Journal of Artificial Intelligence, Machine Learning and Soft Computing
Volume 2, Issue 2, July-December 2017

Zero-Shot and Few-Shot Learning: Enabling Intelligence with Minimal Data

Riya Kulkarni[1], Manav Deshpande[2], Ishita Rao[3], Kunal Bendre[4]
Associate Professor, Students
Department of Computer Science,
Shantideep Institute of Technology, India
Email ID:*Riyakulkarni120@gmail.com[1], manaydeshpande71@rediffmail.com[2],*
ishitaroaobjf@yahoo.com[3]

Abstract

Traditional machine learning models depend heavily on large labeled datasets for achieving good performance. However, in many real-world situations, collecting large amounts of labeled data is expensive, time consuming, or even impossible. This limitation motivated the development of Zero-Shot Learning (ZSL) and Few-Shot Learning (FSL) techniques, which aim to recognize new classes with little or no training examples. These approaches try to mimic the human ability of learning new concepts from very limited experience. Zero-shot learning relies on auxiliary information such as semantic attributes or textual descriptions to identify unseen classes, while few-shot learning leverages a small number of examples with advanced training strategies like meta-learning and metric learning. This paper presents a detailed review of the principles, techniques, architectures, and applications of ZSL and FSL. Recent advancements using deep learning, transformers, and large language models are also discussed. Challenges, evaluation methods, and future research directions are highlighted.

Keywords: *Zero-Shot Learning, Few-Shot Learning, Meta-Learning, Transfer Learning, Metric Learning, Deep Learning, Semantic Embedding*

INTRODUCTION

Machine learning systems generally require thousands or millions of labeled samples to learn meaningful patterns. This requirement limits the use of AI in domains where data is rare, sensitive, or costly. Humans, on the other hand, can recognize new objects after seeing them only once or even by reading a description. For example, a person who has never seen a zebra can identify it from the description “a horse with black and white stripes”.

Zero-Shot Learning (ZSL) and Few-Shot Learning (FSL) are inspired from this human learning behavior. These learning paradigms attempt to reduce dependency on large datasets by transferring knowledge from known classes to unknown classes.

ZSL focuses on identifying classes that are not present during training. FSL focuses on learning from a very small number of labeled examples, typically 1-5 samples per class. Both techniques are becoming very important in computer vision, NLP, robotics, healthcare, and many other domains.

2. BACKGROUND CONCEPTS (ELABORATED)

A clear understanding of Zero-Shot Learning (ZSL) and Few-Shot Learning (FSL) requires some important foundational ideas from modern machine learning. Among them, **transfer learning**, **representation learning**, and the concept of a **shared semantic embedding space** play a central role. These ideas enable models to generalize beyond the data they have directly seen during training and to make meaningful predictions for new or rare classes.

2.1 Transfer Learning

Transfer learning refers to the ability of a model to **reuse knowledge learned from one task (source task)** to improve performance on a **different but related task (target task)**. Instead of training a model from scratch with large data, knowledge from a previously trained model is transferred.

In traditional learning, models start with random parameters and learn only from the given dataset. But in transfer learning, models start with **pretrained knowledge**, usually obtained from very large datasets such as ImageNet (for images) or large text corpora (for language).

Why Transfer Learning is Important for ZSL and FSL

- In **ZSL**, the model has never seen examples of certain classes. It must rely on previously learned visual and semantic knowledge to recognize them.

- In **FSL**, only a few examples are available, so prior knowledge helps the model adapt quickly.

For example, a CNN trained on ImageNet already knows how to detect edges, shapes, textures, and object parts. This visual knowledge can be transferred to identify a new animal class with very few images.

Types of Transfer Learning

Type	Description	Use in ZSL/FSL
Feature Transfer	Using pretrained layers as feature extractors	Common in both ZSL and FSL
Fine-tuning	Slightly updating pretrained weights	Mostly in FSL
Domain Adaptation	Adapting knowledge across domains	Helpful when unseen classes are from different domain
Cross-modal Transfer	Transfer between text and images	Crucial for ZSL

Transfer learning reduces the data requirement, training time, and improves generalization, which is why it is a backbone for both ZSL and FSL approaches.

2.2 Representation Learning

Representation learning is the process where neural networks automatically learn **useful features** from raw data. Instead of manually designing features, deep networks learn hierarchical representations.

For example:

- Early CNN layers learn edges and colors.
- Middle layers learn shapes and textures.
- Final layers learn object-level concepts.

This hierarchy of learned features is extremely valuable when data is limited.

Importance in Few-Shot Learning

In FSL, because only 1–5 examples per class are available, the quality of feature representation becomes more important than the quantity of data. If features are discriminative enough, even a simple classifier can separate classes.

Metric learning methods like Prototypical Networks rely entirely on good embeddings to compute distances between samples.

Importance in Zero-Shot Learning

In ZSL, representation learning helps in mapping images into a space where they can be compared with semantic descriptions. If the representation is weak, unseen classes cannot be identified correctly.

Self-Supervised and Contrastive Representation Learning

Recent techniques such as:

- Contrastive Learning (SimCLR, MoCo)
- Self-supervised Learning (BYOL, DINO)

learn strong representations without labels. These representations are highly useful for ZSL and FSL because they generalize well to unseen data.

2.3 Semantic Embedding Space

A semantic embedding space is a **shared vector space** where data from different modalities (such as images and text) are projected so that semantically similar items lie close together.
This concept is central to Zero-Shot Learning.

How It Works

1. Images are passed through a feature extractor to get visual embeddings.
2. Class descriptions (attributes, text, word vectors) are converted into semantic embeddings.
3. Both embeddings are mapped into the same space.
4. Classification is done by measuring similarity (e.g., cosine distance).

For instance, if the model has learned the semantic meaning of words like *stripes*, *four legs*, and *wild*, then an unseen image with these features can be matched to the class “zebra” even if no zebra image was used in training.

Sources of Semantic Information

Source	Example
Human-defined attributes	has wings, can fly, has fur
Word embeddings	Word2Vec, GloVe, FastText
Sentence embeddings	BERT, Sentence Transformers
Text-image embeddings	CLIP

Role in ZSL

Semantic embedding space allows **knowledge transfer from seen classes to unseen classes** using relationships defined by language or attributes.

Cross-Modal Embedding Example (CLIP)

CLIP learns joint embeddings of images and text. It can recognize unseen objects by comparing image embeddings with textual prompts such as *“a photo of a tiger”*.

3. ZERO-SHOT LEARNING (ZSL) – ELABORATED

Zero-Shot Learning (ZSL) is one of the most fascinating paradigms in modern machine learning because it tries to solve a very difficult problem: **recognizing classes that the model has never seen during training**. This idea is inspired from human intelligence, where a person can identify a new object simply from a description without ever seeing it before.

3.1 Definition

Zero-Shot Learning is the task of classifying or recognizing instances from **unseen classes** by transferring knowledge learned from **seen classes** through semantic relationships.

During training:

- The model is trained only on **seen classes**.
- **Unseen classes** are completely absent from training images/data.

- However, semantic information about unseen classes is available in the form of attributes, word vectors, or text descriptions.

During testing:

- The model must correctly identify instances belonging to unseen classes.

This is different from traditional classification where train and test classes are the same.

3.2 Working Principle of ZSL

The core idea of ZSL is to build a **bridge between visual information and semantic information**.

Instead of learning “image → class label” directly, the model learns:

image features → semantic description → class label

Auxiliary Semantic Information Used

ZSL relies on additional knowledge sources:

- **Attributes**: Human-defined properties (has fur, has wings, can swim)
- **Word Embeddings**: Vector representations of words (Word2Vec, GloVe, FastText)
- **Text Descriptions**: Sentences or paragraphs describing classes
- **Knowledge Graphs**: Relations between classes

Learning Process

1. Extract visual features from images using CNN/ViT.
2. Convert class descriptions into semantic vectors.
3. Learn a mapping function between visual features and semantic space.
4. For a test image, map it into semantic space and find the closest class description.

For example, if the model knows:

- Horse: four legs, tail
- Tiger: four legs, stripes

And receives a description:

- Zebra: four legs, stripes

The model can infer zebra without seeing any zebra image.

3.3 ZSL Framework

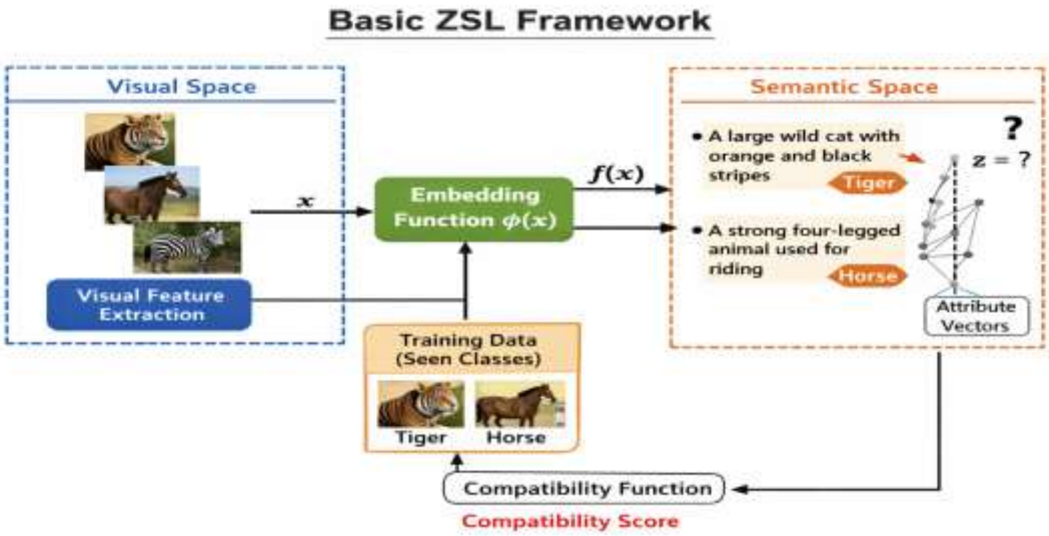

Figure 1: Basic ZSL Framework

Component	Role
Feature Extractor	Extract visual features from images
Semantic Encoder	Convert class description to vectors
Mapping Function	Learn relation between visual and semantic space
Similarity Measure	Cosine/Euclidean distance to predict class

3.4 Types of Zero-Shot Learning

Zero-shot learning methods can be categorized based on how semantic knowledge is used.

1. Attribute-Based ZSL

This is the earliest and most intuitive approach.

- Uses manually defined attributes for each class.
- Attributes are interpretable and human understandable.

Example:

Class	Has Fur	Has Stripes	Has Tail	Can Run
Tiger	Yes	Yes	Yes	Yes
Horse	Yes	No	Yes	Yes
Zebra	Yes	Yes	Yes	Yes

The model predicts attributes from image and matches with class attribute vector.

Advantages

- Interpretable
- Good for small datasets

Limitations

- Requires manual effort
- Hard to scale for many classes

2. Embedding-Based ZSL

Instead of manual attributes, semantic meaning is captured using word or sentence embeddings.

- Word2Vec, GloVe, FastText
- BERT or Sentence Transformers
- CLIP text encoder

Here, class names like “zebra”, “tiger”, “horse” are converted into vectors automatically.

Advantages

- No manual attribute design
- Scalable to thousands of classes

Limitations

- Word embeddings may not capture visual differences properly

3. Generative ZSL

A more advanced approach that uses **GANs or VAEs** to generate artificial visual features for unseen classes.

Process:

1. Use semantic vector of unseen class.
2. Generate synthetic image features.
3. Train a normal classifier using these synthetic features.

This converts ZSL into a traditional supervised problem.

Advantages

- Improves performance significantly

- Reduces bias toward seen classes

Limitations

- Training GAN/VAE is complex
- Quality of generated features matters

Additional Concept: Generalized ZSL (GZSL)

In real scenarios, test data may contain both seen and unseen classes. This is called **Generalized ZSL**.

The challenge here is that models tend to be biased toward seen classes. Special calibration techniques are required to balance predictions.

4. FEW-SHOT LEARNING (FSL)

4.1 Definition

Few-Shot Learning deals with learning new classes from a few examples (1-5).

4.2 N-way K-shot Problem

An N-way K-shot task means N classes with K examples each.

4.3 FSL Approaches

Approach	Description
Metric Learning	Learns similarity function
Meta-Learning	Learns how to learn
Optimization-based	Fine-tunes quickly with few samples
Data Augmentation	Synthetic sample generation

5. METRIC LEARNING IN FSL

Metric learning methods compare distances between samples.

5.1 Siamese Networks

Two identical networks compute similarity.

5.2 Prototypical Networks

Compute prototype (mean vector) for each class.

5.3 Matching Networks

Uses attention mechanism over support set.

6. META-LEARNING TECHNIQUES

Meta-learning trains models on many small tasks.

6.1 Model-Agnostic Meta-Learning (MAML)

Learns parameters that adapt quickly.

6.2 Reptile Algorithm

Simpler alternative to MAML.

6.3 Memory-Augmented Networks

Uses external memory to store knowledge.

7. GENERATIVE MODELS FOR ZSL AND FSL

GANs and VAEs are used to generate artificial features for unseen classes.

- Feature synthesis reduces data scarcity
- Improves classification accuracy

8. ROLE OF TRANSFORMERS AND LARGE MODELS

Recent large vision and language models help ZSL and FSL:

- CLIP aligns image and text
- GPT models provide semantic knowledge
- Vision Transformers (ViT) give strong representations

APPLICATIONS

Domain	Application
Healthcare	Rare disease detection
Robotics	Learning new objects
NLP	Intent detection with few samples
Security	Face recognition with few images
Wildlife	Identifying rare species

CHALLENGES

- Domain shift between seen and unseen classes
- Bias toward seen classes

- Poor semantic descriptions
- Evaluation complexity
- Overfitting in FSL

EVALUATION PROTOCOLS

Dataset	Use
AwA	ZSL animal dataset
CUB	Bird species
miniImageNet	FSL benchmark
Omniglot	Character recognition

Metrics include accuracy, harmonic mean, and top-k accuracy.

COMPARISON BETWEEN ZSL AND FSL

Aspect	ZSL	FSL
Training data	No samples of target class	Few samples
Auxiliary info	Required	Optional
Learning strategy	Semantic mapping	Rapid adaptation

RECENT ADVANCES

- Self-supervised pretraining
- Contrastive learning
- Cross-modal learning
- Prompt learning for ZSL

FUTURE DIRECTIONS

- Better semantic representation
- Unified ZSL-FSL frameworks
- Low-resource domain applications
- Continual learning integration

CONCLUSION

Zero-Shot and Few-Shot Learning are transforming how machine learning systems are trained in data-scarce environments. By leveraging semantic knowledge, meta-learning, and generative models, these approaches reduce dependency on large labeled datasets. With the rise of transformers and foundation models, ZSL and FSL are becoming more practical and powerful. Despite challenges like domain shift and evaluation bias, continuous research is improving robustness and applicability across many fields.

REFERENCES

1. Lampert, C. H., et al. “Attribute-Based Classification for Zero-Shot Learning.”
2. Socher, R., et al. “Zero-Shot Learning Through Cross-Modal Transfer.”
3. Xian, Y., et al. “Zero-Shot Learning: A Comprehensive Evaluation.”
4. Vinyals, O., et al. “Matching Networks for One Shot Learning.”
5. Snell, J., et al. “Prototypical Networks for Few-Shot Learning.”
6. Finn, C., et al. “Model-Agnostic Meta-Learning.”
7. Santoro, A., et al. “Memory-Augmented Neural Networks.”
8. Radford, A., et al. “Learning Transferable Visual Models From Natural Language Supervision.”
9. Wang, Y., et al. “Generalizing from Few Examples: A Survey on Few-Shot Learning.”
10. Liu, Z., et al. “Generative Models for Zero-Shot Learning.”