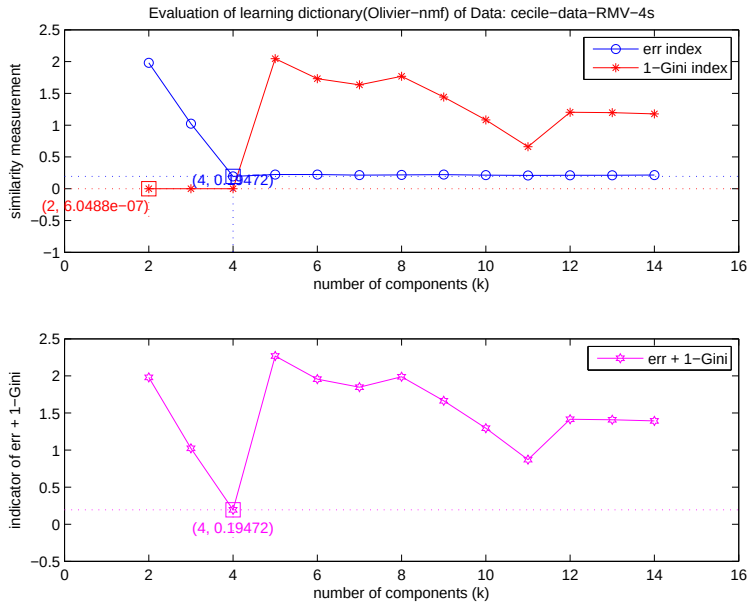


Figure B.45: Evaluation of NMF learned dictionary (4s) in criteria of error and Gini index.

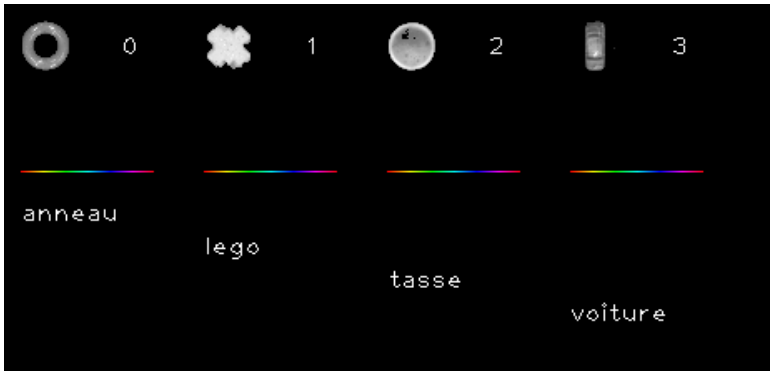


Figure B.46: Visualization of the optimal NMF learned dictionary (4s) in criteria of error and Gini index.

6. *4s-2c* data

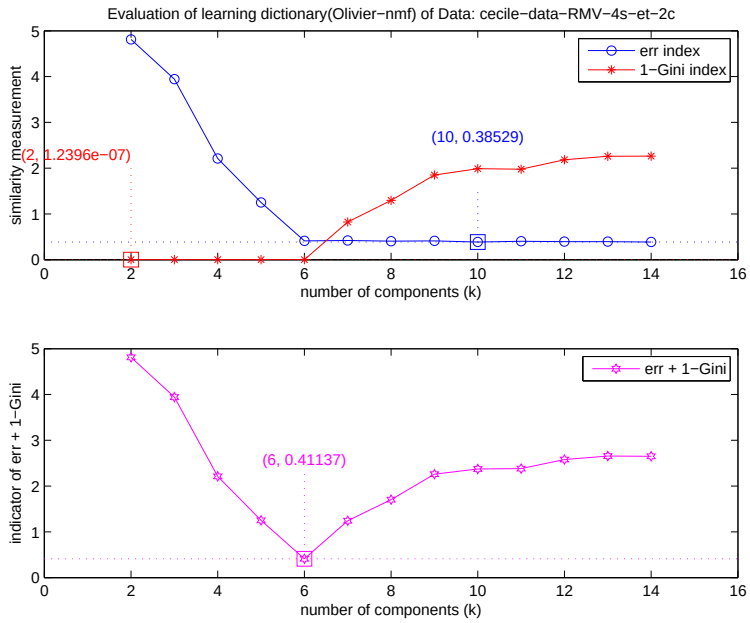


Figure B.47: Evaluation of NMF learned dictionary (4s-2c) in criteria of error and Gini index.

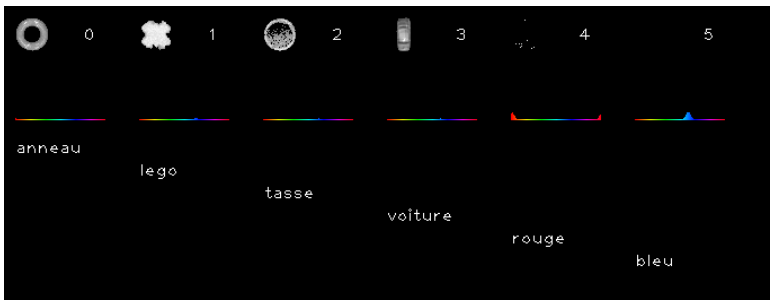


Figure B.48: Visualization of the optimal NMF learned dictionary (4s-2c) in criteria of error and Gini index.

Appendix C

Complementary experimental settings for a complete human-robot intractive learning

In collaboration with Fabio Pardo during his Master internship, we developed an object learning demonstration during human robot interaction.

C.1 Meka robot

This experiment of object learning was conducted on a humanoid robot Meka with interactions with human experimenters. Figure C.1 illustrates the overall setting scenario of the experiment.

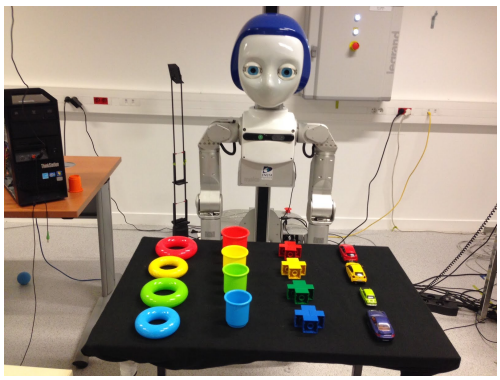


Figure C.1: Meka, the operation platform and a set of sample objects to be learned in the experiment.

Expressively designed for the human-robot interaction by Meka Robotics, this humanoïd robot is composed of a head, a pair of eyes with two cameras inside, a torso with a depth-camera, two arms and a mobile base on omnidirectional wheels. It can perform motion control and object learning by detecting objects and interacting with humans in the front.

The most shining feature of Meka, especially on contrast to the industrial robots, is its compliant and force control. The idea is to let the actions of actuators be under the control of spring, thus 1). on one hand exerting exact force and torque to accomplish a motion in

accordance with the controlling of precise displacement based on inverse dynamics, and 2). on the other hand, by setting appropriate stiffness parameters of spring, different modes of human-robot interactions can be achieved, for example, the robot can maintain a zombie-like posture (high stiffness) or resilient enough to move back and forth when contacting with a human tutor.

As for the software framework, Meka is equipped with real-time control software with ROS extensions. ROS (robot operating system) is an open-sourced system operation frameworks under BSD licence. It is a system of nodes and communications, providing operating system-like functionality on a heterogeneous computer cluster, including hardware abstraction, peripheral equipment drivers, libraries, visualizers for simulations, message transitions, package management, and more. During the experiment, every Python script is run on a node and has the possibility to create topics where messages can be published to other nodes, while other nodes then have the opportunity to subscribe to these topics and thus be notified to the publication of each new message. These nodes can be run on different computers if they are connected to the same Ethernet hub.

An auxiliary camera, installed in the torso, is an Xtion Pro Live-Asus, the same type as the Microsoft Kinect with an infrared sensor to assess the depth and efficiently identify human forms. It is used to make a connection between the spatial position of the robot and the spatial position of the head of the human teacher. Hence, by adjusting the angles of the robot's neck to watch the tutor's eyes, the human-robot interaction will be more natural as interpersonal one.

Finally, all procedures of the experiment are filmed by a webcam, fixed on a tripod on the left of the robot and human tutor.

C.2 Interaction protocol

C.2.0.1 Robot

We performed experiments in order to study the effect of the feedback from the robot (regarded as the reflection of its current learning state) on the human teacher's behavior.

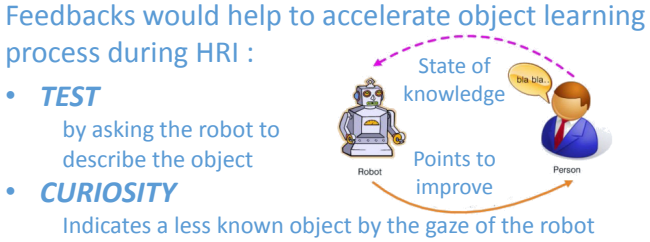


Figure C.2: Two basic approaches during HRI (human robot interaction)

In fact, two basic capabilities (see Figure C.2) are applied in the human robot interaction: giving ***test*** on the part of a human tutor and showing ***curiosity*** on the part of a robot.

Therefore, four sorts of scenarios of interactive learning are devised, as depicted in Table Equation C.1.

	With curiosity of robot	Without curiosity
With test from tutor	Scenario I	Scenario II
Without test	Scenario III	Scenario IV

Table C.1: Four sorts of interactions.

For the realization of a true interaction, both the robot and human tutor should acquire the abilities to express and receive information.

On the part of robot, its actions include the following cases:

```
1.  If a movement is detected on the table
      look in the direction of the moving point that is the nearest to the robot
      If an object is detected in vicinity of the moving point when it comes
  to stop
        Watch the detected object for 3 seconds and wait for the key words
        If the key words are detected
          Learn about this new example
          Say "d'accord (OK)"
    Else
      Look at the head of human tutor
2.  If curiosity about an object is beyond a certain level
      Watch the object
3.  If the keyword is about "question"
      Test the designated object
      Select the words reconstructed by the order of weight in the histogram
      Keep the words whose weights are greater than a certain level
      Utter "je dirais (I think it is)" and the resultant words in a descending
  order of weight.
```

For every mode of the iterative learning in Table Equation C.1, the lasting time is 10 minutes (indicated on the computer screen), within which the human tutor can give instructions and receive feedbacks from the robot.

Among them, its vocal utterance (repeating words during training and speaking the identification result during testing) is made available via Google TTS (text-to-speech) API, which automatically generates a wav file easily obtained with an HTTP request through any .net programming. And the input text source is just from the coefficient matrix H whose column vectors will tell the weight of concepts stored in W.

C.3 Human tutor

The human tutor is sitting on a chair in front of Meka, and a table covered in black is placed in between (see Figure C.3). The 16 objects in total are randomly put on a big table on the left side of the human participant. While the participant can take some objects to show to the robot on the black table, a monitor located a little bit behind the robot shows the remaining time of the current mode (among four scenarios) of interaction.

Figure C.3: The interaction during experiment.

After being informed about the rules of the experiment, every participant is given 15 minutes practice so as to be more familiar with experimental operations.

Then begins the experiment, comprising of four distinct phases of ten minutes, during which the participants have to teach the robot the eight concepts regarding shape (four words) and color (four words) using sixteen objects available on a table left to him. The remaining time of each phase is displayed on a monitor on their left. For each phase, the memory of the robot will be initialized as blank for another round of learning.

For example during training, the human tutor puts four objects on the front table, then points to one of the object. Tracking the finger tip of the tutor, the eyes of the robot will finally focus on the pointed object. The tutor would say "Voici un lego jaune (Here is a yellow lego)", waiting for the confirmation from the robot repeating the key words - "D'accord - lego - jaune". When this confirmation of words is well done, the tutor can go on with other objects or retrain the same object if only partial words or even no words are repeated.

As for testing, the tutor points to one designated object, which will be tracked and focused by robot's eyes, and then propose a question like "C'est quoi ça (What's this)?". Then the multimodal information of this object are acquired and processed for the calculation of NMF. By decomposing the input data V_i into linear combination of basis in W, the corresponding coefficient vector H_i will be computed. So the robot will use the Google TTS API to speak the words with the highest value (larger than a threshold) in H_i. And by judging if the robot speak the right words in terms of shape and color of the object, the tutor gets feedback, which will help him to improve the training in the subsequent grounds.

The experiment ends when the participant finishes all the training and testing operations

of the four phases. Finally, every participant will fill in a questionnaire about the personal status (eg. current understanding about artificial/intelligent learning, the goal of conducting the experiment, the strategy during training, the remark on the role of curiosity and feedback) for other further psychological studies.

The results of these experiments have been presented to the "Mechanisms of learning in social contexts" workshop at the ICDL 2015 conference [141].

Bibliography

[1] Lauren B. Adamson and Roger Bakeman. "The development of shared attention during infancy." In: (1991) (cit. on p. 11).

[2] Henny Admoni et al. "Deliberate delays during robot-to-human handovers improve compliance with gaze communication." In: *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*. ACM. 2014, pp. 49–56 (cit. on p. 11).

[3] N. Akhtar and L. Montague. *Early lexical acquisition: the role of cross-situational learning*. 1999 (cit. on p. 3).

[4] Toomas Altosaar et al. "A Speech Corpus for Modeling Language Acquisition: CAREGIVER." In: *LREC*. 2010 (cit. on p. 13).

[5] Takaya Araki et al. "Autonomous acquisition of multimodal information for online object concept formation by robots." In: *IEEE International Conference on Intelligent Robots and Systems*. 2011, pp. 1540–1547 (cit. on pp. 11–14, 16, 18, 21, 96, 147).

[6] Takaya Araki et al. "Online object categorization using multimodal information autonomously acquired by a mobile robot." In: *Advanced Robotics* 26.17 (2012), pp. 1995–2020 (cit. on p. 152).

[7] Jushan Bai and Serena Ng. *Determining the Number of Factors in Approximate Factor Models*. 2002 (cit. on p. 36).

[8] A. Bandura and R. H. Walters. *Social learning and personality development*. 1963, pp. 1–62 (cit. on p. 3).

[9] Albert Bandura. "Social learning theory." In: *Social Learning Theory*. 1971, pp. 1–46 (cit. on p. 3).

[10] Arindam Banerjee and Sugato Basu. "Topic Models over Text Streams: A Study of Batch and Online Unsupervised Learning." In: *SDM*. Vol. 7. SIAM. 2007, pp. 437–442 (cit. on p. 152).

[11] Paresh Chandra Barman and Soo Young Lee. "Tree cluster of text data by NMF based neural network." In: *Proceedings of 4th International Conference on Electrical and Computer Engineering, ICECE 2006*. 2007, pp. 312–315 (cit. on p. 33).

[12] Herbert Bay et al. "Speeded-up robust features (SURF)." In: *Computer vision and image understanding* 110.3 (2008), pp. 346–359 (cit. on p. 17).

[13] Yoshua Bengio. "Learning deep architectures for AI." In: *Foundations and trends® in Machine Learning* 2.1 (2009), pp. 1–127 (cit. on p. 27).

[14] Yoshua Bengio, Aaron Courville, and Pascal Vincent. "Representation learning: A review and new perspectives." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35 (2013), pp. 1798–1828. arXiv: `arXiv:1206.5538v2` (cit. on p. 27).

[15] Daniel E. Berlyne. *Conflict, arousal, and curiosity*. McGraw-Hill Book Company, 1960 (cit. on p. 45).

[16] Michael W. Berry et al. "Algorithms and applications for approximate nonnegative matrix factorization." In: *Computational Statistics & Data Analysis* 52 (2007), pp. 155–173 (cit. on p. 33).

[17] Michael W. Berry et al. "Algorithms and applications for approximate nonnegative matrix factorization." In: *Computational Statistics & Data Analysis* 52 (2007), pp. 155–173 (cit. on p. 35).

[18] David M. Blei. "Probabilistic Topic Models." In: *Commun. ACM* 55.4 (Apr. 2012), pp. 77–84 (cit. on p. 23).

[19] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. "Latent Dirichlet Allocation." In: *J. Mach. Learn. Res.* 3 (Mar. 2003), pp. 993–1022 (cit. on p. 22).

[20] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. "Latent Dirichlet Allocation." In: *Journal of Machine Learning Research* 3.4-5 (2012). Ed. by John Lafferty, pp. 993–1022. arXiv: `1111.6189v1` (cit. on pp. 42, 43, 149, 150).

[21] David M. Blei et al. "Hierarchical Topic Models and the Nested Chinese Restaurant Process." In: *Advances in Neural Information Processing Systems*. MIT Press, 2004, p. 2003 (cit. on p. 147).

[22] Anna Bosch, Andrew Zisserman, and Xavier Muñoz. "Scene classification via pLSA." In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 3954 LNCS. 2006, pp. 517–530 (cit. on p. 41).

[23] Louis ten Bosch et al. "Unsupervised detection of words–questioning the relevance of segmentation." In: *ISCA ITRW, Speech Analysis and Processing for Knowledge Discovery*. Citeseer. 2008 (cit. on p. 15).

[24] Cynthia Breazeal et al. "Effects of nonverbal communication on efficiency and robustness in human-robot teamwork." In: *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*. IEEE. 2005, pp. 708–713 (cit. on p. 11).

[25] Susan E. Brennan et al. "Coordinating cognition: The costs and benefits of shared gaze during collaborative search." In: *Cognition* 106.3 (2008), pp. 1465–1477 (cit. on p. 11).

[26] J. P. Brunet et al. "Metagenes and molecular pattern discovery using matrix factorization." In: *Proc Natl Acad Sci U S A* 101 (2004), pp. 4164–4169 (cit. on p. 33).

[27] Sylvain Calinon and Aude G. Billard. "What is the Teacher ' s Role in Robot Programming by Demonstration? Toward Benchmarks for Improved Learning." In: *Science* 8 (2007), pp. 441–464 (cit. on p. 2).

[28] Josep Call and Malinda Carpenter. "Three sources of information in social learning." In: *Imitation in animals and artifacts*. 2002, pp. 211–228 (cit. on p. 2).

[29] Angelo Cangelosi. "The grounding and sharing of symbols." In: *Pragmatics & Cognition* 14.2 (2006), pp. 275–285 (cit. on p. 12).

[30] J.F. Fontanari Cangelosi and A. "Cross-situational and supervised learning in the emergence of communication." In: *Interaction Studies* 12.1 (2011), pp. 119–133 (cit. on p. 3).

[31] Kevin R. Canini, Lei Shi, and Thomas L. Griffiths. "Online Inference of Topics with Latent Dirichlet Allocation." In: *Proceedings of the International Conference on Artificial Intelligence and Statistics* (2009), pp. 65–72 (cit. on p. 152).

[32] Raymond B. Cattell. "The scree test for the number of factors." In: *Multivariate behavioral research* 1.2 (1966), pp. 245–276 (cit. on p. 36).

[33] Pramod Chandrashekhariah, Gabriele Spina, and Jochen Triesch. "Let it Learn-A Curious Vision System for Autonomous Object Learning." In: *VISAPP (2)*. 2013, pp. 169–176 (cit. on p. 4).

[34] Yuxin Chen, Jean-Baptiste Borde, and David Filliat. "A comparative study on active learning behavior in word-meaning association acquisition between human and learning machine." In: *ICDL-EpiRob 2016 Workshop on Language Learning*. IEEE. 2016 (cit. on p. 7).

[35] Yuxin Chen, Jean-Baptiste Bordes, and David Filliat. "An experimental comparison between NMF and LDA for active cross-situational object-word learning." In: *ICDL EPIROB 2016*. 2016 (cit. on p. 7).

[36] Yuxin Chen and David Filliat. "Cross-situational noun and adjective learning in an interactive scenario." In: *2015 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*. IEEE. 2015, pp. 129–134 (cit. on p. 7).

[37] Yong Choon Cho and Seungjin Choi. "Nonnegative features of spectro-temporal sounds for classification." In: *Pattern Recognition Letters* 26 (2005), pp. 1327–1336 (cit. on p. 33).

[38] Andrzej Cichocki, Rafal Zdunek, and Shun-Ichi Amari. "New Algorithms for Non-Negative Matrix Factorization in Applications to Blind Source Separation." In: *2006 IEEE International Conference on Acoustics Speed and Signal Processing*. 2006, pp. V–621–V–624 (cit. on p. 33).

[39] David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan. "Active learning with statistical models." In: *Journal of artificial intelligence research* (1996) (cit. on p. 46).

[40] Silvia Coradeschi, Amy Loutfi, and Britta Wrede. "A short review of symbol grounding in robotic and intelligent systems." In: *KI-Künstliche Intelligenz* 27.2 (2013), pp. 129–136 (cit. on p. 12).

[41] Giovanni Costantini, Renzo Perfetti, and Massimiliano Todisco. "Recurrent neural network for approximate nonnegative matrix factorization." In: *Neurocomputing* 138 (2014), pp. 238–247 (cit. on p. 33).

[42] Kenny R. Coventry and Simon C. Garrod. *Saying, seeing and acting: The psychological semantics of spatial prepositions*. Psychology Press, 2004 (cit. on p. 10).

[43] Thomas M. Cover and Joy A. Thomas. *Elements of information theory*. John Wiley & Sons, 2012 (cit. on pp. 14, 25).

[44] Mihaly Csikszentmihalyi. "Flow: The psychology of optimal performance." In: *NY: Cambridge UniversityPress* (1990) (cit. on p. 45).

[45] Navneet Dalal and Bill Triggs. "Histograms of oriented gradients for human detection." In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 1. IEEE. 2005, pp. 886–893 (cit. on p. 19).

[46] L. Davis. *Handbook of Genetic Algorithms*. 1991, pp. 1–6 (cit. on p. 2).

[47] Peter Dayan and Bernard W. Balleine. "Reward, motivation, and reinforcement learning." In: *Neuron* 36.2 (2002), pp. 285–298 (cit. on p. 45).

[48] Gedeon O. Deák, Ian Fasel, and Javier Movellan. "The emergence of shared attention: Using robots to test developmental theories." In: *Proceedings 1st International Workshop on Epigenetic Robotics: Lund University Cognitive Studies*. Vol. 85. 2001 (cit. on p. 11).

[49] Arthur P. Dempster. "Upper and lower probabilities induced by a multivalued mapping." In: *Studies in Fuzziness and Soft Computing* 219 (2008), pp. 57–72 (cit. on p. 2).

[50] Inderjit S. Dhillon and Dharmendra S. Modha. "Concept decompositions for large sparse text data using clustering." In: *Machine Learning* 42 (2001), pp. 143–175 (cit. on p. 33).

[51] Chris Ding, Tao Li, and Wei Peng. "Nonnegative matrix factorization and probabilistic latent semantic indexing: Equivalence chi-square statistic, and a hybrid method." In: *National conference on artificial intelligence* (2006), pp. 342–347 (cit. on p. 41).

[52] Chris Ding, Tao Li, and Wei Peng. *On the equivalence between Non-negative Matrix Factorization and Probabilistic Latent Semantic Indexing*. 2008 (cit. on p. 41).

[53] Marco Dorigo and Thomas Stützle. "Ant Colony Optimization." In: *IEEE Computational Intelligence Magazine* 1 (2004), pp. 28–39 (cit. on p. 2).

[54] Arnaud Doucet, Nando De Freitas, and Neil Gordon. "An introduction to sequential Monte Carlo methods." In: *Sequential Monte Carlo methods in practice*. Springer, 2001, pp. 3–14 (cit. on p. 152).

[55] Arnaud Doucet et al. "Rao-Blackwellised particle filtering for dynamic Bayesian networks." In: *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc. 2000, pp. 176–183 (cit. on p. 152).

[56] Joris Driesen. "Discovering words in speech using matrix factorization." In: *KU Leuven, ESAT* (2012) (cit. on p. 15).

[57] Deci Edward and Richard Ryan. "Intrinsic Motivation and Self-Determination in Human Behavior." In: *New York: Pantheon* (1985) (cit. on p. 45).

[58] Slim Essid. "A single-class SVM based algorithm for computing an identifiable NMF." In: *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE. 2012, pp. 2053–2056 (cit. on pp. 36, 69, 82, 86).

[59] Ian Fasel et al. "Combining embodied models and empirical research for understanding the development of shared attention." In: *Development and Learning, 2002. Proceedings. The 2nd International Conference on*. IEEE. 2002, pp. 21–27 (cit. on p. 11).

[60] Valerii Vadimovich Fedorov. *Theory of optimal experiments*. Elsevier, 1972 (cit. on p. 46).

[61] Christiane Fellbaum. "WordNet and wordnets." In: *Encyclopedia of Language and Linguistics*. Ed. by (Editor-in-Chief) Keith Brown. Oxford: Elsevier, 2005, pp. 665–670 (cit. on p. 148).

[62] Itzhak Fogel and Dov Sagi. "Gabor filters as texture discriminator." In: *Biological cybernetics* 61.2 (1989), pp. 103–113 (cit. on p. 17).

[63] Paul Fogel et al. "Inferential, robust non-negative matrix factorization analysis of microarray data." In: *Bioinformatics* 23 (2007), pp. 44–49 (cit. on p. 36).

[64] Susan R. Fussell, Robert E. Kraut, and Jane Siegel. "Coordination of communication: Effects of shared visual context on collaborative work." In: *Proceedings of the 2000 ACM conference on Computer supported cooperative work*. ACM. 2000, pp. 21–30 (cit. on p. 11).

[65] E. Garcia. *SVD and LSI Tutorial 1: Understanding SVD and LSI*. 2007 (cit. on p. 30).

[66] John S. Garofolo et al. "Getting started with the DARPA TIMIT CD-ROM: An acoustic phonetic continuous speech database." In: *National Institute of Standards and Technology (NIST), Gaithersburgh, MD* 107 (1988) (cit. on p. 16).

[67] Eric Gaussier and Cyril Goutte. "Relation between PLSA and NMF and implications." In: *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM. 2005, pp. 601–602 (cit. on p. 41).

[68] Rainer Gemulla et al. "Large-scale matrix factorization with distributed stochastic gradient descent." In: *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '11*. 2011, pp. 69–77 (cit. on p. 35).

[69] Walter R. Gilks, Sylvia Richardson, and David J. Spiegelhalter. "Introducing markov chain monte carlo." In: *Markov chain Monte Carlo in practice* 1 (1996), p. 19 (cit. on p. 150).

[70] Lila Gleitman. "The Structural Sources of Verb Meanings." In: *Language Acquisition* 1.1 (1990), pp. 3–55 (cit. on p. 44).

[71] Kevin Gold and Brian Scassellati. "Grounded pronoun learning and pronoun reversal." In: *Proceedings of the 5th International Conference on Development and Learning*. 2006 (cit. on p. 10).

[72] Peter John Gorniak. "The affordance-based concept." PhD thesis. Massachusetts Institute of Technology, 2005 (cit. on p. 9).

[73] Thomas L. Griffiths and Mark Steyvers. "A probabilistic approach to semantic representation." In: *Proceedings of the 24th annual conference of the cognitive science society*. Citeseer. 2002, pp. 381–386 (cit. on p. 22).

[74] Thomas L. Griffiths and Mark Steyvers. "Finding scientific topics." In: *Proceedings of the National Academy of Sciences of the United States of America* 101 Suppl (2004), pp. 5228–5235 (cit. on pp. 22, 150–152).

[75] T. Griffiths, Mark Steyvers, et al. "Prediction and semantic association." In: *Advances in neural information processing systems* (2003), pp. 11–18 (cit. on p. 22).

[76] Daniel H. Grollman and Odest Chadwicke Jenkins. "Sparse incremental learning for interactive robot control policy estimation." In: *2008 IEEE International Conference on Robotics and Automation*. 2008, pp. 3315–3320 (cit. on p. 2).

[77] D. Guillamet, J. Vitrià, and B. Schiele. "Introducing a weighted non-negative matrix factorization for image classification." In: *Pattern Recognition Letters* 24 (2003), pp. 2447–2454 (cit. on p. 33).

[78] M. T. Hagan, H. B. Demuth, and M. H. Beale. *Neural network design*. Vol. 2. 1996, p. 734 (cit. on pp. 2, 32).

[79] Joy E. Hanna and Susan E. Brennan. "Speakers' eye gaze disambiguates referring expressions early during face-to-face conversation." In: *Journal of Memory and Language* 57.4 (2007), pp. 596–615 (cit. on p. 11).

[80] Joy E. Hanna and Michael K. Tanenhaus. "Pragmatic effects on reference resolution in a collaborative task: Evidence from eye movements." In: *Cognitive Science* 28.1 (2004), pp. 105–115 (cit. on p. 11).

[81] Stevan Harnad. "The symbol grounding problem." In: *Physica D: Nonlinear Phenomena* 42.1 (1990), pp. 335–346 (cit. on p. 9).

[82] M. Hasenjäger and H. Ritter. "Active learning in neural networks." In: *New learning paradigms in soft computing*. Springer, 2002, pp. 137–169 (cit. on p. 46).

[83] Hynek Hermansky and Nelson Morgan. "RASTA processing of speech." In: *Speech and Audio Processing, IEEE Transactions on* 2.4 (1994), pp. 578–589 (cit. on p. 15).

[84] G. E. Hinton and R. R. Salakhutdinov. "Reducing the dimensionality of data with neural networks." In: *Science (New York, N.Y.)* 313 (2006), pp. 504–507 (cit. on p. 27).

[85] Geoffrey E. Hinton. *Learning multiple layers of representation*. 2007 (cit. on p. 27).

[86] Masako Hirotani et al. "Joint attention helps infants learn new words: event-related potential evidence." In: *Neuroreport* 20 (6 2009), pp. 600–605 (cit. on p. 44).

[87] Matthew D. Hoffman, David M. Blei, and Francis Bach. "Online Learning for Latent Dirichlet Allocation." In: *Advances in Neural Information Processing Systems* 23 (2010), pp. 1–9 (cit. on p. 152).

[88] Thomas Hofmann. "Probabilistic latent semantic indexing." In: *SIGIR*. 1999, pp. 50–57 (cit. on pp. 22, 39).

[89] Thomas Hofmann. "Unsupervised learning by probabilistic Latent Semantic Analysis." In: *Machine Learning* 42 (2001), pp. 177–196 (cit. on pp. 22, 41).

[90] Eva Hörster, Rainer Lienhart, and Malcolm Slaney. "Image retrieval on large-scale image databases." In: *Image Rochester NY* (2007), pp. 17–24 (cit. on p. 41).

[91] Charles Lee Isbell et al. "A Social Reinforcement Learning Agent." In: *Fifth International Conference on Autonomous Agents* (2001), p. 8 (cit. on p. 2).

[92] Douglas Northrop Jackson, Richard D. Goffin, and Edward Helmes. *Problems and solutions in human assessment: Honoring Douglas N. Jackson at Seventy*. Springer, 2000 (cit. on p. 36).

[93] G. Kachergis and Chen Yu. "Continuous measure of word learning supports associative model." In: *Development and Learning and Epigenetic Robotics (ICDL-Epirob), 2014 Joint IEEE International Conferences on*. Oct. 2014, pp. 20–25 (cit. on p. 26).

[94] G. Kachergis, Chen Yu, and R.M. Shiffrin. "Cross-situational word learning is better modeled by associations than hypotheses." In: *Development and Learning and Epigenetic Robotics (ICDL), 2012 IEEE International Conference on*. Nov. 2012, pp. 1–6 (cit. on p. 44).

[95] George Kachergis and Chen Yu. "More Naturalistic Cross-situational Word Learning." In: *Proceedings of the 35th Annual Conference of the Cognitive Science Society*. 2013 (cit. on p. 44).

[96] George Kachergis, Chen Yu, and Richard M. Shiffrin. "Actively Learning Object Names Across Ambiguous Situations." In: *topiCS* 5.1 (2013), pp. 200–213 (cit. on pp. 44, 47, 137, 138, 141).

[97] H. F. Kaiser. *The Application of Electronic Computers to Factor Analysis*. 1960 (cit. on p. 36).

[98] Sham Kakade and Peter Dayan. "Dopamine: generalization and bonuses." In: *Neural Networks* 15.4 (2002), pp. 549–559 (cit. on p. 45).

[99] Dan Kalman. "A Singularly Valuable Decomposition: The SVD of a Matrix." In: *The College Mathematics Journal* 27 (1996), p. 2 (cit. on p. 29).

[100] Frédéric Kaplan, Pierre-Yves Oudeyer, and Benjamin Bergen. "Computational models in the debate over language learnability." In: *Infant and Child Development* 17.1 (2008), pp. 55–80 (cit. on p. 46).

[101] Yan Ke and Rahul Sukthankar. "PCA-SIFT: A more distinctive representation for local image descriptors." In: *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. Vol. 2. IEEE. 2004, pp. II–506 (cit. on p. 18).

[102] Kristian Kersting et al. "Hierarchical Convex NMF for Clustering Massive Data." In: *ACML* (2010), pp. 253–268 (cit. on pp. 33, 147).

[103] Hyunsoo Kim and Haesun Park. *Nonnegative Matrix Factorization Based on Alternating Nonnegativity Constrained Least Squares and Active Set Method*. 2008 (cit. on p. 35).

[104] Jingu Kim and Haesun Park. *Fast Nonnegative Matrix Factorization: An Active-Set-Like Method and Comparisons*. 2011 (cit. on pp. 33, 35).

[105] Diederik P. Kingma and Max Welling. "Auto-encoding variational bayes." In: *arXiv preprint arXiv:1312.6114* (2013) (cit. on p. 27).

[106] Patricia K. Kuhl. *Brain Mechanisms in Early Language Acquisition*. 2010 (cit. on p. 4).

[107] Patricia K. Kuhl. "Early language acquisition: cracking the speech code." In: *Nature reviews neuroscience* 5 (2004), pp. 831–843 (cit. on p. 4).

[108] Thomas K. Landauer. "LSA as a theory of meaning." In: *Handbook of latent semantic analysis*. 2007, pp. 3–34 (cit. on p. 30).

[109] Stephen David Larson. *Intrinsic representation: Bootstrapping symbols from experience*. Springer, 2004 (cit. on p. 9).

[110] Stanislao Lauria et al. "Mobile robot programming using natural language." In: *Robotics and Autonomous Systems* 38 (2002), pp. 171–181 (cit. on p. 2).

[111] C. L. Lawson and R. J. Hanson. *Solving least squares problems*. Vol. 15. 1995, p. 337 (cit. on p. 38).

[112] D. D. Lee and H. S. Seung. "Algorithms for non-negative matrix factorization." In: *Advances in neural information processing systems* (2001), pp. 556–562 (cit. on pp. 2, 32, 35).

[113] D. D. Lee and H. S. Seung. "Learning the parts of objects by non-negative matrix factorization." In: *Nature* 401 (1999), pp. 788–791 (cit. on pp. 2, 29, 32, 33).

[114] Yi-Ou Li, Tülay Adalı, and Vince D. Calhoun. "Estimating the number of independent components for functional magnetic resonance imaging data." In: *Human brain mapping* 28.11 (2007), pp. 1251–1266 (cit. on p. 36).

[115] Dawen Liang, Matthew D. Hoffman, and Daniel P. W. Ellis. "Beta Process Sparse Nonnegative Matrix Factorization for Music." In: *ISMIR*. 2013, pp. 375–380 (cit. on p. 36).

[116] Rainer Lienhart, Stefan Romberg, and Eva Hörster. "Multilayer pLSA for multimodal image retrieval." In: *International Conference on Image and Video Retrieval -CIVR*. 2009, p. 1 (cit. on p. 41).

[117] Chih-Jen. Lin. "On the convergence of multiplicative update algorithms for nonnegative matrix factorization." In: *IEEE Transactions on Neural Networks* 18 (2007), pp. 1589–1596 (cit. on p. 35).

[118] Chih-Jen Lin. "Projected gradient methods for nonnegative matrix factorization." In: *Neural computation* 19 (2007), pp. 2756–2779 (cit. on p. 35).

[119] David G. Lowe. "Distinctive image features from scale-invariant keypoints." In: *International journal of computer vision* 60.2 (2004), pp. 91–110 (cit. on p. 19).

[120] Natalia Lyubova and David Filliat. "Developmental approach for interactive object discovery." In: *Neural Networks (IJCNN), The 2012 International Joint Conference on*. IEEE. 2012, pp. 1–7 (cit. on p. 17).

[121] Richard Maclin et al. "Giving advice about preferred actions to reinforcement learners via knowledge-based kernel regression." In: *Proceedings of the 20th National Conference on Artificial Intelligence*. Vol. 2. 2005, pp. 819–824 (cit. on p. 2).

[122] José M. Maisog. "Non-negative Matrix Factorization: Assessing Methods for Evaluating the Number of Components, and the Effect of Normalization Thereon." PhD thesis. Georgetown University, 2009 (cit. on p. 36).

[123] Olivier Mangin. "The Emergence of Multimodal Concepts." PhD thesis. 2014 (cit. on pp. 12–15, 33, 61, 63, 93).

[124] Olivier Mangin and Pierre Yves Oudeyer. "Learning semantic components from subsymbolic multimodal perception." In: *2013 IEEE 3rd Joint International Conference on Development and Learning and Epigenetic Robotics, ICDL 2013 - Electronic Conference Proceedings* (2013) (cit. on p. 63).

[125] Olivier Mangin et al. "MCA-NMF: Multimodal Concept Acquisition with Non-Negative Matrix Factorization." In: *PLoS ONE* 10.10 (Oct. 2015), e0140732 (cit. on pp. 12, 17, 21, 24, 63).

[126] Nikolaos Mavridis and Deb K. Roy. "Grounded situation models for robots: Where words and percepts meet." In: *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. IEEE. 2006, pp. 4690–4697 (cit. on p. 10).

[127] Nikolaos Mavridis et al. "FaceBots: social robots utilizing facebook." In: *Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*. ACM. 2009, pp. 195–196 (cit. on p. 10).

[128] Tamara Nicol Medina et al. "How words can and cannot be learned by observation." In: *Proceedings of the National Academy of Sciences* 108.22 (2011), pp. 9014–9019. eprint: `http://www.pnas.org/content/108/22/9014.full.pdf+html` (cit. on p. 26).

[129] George Miller and Christiane Fellbaum. *Wordnet: An electronic lexical database*. 1998 (cit. on p. 16).

[130] T. P. Minka. "Automatic choice of dimensionality for PCA." In: *Advances in neural information processing systems* (2001), pp. 598–604 (cit. on p. 36).

[131] Thomas Minka and John Lafferty. "Expectation-Propagation for the Generative Aspect Model." In: *Uncertainty in Artificial Intelligence*. 2002, pp. 352–359 (cit. on pp. 150, 151).

[132] Tom Mitchell. *Never-ending learning*. Tech. rep. DTIC Document, 2010 (cit. on pp. 3, 4).

[133] Florent Monay and Daniel Gatica-perez. "PLSA-based Image Auto-Annotation : Constraining the Latent Space." In: *Proceedings of the 12th annual ACM international conference on Multimedia* (2004), pp. 348–351 (cit. on p. 41).

[134] AJung Moon et al. "Meet me where i'm gazing: how shared attention gaze affects human-robot handover timing." In: *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*. ACM. 2014, pp. 334–341 (cit. on p. 11).

[135] Peter Mundy and Lisa Newell. "Attention, joint attention, and social cognition." In: *Current directions in psychological science* 16.5 (2007), pp. 269–274 (cit. on p. 143).

[136] Kuniaki Noda et al. "Multimodal integration learning of robot behavior using deep neural networks." In: *Robotics and Autonomous Systems* 62.6 (June 2014), pp. 721–736 (cit. on pp. 11–14, 18, 21, 27, 28).

[137] Dan Oneata. *Probabilistic Latent Semantic Analysis*. Tech. rep. The University of Edinburgh School of Informatics (cit. on p. 39).

[138] Pierre Yves Oudeyer, Frédéric Kaplan, and Verena V. Hafner. "Intrinsic Motivation Systems for Autonomous Mental Development." In: *IEEE Transactions on Evolutionary Computation* 11.2 (Apr. 2007), pp. 265–286 (cit. on pp. 4, 46, 47, 65).

[139] Pierre-Yves Oudeyer and Frédéric Kaplan. "What is intrinsic motivation? a typology of computational approaches." In: *Frontiers in neurorobotics* 1 (2007), p. 6 (cit. on p. 45).

[140] Art B. Owen and Patrick O. Perry. "Bi-cross-validation of the SVD and the nonnegative matrix factorization." In: *The Annals of Applied Statistics* (2009), pp. 564–594 (cit. on p. 36).

[141] Fabio Pardo, Yuxin Chen, and David Filliat. "Effects of robot feedback on teacher during word-referent learning." In: *ICDL-EpiRob 2015 Workshop on Mechanisms of Learning in Social Contexts*. IEEE. 2015 (cit. on pp. 7, 179).

[142] Karl Pearson. "LIII. On lines and planes of closest fit to systems of points in space." In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2 (1901), pp. 559–572 (cit. on p. 31).

[143] J. Piaget. *Play, Dreams and Imitation in Childhood*. Developmental psychology. Routledge, 1999 (cit. on p. 4).

[144] Steven Pinker. *Language Learnability and Language Development*. Vol. 193. 1984, p. 435 (cit. on p. 3).

[145] Willard Van Orman Quine, Otto Neurath, and James Grier Miller. "Word and Object." In: *Language* (1960), pp. 1–201 (cit. on pp. 3, 44).

[146] Terry Regier and Laura A. Carlson. "Grounding spatial language in perception: an empirical and computational investigation." In: *Journal of experimental psychology: General* 130.2 (2001), p. 273 (cit. on p. 10).

[147] Edwin M. Robertson, Daniel Z. Press, and Alvaro Pascual-Leone. "Off-line learning and the primary motor cortex." In: *The Journal of Neuroscience* 25.27 (2005), pp. 6372–6378 (cit. on p. 45).

[148] Deb K. Roy. "A computational model of word learning from multimodal sensory input." In: *Proceedings of the International Conference of Cognitive Modeling (ICCM2000), Groningen, Netherlands*. Citeseer. 2000 (cit. on p. 10).

[149] Deb K. Roy. "Learning visually grounded words and syntax for a scene description task." In: *Computer Speech & Language* 16.3 (2002), pp. 353–385 (cit. on p. 10).

[150] Deb K. Roy and A. Pentland. "Learning words from sights and sounds: A computational model." In: *Cognitive science* 26 (2002), pp. 113–146 (cit. on pp. 12, 14, 15, 18, 24, 25, 44).

[151] J. Ruiz-del-Solar and P. Navarrete. "Eigenspace-based face recognition: a comparative study of different approaches." In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 35 (2005) (cit. on p. 31).

[152] Radu Bogdan Rusu et al. "Detecting and segmenting objects for mobile manipulation." In: *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*. IEEE. 2009, pp. 47–54 (cit. on p. 18).

[153] Radu Bogdan Rusu et al. "Fast 3d recognition and pose using the viewpoint feature histogram." In: *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE. 2010, pp. 2155–2162 (cit. on p. 18).

[154] Paul Sajda et al. "Nonnegative matrix factorization for rapid recovery of constituent spectra in magnetic resonance chemical shift imaging of the brain." In: *IEEE Transactions on Medical Imaging* 23 (2004), pp. 1453–1465 (cit. on p. 33).

[155] Gerard Salton and Christopher Buckley. "Term-weighting Approaches in Automatic Text Retrieval." In: *Inf. Process. Manage.* 24.5 (Aug. 1988), pp. 513–523 (cit. on p. 60).

[156] Brian Scassellati. "Imitation and mechanisms of joint attention: A developmental structure for building social skills on a humanoid robot." In: *Computation for metaphors, analogy, and agents*. Springer, 1999, pp. 176–195 (cit. on p. 11).

[157] Brian Scassellati. "Mechanisms of shared attention for a humanoid robot." In: *Embodied Cognition and Action: Papers from the 1996 AAAI Fall Symposium*. Vol. 4. 9. 1996, p. 21 (cit. on p. 11).

[158] Stefan Schaal. *Is imitation learning the route to humanoid robots?* 1999 (cit. on p. 2).

[159] Bernt Schiele and James L. Crowley. "Recognition without correspondence using multidimensional receptive field histograms." In: *International Journal of Computer Vision* 36.1 (2000), pp. 31–50 (cit. on p. 17).

[160] Jürgen Schmidhuber. "Curious model-building control systems." In: *Neural Networks, 1991. 1991 IEEE International Joint Conference on*. IEEE. 1991, pp. 1458–1463 (cit. on p. 47).

[161] B. Schölkopf. "Learning with kernels." In: *Journal of the Electrochemical Society* 129 (2002), p. 2865 (cit. on p. 38).

[162] B. Schölkopf, A. J. Smola, and K. R. Muller. "Kernel Principal Component Analysis." In: *Computer Vision And Mathematical Methods In Medical And Biomedical Image Analysis* 1327 (2012), pp. 583–588 (cit. on p. 32).

[163] Bernhard Schölkopf et al. "Support Vector Method for Novelty Detection." In: *Advances in Neural Information Processing Systems 12*. 1999, pp. 582–588 (cit. on p. 37).

[164] William Schueller and Pierre-Yves Oudeyer. "Active learning strategies and active control of complexity growth in naming games." In: *the 5th International Conference on Development and Learning and on Epigenetic Robotics*. 2015 (cit. on pp. 12, 47, 66, 67, 148).

[165] Karin Kipper Schuler. "VerbNet: A broad-coverage, comprehensive verb lexicon." PhD thesis. University of Pennsylvania, Philadelphie, 2005 (cit. on p. 148).

[166] Claude Elwood Shannon. "A mathematical theory of communication." In: *ACM SIGMOBILE Mobile Computing and Communications Review* 5.1 (2001), pp. 3–55 (cit. on p. 12).

[167] Xueguang Shao et al. "Extraction of mass spectra and chromatographic profiles from overlapping GC/MS signal with background." In: *Analytical Chemistry* 76 (2004), pp. 5143–5148 (cit. on p. 36).

[168] Jeffrey Mark Siskind. "A computational study of cross-situational techniques for learning word-to-meaning mappings." In: *Cognition* 61.1–2 (1996). Compositional Language Acquisition, pp. 39–91 (cit. on p. 44).

[169] Jeffrey Mark Siskind. "Grounding the lexical semantics of verbs in visual perception using force dynamics and event logic." In: *J. Artif. Intell. Res.(JAIR)* 15 (2001), pp. 31–90 (cit. on p. 20).

[170] Marjorie Skubic et al. "Spatial language for human-robot dialogs." In: *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 34.2 (2004), pp. 154–167 (cit. on p. 10).

[171] Daniel D.K. Sleator and Davy Temperley. "Parsing English with a link grammar." In: *Technical Report CMU-CS-91- 196* (1991) (cit. on p. 16).

[172] W.D. Smart and L. Pack Kaelbling. "Effective reinforcement learning for mobile robots." In: *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)* 4 (2002) (cit. on p. 2).

[173] Linda Smith and Michael Gasser. "The development of embodied cognition: Six lessons from babies." In: *Artificial life* 11.1-2 (2005), pp. 13–29 (cit. on p. 4).

[174] Linda Smith and Chen Yu. "Infants rapidly learn word-referent mappings via cross-situational statistics." In: *Cognition* 106.3 (2008), pp. 1558–1568 (cit. on pp. 12, 44, 45).

[175] Hyun Ah Song and Soo-Young Lee. "Hierarchical Data Representation Model-Multi-layer NMF." In: *arXiv preprint arXiv:1301.6316* (2013) (cit. on p. 33).

[176] Xiaodan Song et al. "Modeling and predicting personal information dissemination behavior." In: *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. ACM. 2005, pp. 479–488 (cit. on p. 152).

[177] Thorsten Spexard et al. "BIRON, where are you? Enabling a robot to learn new places in a real home environment by integrating spoken dialog and visual localization." In: *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. IEEE. 2006, pp. 934–940 (cit. on p. 10).

[178] Luc Steels. "Evolving grounded communication for robots." In: *Trends in Cognitive Sciences* 7.7 (2003), pp. 308–312 (cit. on pp. 9, 12–14, 16, 18, 19).

[179] Luc Steels. *The Talking Heads experiment: Origins of words and meanings*. Vol. 1. Language Science Press, 2015 (cit. on pp. 9, 18).

[180] G. W. Stewart. *On the Early History of the Singular Value Decomposition*. 1993 (cit. on p. 29).

[181] Michael J. Swain and Dana H. Ballard. "Color indexing." In: *International journal of computer vision* 7.1 (1991), pp. 11–32 (cit. on p. 17).

[182] Leonard Talmy. "Force dynamics in language and cognition." In: *Cognitive science* 12.1 (1988), pp. 49–100 (cit. on p. 20).

[183] Yee Whye Teh, David Newman, and Max Welling. "A Collapsed Variational Bayesian Inference Algorithm for Latent Dirichlet Allocation." In: *NIPS* 19 (2007), pp. 1353–1360 (cit. on pp. 150, 151).

[184] Stefanie Tellex et al. "Approaching the symbol grounding problem with probabilistic graphical models." In: *AI magazine* 32.4 (2011), pp. 64–76 (cit. on p. 10).

[185] Stefanie Tellex et al. "Learning perceptually grounded word meanings from unaligned parallel data." In: *Machine Learning* 94.2 (2014), pp. 151–167 (cit. on p. 10).

[186] Stefanie Tellex et al. "Understanding natural language commands for robotic navigation and mobile manipulation." In: (2011) (cit. on p. 10).

[187] Andrea L. Thomaz and Cynthia Breazeal. "Teachable robots: Understanding human teaching behavior to build more effective robot learners." In: *Artificial Intelligence* 172 (2008), pp. 716–737 (cit. on p. 2).

[188] Andrea L. Thomaz and Maya Cakmak. "Social learning mechanisms for robots." In: (2009) (cit. on p. 2).

[189] Michael Tomasello. *The Cultural Origins of Human Cognition*. Vol. 114. 1999, p. 248 (cit. on p. 2).

[190] John C. Trueswell et al. "Propose but verify: Fast mapping meets cross-situational word learning." In: *Cognitive psychology* 66.1 (2013), pp. 126–156 (cit. on p. 26).

[191] Alan M. Turing. "Computing machinery and intelligence." In: *Mind* 59.236 (1950), pp. 433–460 (cit. on p. 4).

[192] M.A. Turk and A.P. Pentland. "Face recognition using eigenfaces." In: *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (1991) (cit. on pp. 29, 31).

[193] Mark H. Van Benthem and Michael R. Keenan. "Fast algorithm for the solution of large-scale non-negativity-constrained least squares problems." In: *Journal of Chemometrics* 18 (2004), pp. 441–450 (cit. on p. 35).

[194] Hugo Van Hamme. "H.: Hac-models: a novel approach to continuous speech recognition." In: *Proceedings Interspeech, ISCA*. Citeseer. 2008 (cit. on p. 15).

[195] Wayne F. Velicer. "Determining the number of components from the matrix of partial correlations." In: *Psychometrika* 41 (1976), pp. 321–327 (cit. on p. 36).

[196] Paul Vogt. "The physical symbol grounding problem." In: *Cognitive Systems Research* 3.3 (2002), pp. 429–457 (cit. on p. 12).

[197] Alex Waibel et al. "Phoneme recognition using time-delay neural networks." In: *IEEE transactions on acoustics, speech, and signal processing* 37.3 (1989), pp. 328–339 (cit. on p. 27).

[198] J. Weng et al. "Artificial intelligence. Autonomous mental development by robots and animals." In: *Science (New York, N.Y.)* 291 (2001), pp. 599–600 (cit. on p. 4).

[199] Zhirong Yang, Zhanxing Zhu, and Erkki Oja. "Automatic rank determination in projective nonnegative matrix factorization." In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 6365 LNCS. 2010, pp. 514–521 (cit. on p. 36).

[200] Chen Yu and Dana H. Ballard. "A Multimodal Learning Interface for Grounding Spoken Language in Sensory Perceptions." In: *ACM Trans. Appl. Percept.* 1.1 (July 2004), pp. 57–80 (cit. on p. 20).

[201] Chen Yu and Dana H. Ballard. "On the integration of grounding language and learning objects." In: *AAAI*. Vol. 4. 2004, pp. 488–493 (cit. on pp. 12–14, 16, 17).

[202] Chen Yu, Linda B. Smith, and Alfredo F. Pereira. "Grounding word learning in multimodal sensorimotor interaction." In: *Proceedings of the 30th annual conference of the cognitive science society*. 2008, pp. 1017–1022 (cit. on p. 9).

[203] Chen Yu et al. "Rapid word learning under uncertainty via cross-situational statistics." In: *Psychological Science* (2007), pp. 414–420 (cit. on p. 44).

[204] Hendrik Zender et al. "Conceptual spatial representations for indoor mobile robots." In: *Robotics and Autonomous Systems* 56.6 (2008), pp. 493–502 (cit. on p. 10).

[205] Xiaoge Zhang et al. "An improved Physarum polycephalum algorithm for the shortest path problem." In: *The Scientific World Journal* 2014 (2014) (cit. on p. 2).

université PARIS-SACLAY
ÉCOLE DOCTORALE
INTERFACES
Approches Interdisciplinaires : Fondements, Applications et Innovation

Titre : Apprentissage interactif de mots et d'objets pour un robot humanoïde

Mots clefs : robotique développementale, apprentissage de mot-référent, apprentissage cross-situationnel, apprentissage actif, Factorisation en Natrices Non-Négatives (NMF), Allocation de Dirichlet latente (LDA)

Résumé : Les applications futures de la robotique exigeront des capacités d'adaptation à l'environnement, notamment la capacité à reconnaître des nouveaux objets via l'interaction avec les humains. Or les enfants de deux ans ont une capacité impressionnante à apprendre à reconnaître de nouveaux objets par l'interaction avec les adultes et sans supervision précise. Par conséquent, suivant l'approche de la robotique développementale, nous développons dans la thèse des approches d'apprentissage pour les objets, en associant leurs noms et leurs caractéristiques correspondantes, inspirées par les capacités des enfants qui sont exploitées lors de l'interaction ambiguë avec les parents.
Dans le cadre de l'apprentissage cross-situationnel, deux approches de découverte de thèmes latents (la Factorisation en Natrices Non-Négatives (NMF) et l'Allocation de Dirichlet latente (LDA)) sont utilisées pour la découverte de concepts multi-modaux, découvrant les régularités sous-jacentes dans le flux de données brutes afin de parvenir à produire des ensembles de mots et leur signification visuelle associée, p.ex le nom d'un objet et sa forme, ou un adjectif de couleur et sa correspondance dans les images, malgré les *ambiguïtés référentielles* et les *ambiguïtés linguistiques*. Par ailleurs, deux stratégies d'apprentissage actives: la Sélection par l'Erreur de Reconstruction Maximale (MRES) et l'Exploration Basée sur la Confiance (CBE) sont proposées pour améliorer l'apprentissage incrémental en termes de qualité et de vitesse. Enfin, les comportements d'apprentissage sont comparés entre les humains et les modèles proposés qui ont été étudiés non seulement à travers de simulations extensives mais aussi lors d'interactions réelles entre des humains et un robot.

Title : Interactive learning of words and objects for a humanoid robot

Keywords : developmental robotics, word-referent learning, cross-situational learning, active learning, Non negative Matrix Factorization (NMF), Latent Dirichlet Association (LDA)

Abstract : Future applications of robotics will require adaptability to the environment, particularly the ability to recognize new objects through interaction with humans. In fact, two year old children have an impressive ability to learn to recognize new objects via interaction with adults and without precise supervision. Therefore, following the developmental robotics approach, we develop in the thesis learning approaches for objects, associating their names and corresponding features, inspired by the capabilities of infants that are used during ambiguous interaction with parents.
Under the framework of cross-situational learning, two latent topic discovery approaches, that is Non Negative Matrix Factorization (NMF) and Latent Dirichlet Association (LDA), are utilized to implement multi-modal concept discovery, finding the underlying regularities in the raw dataflow to produce sets of words and their associated visual meanings, eg. the name of an object and its shape, or a color adjective and its correspondence in images, despite *referential ambiguities* and *linguistic ambiguities*. Besides, two active learning strategies: Maximum Reconstruction Error Based Selection (MRES) and Confidence Based Exploration (CBE), are proposed to improve incremental learning in terms of quality and speed. Finally, learning behaviors are compared between humans and the proposed models not only through extensive simulations but also through applications in real human-robot interactions.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France