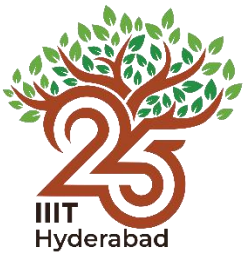
International Institute of Information Technology, Hyderabad

1st Year 2nd Semester



Introduction to NLP

Final Report

Team Name: Semantic Insighters

Project Mentor: Patanjali Bhamidipati

Project: Hypernym Discovery

S. No.	Name	Roll No.
1.	Sagnick Bhar	2023201008
2.	Soham Ghosh	2023202011
3.	Aman Khurana	2023201017

Problem Statement

The task of Hypernym Discovery in the paper is defined as finding and extracting suitable hypernyms for a target input term from a large textual corpus. For each input term the expected output is a ranked list of candidate hypernyms (up to 15) drawn from the provided vocabulary.

We are provided with a target term, a source corpus, and a vocabulary, and are required to retrieve a ranked list of candidate hypernyms for the input term. The task is split into two subtasks, general-purpose hypernym discovery and domain-specific hypernym discovery. Each subtask features specific training testing and trial data containing input terms and corresponding gold hypernym lists to evaluate the performance of participating systems.

About Dataset

- In this subtask, we consider 3 different languages:
  - English (subtask 1A), with a gold standard of 3,000 labelled terms
  - Italian (subtask 1B) and Spanish (subtask 1C), each with a gold standard of 2,000 labelled terms
- In this subtask we focus on English and consider two different domains of knowledge:
  - Medical (subtask 2A), with a gold standard of 1,000 labelled terms
  - Music (subtask 2B), also with a gold standard of 1,000 labelled terms

All the corpus are divided into train, validation and test splits along with additional vocabulary (upto tri-grams) which contains the subset of primary hypernyms that occurred less than 5-3 times and are over generic in nature. These filtered vocabularies can be used to reduce search space for potential candidates. The hyponyms are divided into two categories- Entity (names, location, etc.) and Concepts (phenomenon, activity, etc.).

	1A	1B	1C	2A	2B
Trial	50	25	25	15	15
Training	1,500	1,000	1,000	500	500
Test	1,500	1,000	1,000	500	500

From the above table, it is clear that the dataset was divided equally into training and testing sets for each subtask. This division ensures that the hypernym discovery systems are trained on a portion of the data and tested on another portion to evaluate their performance accurately. The trial data, which contains fewer examples compared to the training and testing sets, can also be utilized as a development set. Development sets are often used for fine-tuning models and evaluating performance before final testing.

Sample data of each dataset:

Hyponym	Hypernym	Corpus
pollution	Atmosphere, windstorm, violent storm, air current, atmospheric state, density current of air, storm damage, atmospheric phenomenon, storm, cyclone, natural phenomenon, tempest wind	English
bagpipe	Instrument, musical instrument, pipe, wind, wind instrument, aerophone	Music
bone spur	clinical finding, disease	Medical
lengüeta	Aleteo, instrumento de viento-madera, instrumento musical	Spanish
sesto	Grado, numero ordinale frazione, carica	Italian

Our Approach

1. **Baseline Method:**
   In our base-line strategy, we decided to divide our task into two key subtasks:
   1. "Embedding learning" (using Word2vec).
   2. Hypernym-hyponym relationship learning.

   In the supervised approach, we use neural network models (e.g., GRU and LSTM) to extract the latent representation of hyponym and hypernym embeddings. We then compute the similarity between e_q (query's latent representation) and e_h (corresponding hypernym's representation) using cosine similarity. The following approaches have been addressed in the literature and will serve as supervised baselines for our progress.

2. **Projection Method:**
   A projection model in the context of neural networks is generally a part of the architecture that involves transforming input data (such as embeddings) into a new space. This transformation is typically achieved through linear layers (or matrices), which can project high-dimensional data into a lower or differently structured dimensional space. This is beneficial for:
   Dimensionality Reduction: Reducing the number of features to improve model efficiency and potentially enhance generalization capabilities.
   Feature Extraction: Creating new features that capture essential aspects of the data more effectively.
   Improving Model Performance: Helping the model to focus on the most relevant aspects of the data by transforming it into a representation that enhances the signal-to-noise ratio.
   Embedding Exploding: The term "embedding exploding" is not standard, but it likely refers to an issue similar to the exploding gradients problem, which occurs when large error gradients accumulate and result in very large updates to the network's weights during training. This can cause a model to diverge and lead to numerical instability. This phenomenon is particularly critical in the context of:

Training Stability: Ensuring that the network learns steadily over time without the updates causing the model parameters to become too large.
Gradient Clipping: A common technique used to prevent gradients from becoming too large, which involves capping the gradients during backpropagation to not exceed a defined threshold.

Comparison and Contextual Suitability
The choice between focusing on a projection model and managing embedding exploding (if we interpret this as managing large updates due to gradients) depends on the specific needs and challenges of your machine learning project:

This approach is about transforming the data into a more useful form for the model to process. Projection models can be designed with stability in mind, ensuring that the transformations do not amplify the gradients.
Gradient management techniques ensure that even if the data or the model architecture predisposes the gradients to grow large, the training does not become unstable.

3. **Exploiting Embedding Space:**
   For hypernym discovery task, most of the methods rely on is-a hypernymic relations as their backbone for analysis.
   This leaves an opportunity to exploit the embedding space where we can use knowledge-based embeddings, specifically SenseEmbeddings for hypernym analysis. Here, we attempt to discover the hypernymic relations by exploiting linear transformations in embedding space.
   We first define sense vectors as latent continuous representations of word senses from Word2Vec. We then cluster individual word embeddings into different domain clusters using k-means clustering. We generate lexical domain vectors by concatenating hypernyms within each cluster and compare lexical vectors with pre-trained domain vectors (Word2Vec vectors in our case) using Weighted Overlap (W_O) similarity measure. We assign clusters to domains based on highest similarity scores, ensuring reliability with a similarity threshold.
   Then we prepare data by creating pairs of hypernyms and their domains from domain-all_hypernym mappings. We create hyponym and hypernym matrices composed of SenseEmbed vectors within each domain cluster. For each domain cluster a transformation matrix is learnt by minimizing a loss function that measures the difference between predicted and actual hypernym vectors.
   Finally, we derive a ranked list of probable hypernyms for unseen terms using a cosine similarity measure and finally extract the probable candidates using top-k filters.

4. **Using Transformers:**
   In order to prepare data, we created contrastive set of data where we create positive and negative pairs for training. Positive pairs consist of query-hypernym pairs where the hypernym is a true hypernym associated with the query word. Negative pairs, on the other hand, are formed by pairing the query word with randomly selected words from the vocabulary that are not actual hypernyms.

We utilized pre-trained transformer model “bert-base-uncased” to learn tasks for English (subtask 1A), Medical (subtask 2A) and Music (subtask 2B) datasets and “bert-base-multilingual-uncased” and its tokenizer for Italian (subtask 1B) and Spanish (subtask 1C).
We fine-tuned on these pre-trained BERT models on each language-specific dataset, leveraging the embedding representations learned by BERT to predict hypernyms. We chose the best model on the basis of validation accuracy. This transfer learning approach enabled the models to capture language-specific semantic relationships and generalize to predict hypernyms across different domains.

Experiments:

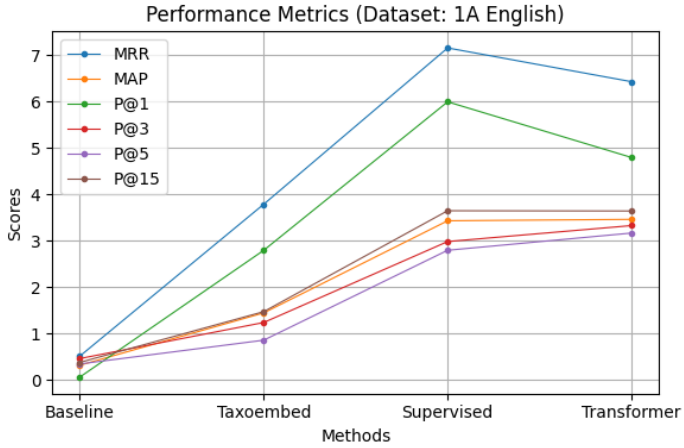
We evaluate the performance of our model on SemEval-2018 Task 9 [7] benchmark for hypernym discovery.
This shared task consists of five different subtasks covering both general-purpose (multiple languages-English, Italian, and Spanish) and domain-specific (Music and Medicine domains) tasks. For each subtask, a large textual corpus, a vocabulary including all valid hypernyms and a training and testing set of hyponyms and its gold hypernyms are provided.
Three metrics were used for the performance evaluation:

- **Mean Average Precision (MAP):** For a given query word, average precision (AP) is the average of the correctness of each obtained hypernym from the search space. MAP is the mean of this value among all queries in the data set.
- **Mean Reciprocal Rank (MRR):** Since MAP ignores the exact rank of the true hypernyms, we introduce the Mean Reciprocal Rank (MRR) metric which focuses on the top results performance. It is the average of the reciprocal ranks over all queries. The reciprocal rank of an individual query is the reciprocal of the rank in which the first true hypernym is returned.
- **Precision at K (P@K):** Precision at K is the proportion of the top-K results that are true hypernyms of a given query.

Results & Plots

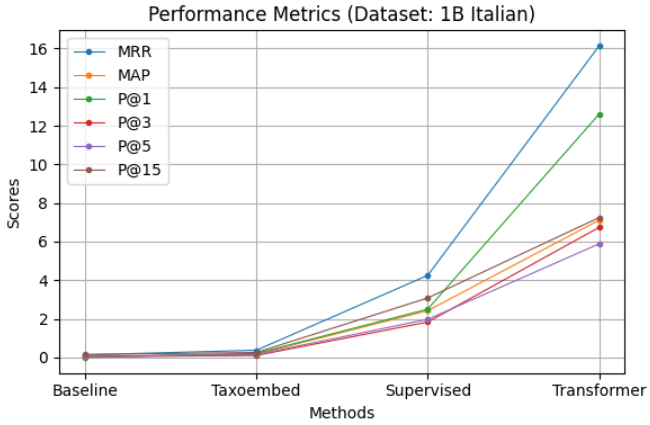
Dataset: 1A English

	Baseline	Taxoembed	Supervised	Transformer
MRR	0.5187	7.7926	7.1614	5.4335
MAP	0.3223	3.4517	3.4367	2.4670
P@1	0.0667	6.8000	6.0000	3.8000
P@3	0.4667	3.2444	2.9889	2.3333
P@5	0.3511	2.8633	2.8011	2.1722
P@15	0.3840	3.4781	3.6528	2.6473



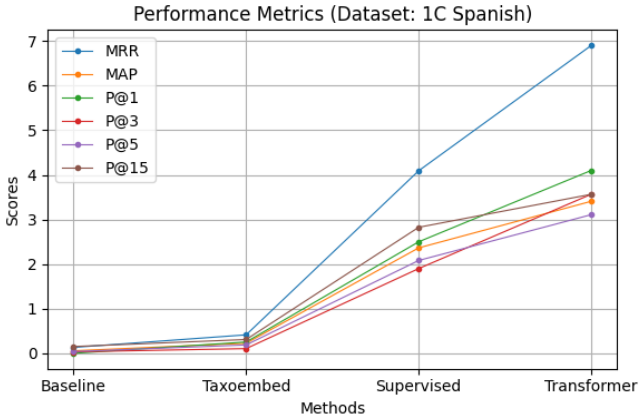
Dataset: 1B Italian

	Baseline	Taxoembed	Supervised	Transformer
MRR	0.1322	0.3722	4.2498	16.1288
MAP	0.0572	0.1704	2.4218	7.1068
P@1	0.0000	0.2075	2.5000	12.6000
P@3	0.0333	0.1037	1.8333	6.7333
P@5	0.0400	0.1487	1.9733	5.8833
P@15	0.1519	0.2547	3.0798	7.2409



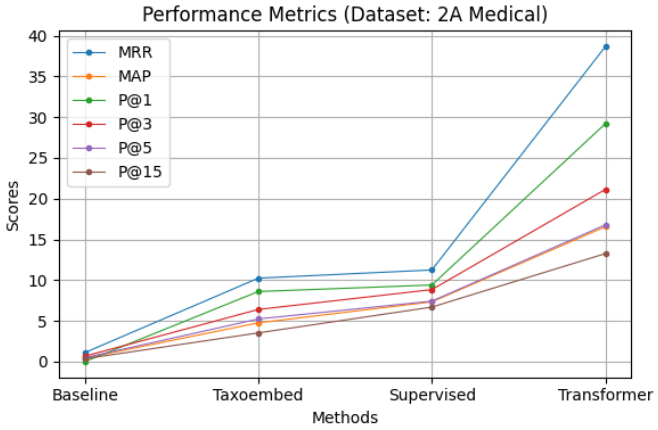
Dataset: 1C Spanish

	Baseline	Taxoembed	Supervised	Transformer
MRR	0.1322	0.4141	4.0927	6.9026
MAP	0.0572	0.2255	2.3638	3.4084
P@1	0.0000	0.2532	2.5000	4.1000
P@3	0.0333	0.1055	1.9000	3.5667
P@5	0.0400	0.1878	2.0817	3.1083
P@15	0.1519	0.3091	2.8252	3.5648



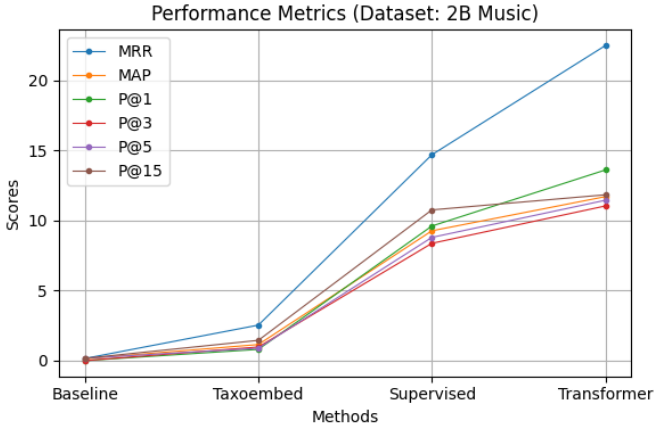
Dataset: 2A Medical

	Baseline	Taxoembed	Supervised	Transformer
MRR	1.1033	10.2430	11.2471	38.6679
MAP	0.4224	4.7574	7.3344	16.5555
P@1	0.0000	8.6000	9.4000	29.2000
P@3	0.6667	6.4000	8.8333	21.1333
P@5	0.4700	5.2333	7.4200	16.7833
P@15	0.3509	3.5233	6.6920	13.2547



Dataset: 2B Music

	Baseline	Taxoembed	Supervised	Transformer
MRR	0.15	2.5316	14.6869	22.4762
MAP	0.13	1.1458	9.2596	11.6979
P@1	0.00	0.8016	9.6000	13.6000
P@3	0.00	0.9686	8.3667	11.0333
P@5	0.17	0.8818	8.7900	11.4400
P@15	0.16	1.4444	10.7514	11.8290



QUERY	Gold	Predictions
Aneurysm (Concept)	Malady,body structure, disease, vascular disease, disorder, …	Sickness, body_structure, clinical_symptom, clinical_finding, …
Maliciousness (Concept)	Malevolence, distaste, hatred, hate, malignity	Disorder, harm, dislike, misfortune, wrongful_act, animosity, …
Bread (Concept)	Foodstuff, food product	Crust, tater, piece_of_cloth, cooking_ingredient, durable_good, consumer_durables, …
Drumstick (Concept)	Instrument, musical instrument	Aerophone, idiophone, drum, membranophone, percussion, …

Table: Examples of predictions made by our system on the test queries of multiple datsets.

Conclusion

Now for most datasets, the Transformer model worked better. The reason might be:

Language's Structure: Italian and Spanish, have a lot of consistent patterns in how words are formed and sentences are put together. Transformers are great at understanding these patterns over long texts, which helps them get better at figuring out the context or meaning in texts. This skill is especially useful for tasks that need understanding of how words relate to each other, like figuring out broader terms from specific ones.

Pre-training and Fine-tuning: Since the transformer was initially pre-trained on a large amount of corpus which encompasses text written with the same words as in our dataset, it would already have a good grasp of its context before it even starts learning the task of hypernym discovery. This background knowledge helps it perform better when fine-tuning for specific tasks.

Language-based Tokenization: Since we use bert multilingual model which has been pre-trained on 102 languages including Spanish and Italian, its able to represent the words context better than any other model trained on the limited dataset provided.

Model Complexity: BERT models' complexity, enabled by transformer architectures and large-scale pretraining, allows them to capture intricate linguistic patterns and contextual information, leading to superior classification performance across various natural language understanding tasks.

References

[1] Jose Camacho-Collados, Claudio Delli Bovi, Luis Espinosa-Anke, Sergio Oramas, Tommaso Pasini, Enrico Santus, Vered Shwartz, Roberto Navigli, and Horacio Saggion. 2018. SemEval-2018 Task 9: Hypernym Discovery. In Proceedings of the 12th International Workshop on Semantic Evaluation, pages 712–724, New Orleans, Louisiana. Association for Computational Linguistics.

[2] Gabriel Bernier-Colborne and Caroline Barrière. 2018. CRIM at SemEval-2018 Task 9: A Hybrid Approach to Hypernym Discovery. In Proceedings of the 12th International Workshop on Semantic Evaluation, pages 725–731, New Orleans, Louisiana. Association for Computational Linguistics.

[3] Luis Espinosa-Anke, Jose Camacho-Collados, Claudio Delli Bovi, and Horacio Saggion. 2016. Supervised Distributional Hypernym Discovery via Domain Adaptation. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 424–435, Austin, Texas. Association for Computational Linguistics.

[4] Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. Linear algebraic structure of word senses, with applications to polysemy. Transactions of the Association for Computational Linguistics, 6:483–495, 2018.

[5] Yuhang Bai, Richong Zhang, Fanshuang Kong, Junfan Chen, and Yongyi Mao. Hypernym discovery via a recurrent mapping model. In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 2912–2921, Online, August 2021. Association for Computational Linguistics.