A PROMPTS

When constructing contrastive learning training data, we use GPT's classification to obtain L_{pos} and L_{neg}. For each entity in the top-T of L_0, determine whether its attributes are consistent with S^{pos} (S^{neg}). If it is consistent, GPT should output 1; otherwise, GPT should output 0. Next, these entities that GPT deems consistent with the attributes of S^{pos} (S^{neg}) will be merged with S^{pos} (S^{neg}) to form L_{pos} (L_{neg}). The prompt to classify entity is shown in Table 13.

In GenExpan, each round uses three example entities to construct $Prompt_g$, which is used to generate entities that are semantically similar to them. $Prompt_g$ is as shown in Table 14.

In the process of enhancing GenExpan with Chain of Thought prompting, we employ LLM to generate fine-grained class names. The prompt, $Prompt_c$, is shown in Table 15.

B IMPLEMENTATION DETAILS OF RETEXPAN

In RetExpan, sentences are tokenized using the WordPiece tokenizer and then fed into a 12-layer Transformer initialized with $BERT_{BASE}$ weights. To optimize training efficiency and preserve semantic knowledge learned by BERT, we freeze the first 11 layers of the encode, thus only the last layer is fine-tuned. When tokenizing, to ensure that mentions of entities in long sentences are not truncated, we have implemented entity focus, guaranteeing the presence of entities within the sentence. During expansion, we use one-shot expansion to directly obtain the preliminary expansion results L_0. In RetExpan with contrastive learning strategy, each epoch during training alternates between computing and optimizing entity prediction loss and contrastive learning loss.

To ensure that the encoder acquires knowledge from the corpus, we trained it for 20 epochs on 8 RTX 3090 GPUs. The hyperparameters for training learning rate, batch size, weight decay, label smoothing factor η were set to 4e-5, 128, 1e-2, and 0.075 respectively.

C IMPLEMENTATION DETAILS OF GENEXPAN

In GenExpan, we use LLaMA-7b as the base model. Initially, we train for 1 epoch on the corpus using 6 A100 GPUs. The training hyperparameters learning rate, batch size, gradient accumulation steps, weight decay, gradient clipping are set to 1e-5, 4, 8, 1e-4, and 1.0 respectively.

During entity generation in GenExpan, we utilize prefix-constrained beam search with a beam size of 40 to generate 40 entities at a round. For entity selection, we compute the positive similarity score for each entity and select those whose scores are in the top 0.7 as the results for the current round. When no new entity is generated for 20 consecutive rounds, the generation process will end and move to re-ranking.

D DETAILS OF ULTRAWIKI

UltraWiki contains 10 fine-grained semantic classes, which fall under 5 coarse-grained semantic categories: Organization, Location, Product, Person, and Miscellaneous. The number of entities in each fine-grained semantic class ranges from 45 to 952. Each fine-grained semantic class has 2 to 3 attributes, and depending on the combination of attributes, several ultra-fine-grained semantic classes can be derived from a single fine-grained class. Detailed data is displayed in Table 12.

Table 11: Types of ultra-fine-grained semantic classes. CLS.: Semantic Class

$\lvert\mathcal{A}^{pos}\rvert$	$\lvert\mathcal{A}^{neg}\rvert$	#Ultra-fine-grained CLS.
1	1	238
1	2	5
2	1	9
2	2	7
3	3	2

The number of combinations from positive and negative attributes varies significantly, leading to substantial differences in the number of ultra-fine-grained semantic classes. The count of UltraWiki's positive and negative attributes and their corresponding ultra-fine-grained classes are presented in Table 11.

Table 12: Fine-grained semantic classes detail. CLS.: Semantic Class

Coarse CLS.	Fine-grained CLS.	#Entities	#Ultra-fine-grained CLS.	Attributes
Organization	Canada universities	99	10	<Loc-Province>, <Type>
Location	China cities	675	50	<Province>, <Prefecture>
	Countries	190	68	<Continent>, <Driving-Side>, <Per-Capita-Income>
	US airports	370	74	<Role>, <Loc-State>
	US national monuments	112	12	<Loc-State> <Agency>
Product	Mobile phone brands	159	7	<Loc-Continent>, <Status>
	Percussion instruments	128	10	<Type>, <Source-Continent>
Person	Nobel laureates	952	11	<Prize>, <Gender>
	US presidents	45	5	<Party>, <Birth-State>
Miscellaneous	Chemical elements	118	14	<Period>, <Phase-at-R.T.>

Table 13: The prompt used to select the entities that are consistent with the attributes of S^{pos} (S^{neg}) in the top-T entities of L_0 to construct L_{pos} (L_{neg}).

I have a task that involves classifying candidate entities based on their alignment with a seed entity set. The seed entities are grouped together because they share certain attributes, referred to as seed attributes. I will provide a list of seed entities along with their seed attributes. Additionally, I have a list of candidate entities that are similar to the seed entities but may not necessarily share the same seed attributes. I need you to identify the seed attributes and use them to classify each candidate entity into one of two categories: 1) consistent with the seed entity set in terms of seed attributes, or 0) inconsistent with the seed entity set in terms of seed attributes. For the given N candidate entities, please output N values, each being 1 or 0, indicating whether each candidate is consistent (1) or inconsistent (0) with the seed entity set based on the seed attributes.

Input:
Seed entities: [Mark Twain, Ernest Hemingway, F. Scott Fitzgerald]
Candidate entities: [J.K. Rowling, Stephen King, Agatha Christie, John Steinbeck, Harper Lee, Charles Dickens, Virginia Woolf], total 7 entities
Output:
"result": [0,1,0,1,1,0,0]

Input:
Seed entities: [Golden Retriever, German Shepherd, Labrador Retriever]
Candidate entities: [Bengal Tiger, Beagle, Siberian Husky, African Elephant, Pug], total 5 entities
Output:
"result": [0,1,1,0,1]

Input:
Seed entities: [{Entity1}, {Entity2}, {Entity3}]
Candidate entities: [{Entity1'}, {Entity2'}, {Entity3'}, ...], total {} entities
Output:

Table 14: The prompt used to generate entities that are semantically similar to 3 given entities.

iron, copper, aluminum and zinc.
math, physics, chemistry and biology.
{Entity1}, {Entity2}, {Entity3} and ______

Table 15: The prompt used to generate a class name that covers the given entities.

Generate a class name that accurately represents the following entities. This class name should encompass all the given entities and reflect their shared characteristics. Examples: [Tiger, Lion, Cheetah] → Big Cats [Shakespeare, Tolstoy, Hemingway] → Famous Authors [Mercury, Venus, Mars] → Planets in the Solar System [{Entity1}, {Entity2}, {Entity3}] → ______