Modeling Background Knowledge with Frame Semantics for Fine-grained Sentiment Classification

Anonymous ACL submission

Abstract

Few-shot learning via in-context learning (ICL) is widely used in NLP, but its effectiveness is highly sensitive to example selection, leading to performance variance. To address this, we introduce BACKGEN, a framework to generate structured Background Knowledge (BK) as an alternative to example-based prompting. Our approach leverages Frame Semantics to identify recurring conceptual patterns in a dataset, clustering similar instances based on shared event structures and semantic roles. Using an LLM, we synthesize these patterns into generalized knowledge statements, which are then incorporated into prompts to enhance contextual reasoning beyond individual sentence interpretations. We apply BACKGEN to Sentiment Phrase Classification (SPC), where sentiment polarity often depends on implicit commonsense knowledge. Experimental results with Mistral-7B and Llama3-8B show that BK-based prompting consistently outperforms standard few-shot approaches, yielding up to 29.94% error reduction[1].

1 Introduction

Few-shot learning has become a standard approach in NLP, enabling models to generalize from limited labeled data. In particular, *in-context learning* (ICL) (Brown et al., 2020) allows large language models (LLMs) to perform tasks without parameter updates, relying instead on a well-designed prompt that includes relevant examples (Dong et al., 2024; Liu et al., 2022a; Lu et al., 2022; Wu et al., 2023). However, ICL suffers from high variance due to its sensitivity to example selection (Zhang et al., 2022; Köksal et al., 2023; Pecher et al., 2024a). Prior research has attempted to mitigate this issue by selecting examples based on informativeness (Liu et al., 2022a; Liu and Wang, 2023; Köksal et al., 2023), representativeness (Levy et al., 2023), or learnability (Song et al., 2023), but these methods often come at a high computational cost.

A complementary approach is *knowledge prompting*, where explicit background knowledge (BK) replaces example-based selection in prompts. Prior work has explored using LLM-generated knowledge for commonsense reasoning (Liu et al., 2022b) or integrating structured knowledge from external sources (Baek et al., 2023). In this paper, we hypothesize that BK can be particularly useful for Sentiment Phrase Classification (SPC), where the goal is to determine the sentiment polarity of a target phrase in a given text.

SPC is especially challenging when the sentiment of a phrase is context-dependent. Consider the sentence "*The government phases out fossil fuels.*" The phrase "*phases out*" usually has a negative connotation, as it denotes abandonment. However, in the context of environmental policies, to phase out fossil fuels generally has a positive connotation. By relying on surface-level heuristics rather than contextual understanding, a zero-shot LLM may misclassify this instance. To address such kind of ambiguities, BK can provide crucial guidance. For example, knowing that "*the fact that a public entity wants to remove something related to green initiatives is perceived negatively*" and that "*public entities' intention to reduce non-renewable energy sources is seen as a positive step*" allows for a more accurate classification of the instance above. Without this information, the model runs the risk of drawing incorrect inferences or hallucinating reasoning patterns.

ICL typically addresses these issues and mitigates the negative impact of missing context by injecting example sentences into the prompt. However, in tasks involving short texts, the relationship between a support example and the test instance may be weak or even nonexistent, reducing the effectiveness of example-based prompting. Instead,

[1] We will release our code, dataset and BK upon paper acceptance with license to open and distribute.

structured BK provides a more reliable alternative, as it captures the higher level generalizations that underpin sentiment-bearing expressions.

In scenarios where we have annotated examples but do not perform fine-tuning, an alternative approach is to transform these examples into structured knowledge statements that generalize beyond individual instances. The goal is to construct a BK repository where each entry captures recurring conceptual patterns that can support multiple examples from the original dataset.

To achieve this, we propose a methodology for clustering similar examples and extracting their underlying commonalities. Instead of selecting instances arbitrarily, we group them based on shared semantic properties and identify the minimal conceptual structure that describes their sentiment polarity in both positive and negative contexts. The clustering process leverages Frame Semantics (Fillmore, 1985), as it provides a structured representation of situations by encoding events, participants, and their relationships. This enables us to generalize beyond lexical choices and focus on the core elements that shape sentiment interpretation. Once structured, the extracted knowledge is verbalized using an LLM, producing natural language statements that encapsulate the core sentiment-related concepts within each cluster. These statements are then injected into the prompt as BK, replacing explicit few-shot examples. This approach aims at mitigatating performance variance due to instance selection (Zhang et al., 2022) and enhances the model's ability to reason over sentiment phrases in context, particularly in ambiguous cases.

Experiments with two LLMs show that integrating BK into prompts systematically improves performance over zero-shot and few-shot learning, yielding a 26-29% error reduction. These results confirm that structured BK enhances sentiment classification by providing essential context and reducing misinterpretations.

The remainder of this paper is organized as follows. Section 2 reviews related work, Section 3 describes the proposed methodology, Section 4 presents experiments and results, and Section 6 concludes with future directions.

2 Related Works

Few-shot learning via ICL. The ICL (Brown et al., 2020) has an essential role in solving many NLP tasks as it allows the LLM to learn some examples via specific template (then, this technique is called as few-shot prompting) without updating the model parameters (Dong et al., 2024; Liu et al., 2022a; Lu et al., 2022; Wu et al., 2023). Unfortunately, the classical few-shot prompting is very sensitive to sample selection strategies (Zhang et al., 2022; Köksal et al., 2023; Pecher et al., 2024a). Despite many techniques that have been introduced to solve that problem (Liu et al., 2022a; Liu and Wang, 2023; Köksal et al., 2023; Levy et al., 2023; Song et al., 2023; Pecher et al., 2024b), most of them come at a high computational cost since the procedure to retrieve complex examples should be run for each instance, leading to a new ICL approach called knowledge prompting.

Knowledge Prompting. A new approach of ICL was introduced to inject knowledge to the prompt where the knowledge is retrieved from a particular source or generated based on the instance. Guu et al. (2020) and Lewis et al. (2020) inject documents to LLM so that the model can retrieve answers from them. Baek et al. (2023) gives additional information to the LLM by retrieving knowledge graph triplet knowledge and converting it to strings to be injected to the prompt. Liu et al. (2022b) generate knowledge for each instance to be added to the prompt. In knowledge prompting via knowledge retrieval, a problem arises if the selected knowledge is not close enough to the instance. This can lead the model to a confusion and later it to give a wrong result. Meanwhile, the knowledge generation method proposed by Liu et al. (2022b) may produce hallucination since it simply asks the model to generate knowledge based on the instance only, without giving a context, thus leading the LLM to give a wrong answer because of misinformation. Moreover, as they generate knowledge for each instance, the computational cost of this approach is high.

Background Knowledge Prompting. In contrast with the approaches described above, we propose to inject common-sense knowledge into the prompt. We postulate that this approach can be better than the classical prompting with few-shot in terms of the number of required examples, since it synthesizes several similar examples. As BK generalizes the information, the LLM can learn the reasoning from this generalization rather than focusing on a specific input-output pair. Moreover, our proposed method does not rely on specific knowledge sources as in the case of knowledge prompting

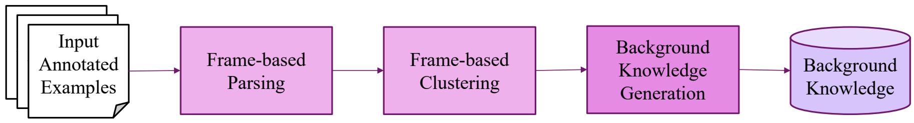

Figure 1: The BACKGEN pipeline.

via knowledge graph retrieval. The proposed BK generation is inspired by Shah et al. (2017) and Basile et al. (2018) who propose to utilize frame semantics theory to build default knowledge by extracting frames from raw texts, cluster them, and finally extract the prototypical frame from that cluster. Nevertheless, our approach differs from theirs in that our goal is to synthesize the clustered frame into BK in the form of natural language via LLM prompting.

3 BACKGEN: A BK Generation Framework

The BACKGEN framework is a structured pipeline for generating Background Knowledge (BK) to support Sentiment Phrase Classification (SPC). As shown in Figure 1, it consists of three main steps: (i) **Frame-based Parsing**, where semantic frames and their elements are extracted from annotated examples; (ii) **Frame-based Clustering**, which groups similar frames to identify shared conceptual structures; and (iii) **Background Knowledge Generation**, where a generative model verbalizes the common information in each cluster into reusable BK.

Frame-based Abstraction for Background Knowledge. To generalize beyond individual examples, we rely on Frame Semantics (Fillmore, 1985), which models meaning through structured representations called *frames*. A frame encapsulates a conceptual scenario, consisting of a *Lexical Unit* (LU) and its associated *Frame Elements* (FEs), which define roles such as agents, attributes, or affected entities. Unlike lexical approaches, frames capture abstract relationships that recur across different linguistic expressions, enabling a more structured and reusable representation of meaning.

One of the key advantages of Frame Semantics is its ability to disambiguate lexical meaning based on conceptual structures. Consider the verb *reduce*, which can evoke different frames depending on the context: in "*The government is reducing coal power*", it evokes the frame CAUSE CHANGE OF POSITION ON A SCALE, where an AGENT actively decreases a QUANTITY. In "*The army reduced enemy resistance*", however, the verb belongs to the frame CONQUERING, where a CONQUEROR overcomes a THEME rather than simply decreasing something. If we relied only on lexical similarity we would will not be able to distinguish between these cases, whereas with frame-based parsing we can generalize meaning in a structured way that aligns with conceptual distinctions rather than surface word forms.

Beyond disambiguation, frames also facilitate generalization by capturing shared prototypical structures rather than simple text-level similarities. A key property of frames is their Frame Elements, which define the roles participating in an event. By clustering instances based on frames and their arguments (such as AGENT or ASSET) we can link sentences that share the same underlying linguistic primitive, regardless of the lexical items they use. For example, "*The government is phasing out coal power*" and "*Public authorities are limiting nuclear energy*" both evoke the CAUSE CHANGE OF POSITION ON A SCALE frame, despite differing in lexical selection. The presence of an AGENT (e.g., *government*, *public authorities*) and an ATTRIBUTE (e.g., *coal power*, *nuclear energy*) establishes a conceptual equivalence, allowing the method to identify structurally similar examples even when surface-level word similarity is low. Our aim is to go beyond traditional vector-space models, which primarily capture lexical and distributional similarity (Reimers and Gurevych, 2019), by leveraging frame semantics to identify deeper conceptual patterns.

Structuring Background Knowledge. A key step in our approach is clustering examples that evoke similar situations (frames), involve analogous participants (frame elements), and exhibit comparable role-filler relations. The objective is to group instances based on deeper structural properties, ensuring that clusters capture prototypical conceptual structures rather than surface-level resemblances. To achieve this, we structure each parsed instance as a tree representation, as illustrated in Figure 2. In this representation, the frame serves as the root node, while frame elements and lexical units form intermediate nodes. The role

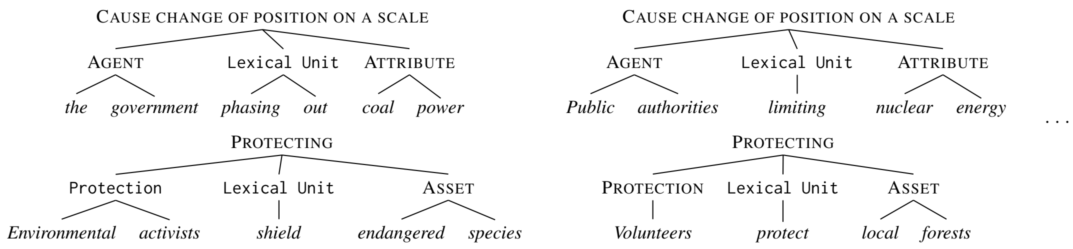

Figure 2: Examples of frame-semantic parse trees. Each tree represents a frame (root node) with its frame elements (children) and lexical unit (LU).

fillers, which instantiate the semantic arguments of the frame, appear as terminal nodes. This hierarchical encoding allows us to compare examples not merely by their lexical content but through their structural alignment within the frame-semantic paradigm.

Measuring the similarity between these structured representations requires a metric sensitive to both tree structure and semantic similarity of role fillers. We employ the Smoothed Partial Tree Kernel (SPTK) (Croce et al., 2011), which extends the Partial Tree Kernel (Moschitti, 2006) by incorporating distributed word representations into the kernel computation. This method evaluates the similarity of two trees by counting the number of shared substructures, while also weighting the contribution of lexically different but semantically related elements. In this way, two instances that share the same frame and structural configuration but differ in the lexical realizations of their role fillers will still be considered as similar. For example, the sentences "*The government is phasing out coal power*" and "*Public authorities are limiting nuclear energy*" both evoke the frame CAUSE CHANGE OF POSITION ON A SCALE, with an AGENT and an ATTRIBUTE: they are structurally analogous, and SPTK ensures that their similarity is preserved in the clustering process.

With a well-defined kernel function, we perform clustering using Kernel-based k-means (Dhillon et al., 2004), which embeds the tree structures in an implicit feature space where each dimension corresponds to a possible substructure. Unlike traditional k-means, which relies on explicit Euclidean distances, Kernel-based k-means operates in this high-dimensional space, ensuring that structurally similar examples are grouped together even if their surface forms differ significantly.

Since our task involves sentiment classification, we cluster positive and negative instances separately to maintain polarity coherence. To determine the number of clusters k, we follow a standard heuristic by setting it to the square root of the number of instances in each polarity group.

Background Knowledge Generation. The final step of BACKGEN is the generation of the structured BK from the clustered examples. At this stage, each cluster contains instances that share key semantic properties (such as the evoked frame, the roles of its participants, and the fillers of these roles) while allowing for lexical and syntactic variability. Given this structure, we employ a LLM to generate a concise generalization that synthesizes the core meaning of each cluster.

The strong capabilities demonstrated by LLMs in summarization and abstraction (Liu et al., 2024) make them well-suited for this synthesis step. The prompt, exemplified in Figure 3, instructs the model to generate a general statement based on the provided examples, explicitly leveraging Frame Semantics. The input consists of clustered sentences along with the identified frames, their definitions, and the corresponding lexical units and role assignments. Additionally, the prompt enforces a sentiment constraint, ensuring that the generated BK aligns with the sentiment orientation of the cluster. By providing explicit semantic constraints (such as frame definitions, role structures, and example sentences directly extracted from the dataset) we also aim to mitigate the risk of hallucinations, a common issue in open-ended text generation. This controlled setting ensures that the generated BK remains grounded in the linguistic and conceptual structure of the dataset while still allowing for generalization. For the example shown in Figure 3, where the clustered sentences evoke the PROTECTING frame, the generated BK is: "*The efforts of environmental activists to protect wildlife from harm are viewed as a positive and crucial step toward conservation.*" The generated statements are then stored as BK, forming a knowledge base that can later be queried to enhance in-context learning.

> Write one sentence expressing general background knowledge based on the provided input sentences that are grouped by shared situations (or frames) modeled according to Frame Semantics Theory. Each input sentence explicitly indicates the Lexical Unit (evoking the frames) and the corresponding role. Definitions of the frames will also be provided to guide the generation. Ensure that the generated text conveys a positive sentiment.
>
> Here are the definitions of the involved frame(s):
> - Protecting: Some Protection prevents a Danger from harming an Asset.
>
> Here are the input texts:
> 1. Environmental activists shield endangered species from extinction caused by poaching.
>   - Protecting:
>   - Lexical Unit (LU): shield
>   - Roles: Asset(endangered species), Protection(environmental activists)
> 2. Volunteers protect local forests from the threat of wildfires by maintaining firebreaks.
>   - Protecting:
>   - Lexical Unit (LU): protect
>   - Roles: Asset(local forests), Protection(volunteers)
>
> Answer:

Figure 3: Example prompt for generating positive Background Knowledge (BK) from clustered instances, using frames, original text, and frame definitions. The full prompt is in Appendix A, with a simplified version shown here.

Prompt Injection with BACKGEN's Generated Knowledge. Once the BK base has been populated, the next challenge is determining how to retrieve relevant information when processing a new instance. Given a new example, the goal is to retrieve BK instances that offer useful generalizations and can be integrated into a prompt in a one-shot or few-shot learning setting. An efficient retrieval strategy is needed that allows selecting representative knowledge from the BK collection. Since the BK is structured into clusters, each containing semantically related examples, retrieval can be efficiently performed by selecting the *medoid* of each cluster as an entry point. The medoid is the instance within the cluster that is closest to the *centroid* in the implicit space induced by the similarity measure (Dhillon et al., 2004), ensuring that it corresponds to a real example in the dataset. This choice allows selecting representative knowledge without needing to compare against all examples.

To retrieve the most relevant BK for a new input, we explore two alternative similarity-based approaches: one leveraging structural similarity through kernel functions and another using semantic similarity in a dense embedding space. The first method is consistent with the clustering process used in BACKGEN as it relies on the same tree-structured representation of frames. Given a new input sentence, its frame representation is extracted and compared against each cluster medoid using the adopted tree kernel function (Croce et al., 2011), selecting those entry whose medoid maximizes the kernel function, i.e. the similarity. This approach captures fine-grained structural alignment between examples, reflecting similarities in event structures and role assignments. The main advantage is that it ensures coherence between the retrieved BK and the input instance. However, it requires parsing the new input according to FrameNet, which may introduce additional computational overhead, particularly in tasks where fast inference is required. An alternative retrieval strategy is based on text similarity. Instead of relying on structured frame representations, dense vector embeddings of both the new input and the BK entry points are computed using a pre-trained language model such as BERT (Reimers and Gurevych, 2019). The similarity between the new instance and each cluster medoid is then measured using cosine similarity, based on the original, unaltered text without frame labeling. This approach avoids the need for explicit frame parsing, making it more adaptable across different tasks, and captures broader contextual relationships beyond frame-level structures. Each retrieval method presents a trade-off between interpretability and efficiency. In our hypothesis, kernel-based retrieval maintains structural coherence, making it preferable when fine-grained semantic consistency is required. Embedding-based retrieval, however, provides a more flexible and computationally efficient alternative. In the experimental section, we evaluate both approaches in terms of their effectiveness in selecting useful BK for prompt augmentation and analyze their impact on task performance. This approach also keeps retrieval efficient, as the number of cluster medoids remains at most $O(\sqrt{n})$, where n is the number of original instances.

4 Experimental Validation

Evaluating a Background Knowledge (BK) repository typically involves assessing the factual accuracy of its statements with respect to real-world knowledge. However, such an evaluation is beyond the scope of this work. Instead, we assess the practical utility of BACKGEN by measuring its impact on a downstream task-Sentiment Phrase Classification (SPC). Specifically, we examine whether integrating BACKGEN-derived BK into prompts improves the ability of a Large Language Model (LLM) to classify the sentiment polarity of a given phrase in context.

Experimental Setup. We created an SPC dataset for the environmental sustainability (ES) domain by extending the English dataset by Bosco et al. (2023) with additional language data from the social media platform X. The dataset consists of tweets discussing environmental and socio-political issues, where sentiment interpretation often relies on domain-specific background knowledge. Given the nuanced nature of these discussions, implicit assumptions and contextual understanding play a crucial role in correctly assessing sentiment polarity. The extended dataset follows the same data collection and annotation process as the original, ensuring safety regarding identifying individual people and absence of offensive content. Each message is annotated by three native English speakers from the crowdsourcing platform Prolific[2], at a rate of 9 GBP per hour, and the labels are aggregated by majority voting over sequence (Rodrigues et al., 2014). Personal information on the annotators is not disclosed in the final dataset. After filtering out the instances with no sentiment phrases, the dataset comprises 2,573 phrases (Table 1).

Attribute		Statistic
# negative phrase		1,697
# positive phrase		876
avg. span length (# token)	neg. phrase	3.09
	pos. phrase	2.69
# tweets no sentiment phrase		198
# tweets - total		1,500

Table 1: Data overview of the aggregated dataset for the sentiment phrase layer.

To parse the text with Frame Semantics, we employ LOME (Xia et al., 2021), a state-of-the-art parser for FrameNet that performs the full pipeline from lexical unit (LU) detection to complete semantic role labeling (SRL). For computing similarity between frame representations, we use the Smoothed Partial Tree Kernel (SPTK) (Croce et al., 2011), implemented within the KELP library (Filice et al., 2018), which also provides the kernel-based k-means clustering algorithm (Dhillon et al., 2004). For generating BK, we use a LLM with a structured prompt following the example in Figure 3. The prompt template, detailed in Appendix A, is designed to extract generalizable knowledge from clustered examples by summarizing their common conceptual patterns. The binary task distinguishes positive and negative sentiment. Due to class imbalance, we report per-class precision, recall, and weighted F1-score. Experiments were run on an NVIDIA A-100.

Experiment and Results. We evaluate the effectiveness of BACKGEN using two state-of-the-art open-source models, Mistral-7B[3] and Llama3-8B[4] (Dubey et al., 2024). Each model is employed both for generating background knowledge (BK) and for performing sentiment phrase classification (SPC), ensuring a consistent evaluation across the entire pipeline. The evaluation follows a 5-fold cross-validation setup. For each fold, BACKGEN is applied to 4/5 of the dataset (training set) to generate a BK database, while the remaining 1/5 is used for testing. The models are tested under different prompting conditions. In the 0-*shot* setting, the LLM receives only the input text and target phrase, without additional context. In the few-shot setting, one (1-*shot*) or two (2-*shot*) examples from the training set are provided in the prompt, either selected randomly ($_{\text{Rand}}$) or based on text similarity ($_{\text{TSim}}$). The text similarity is computed via Sentence-BERT embeddings (Reimers and Gurevych, 2019) using all-MiniLM-L6-v2[5]. For background knowledge prompting, the examples are replaced with retrieved BK entries. The retrieval process selects entries based either on frame-based similarity ($_{\text{Kernel}}$) or text similarity ($_{\text{TSim}}$), the latter computed using the same Sentence-BERT model. In both cases, the number of BK entries matches the few-shot setting, with one or two retrieved statements included in the prompt. The specific templates used for 0-*shot*, few-*shot*, and BK-*shot* prompting are reported in Appendix B[6].

[2] `https://www.prolific.com/`

[3] `https://huggingface.co/mistralai/Mistral-7B-v0.1`

[4] `https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct`

[5] `https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2`

[6] The model is expected to output *Positive* or *Negative* as the first word. If absent, the first occurrence of either label in the response is used; if neither is found, the instance is marked as unanswered, lowering recall.

Shot	Mistral-7B								
	Negative			Positive			Weighted F_1	Absolute Error	Relative Error Reduction
	Precision	Recall	F_1	Precision	Recall	F_1			
0-*shot*	0.966	0.923	0.944	0.886	0.911	0.898	0.928	0.072	-
1-*shot*$_{\text{Rand}}$	0.957	0.944	0.950	0.917	0.876	0.896	0.931	0.069	4.46%
2-*shot*$_{\text{Rand}}$	0.969	0.931	0.949	0.895	0.919	0.907	0.935	0.065	9.33%
1-*shot*$_{\text{TSim}}$	0.957	0.955	0.956	0.931	0.877	0.903	0.938	0.062	13.09%
2-*shot*$_{\text{TSim}}$	0.969	0.939	0.953	0.910	0.918	0.914	0.940	0.060	16.30%
1-*BK*$_{\text{Kernel}}$	0.964	0.947	0.955	0.909	0.925	0.917	0.942	0.058	19.50%
2-*BK*$_{\text{Kernel}}$	0.963	0.949	0.956	0.913	0.922	0.919	0.943	0.057	20.89%
1-*BK*$_{\text{TSim}}$	0.965	0.952	0.959	0.917	0.927	0.922	0.946	0.054	24.79%
2-*BK*$_{\text{TSim}}$	0.968	0.956	0.962	0.922	0.930	0.926	**0.950**	0.050	**29.94%**

Table 2: Results for the 5-fold cross-validation of the SPC task based on Mistral-7B, separately for the positive and negative classified instances in terms of Precision, Recall, and F_1 score, as well as in terms of Weighted F_1 score and Error Reduction for the two classes.

Shot	Llama3-8B								
	Negative			Positive			Weighted F_1	Absolute Error	Relative Error Reduction
	Precision	Recall	F_1	Precision	Recall	F_1			
0-*shot*	0.894	0.922	0.908	0.854	0.731	0.787	0.867	0.133	-
1-*shot*$_{\text{Rand}}$	0.866	0.954	0.908	0.888	0.706	0.786	0.866	0.134	-0.15%
2-*shot*$_{\text{Rand}}$	0.881	0.949	0.914	0.884	0.749	0.810	0.879	0.122	8.92%
1-*shot*$_{\text{TSim}}$	0.867	0.958	0.910	0.901	0.707	0.792	0.870	0.130	2.40%
2-*shot*$_{\text{TSim}}$	0.882	0.955	0.917	0.900	0.751	0.819	0.884	0.116	12.89%
1-*BK*$_{\text{Kernel}}$	0.890	0.942	0.915	0.873	0.767	0.816	0.881	0.119	10.87%
2-*BK*$_{\text{Kernel}}$	0.887	0.951	0.919	0.893	0.759	0.820	0.885	0.115	13.87%
1-*BK*$_{\text{TSim}}$	0.882	0.948	0.914	0.882	0.748	0.809	0.878	0.122	8.62%
2-*BK*$_{\text{TSim}}$	0.900	0.962	0.930	0.915	0.791	0.848	**0.902**	0.098	**26.76%**

Table 3: 5-fold cross-validation results using Llama3-8B, following the same setup as in Table 2.

In all cases, greedy search is used for token generation to ensure reproducibility and robustness.

Tables 2 and 3 summarize the results in terms of per-class precision, recall, and F_1 score. The weighted F_1 score, which accounts for class imbalance, provides an overall measure of performance. As expected, few-*shot* prompting improves over 0-*shot*, with 2-*shot* generally outperforming 1-*shot*. Additionally, selecting examples based on their similarity to the test instance ($_{\text{TSim}}$) leads to better performance than random selection ($_{\text{Rand}}$), confirming that more relevant examples contribute to better predictions. The most significant improvement comes from replacing explicit examples with structured background knowledge. In particular, BK-based prompting consistently outperforms traditional few-*shot* methods, demonstrating that synthesized knowledge captures generalizable patterns that are more informative than individual training examples. The 2-BK$_{\text{TSim}}$ configuration achieves the best weighted F_1 scores across both models, with a relative error reduction of 29.94% for Mistral-7B and 26.76% for Llama3-8B. Comparing the two BK selection methods, text similarity-based retrieval (BK_{TSim}) performs better than frame similarity-based retrieval (BK_{Kernel}). This suggests that text-based embeddings provide a more robust signal for retrieving relevant knowledge, while frame-based retrieval is more sensitive to parsing errors and the specificity of extracted structures. Overall, these results highlight the potential of structured background knowledge to enhance sentiment phrase classification. By capturing conceptual generalizations rather than relying on specific examples, BACKGEN mitigates the performance variability associated with example selection and provides a more stable and effective alternative to few-*shot* learning.

5 Error Analysis

To better understand the impact of BK on model predictions, we analyze cases where BK improves classification as well as those where it introduces errors. The goal is to identify patterns in both helpful and harmful BK selections. Given that Mistral-7B outperforms Llama3-8B, we conduct this analysis using Mistral-7B with the 2-*shot* BK selection based on text similarity.

BK is particularly useful when the sentiment polarity of a phrase depends on contextual understanding. For example, in the instance "*big problems may arise if your ductwork system is not installed correctly homeowners will encounter* ***discomfort poor indoor air quality inflated electricity bills*** *periodic repairs and in some cases complete replacement*", the 0-*shot* model incorrectly classifies the target phrase "*big problems may arise*" as positive. However, a retrieved negative BK statement, i.e., "*The* ***constant increase in expenses for various reasons, such as pollution*** *and gentrification, is a major issue that negatively impacts our lives.*", helps the model correctly reclassify the phrase as negative by reinforcing the association between financial burdens and negative sentiment.

Errors in BK selection primarily arise when (i) the retrieved BK is not sufficiently similar to the test instance, (ii) the BK is too generic, or (iii) the BK is overly specific. In cases where the retrieved BK does not align closely with the input, the model struggles to integrate it into the classification decision. Although the BK may contain relevant commonsense knowledge, it fails to provide meaningful guidance due to its semantic distance from the test instance. This can lead to the model overriding a previously correct classification, sometimes defaulting to a neutral response such as "*... The background knowledge does not provide enough information to determine the polarity of the target phrase.*" This suggests that, beyond BK retrieval, there is potential value in using model uncertainty as a signal, if no sufficiently relevant BK is found, the test instance itself may be an outlier relative to the training data. Another failure mode occurs when the retrieved BK is too generic. This typically results from poor clustering, where multiple frames that are not semantically aligned are grouped together, leading to vague or uninformative statements. For example, a BK entry such as "*Changes in policies can have a significant impact on society*" lacks specificity, making it difficult for the model to determine sentiment in a meaningful way. Overly specific BK can also introduce bias, particularly when the generated knowledge repeatedly mentions the same entity across multiple instances. Consider the instance "*you do realize* ***bill gates is heavily invested in animal agriculture*** *right he has enormous feed crop landholdings for animal ag supplying factory farms amp feedlots he also he invests in gmo cow research*", where the 0-*shot* model correctly classifies the target phrase "*heavily invested*" as positive. However, one retrieved BK statement, i.e., "*The fact that* ***Bill Gates is involved in funding and promoting synthetic meat****, despite Jeremy's disdain for him, is a disappointing turn of events.*", introduces a negative stereotype, leading the model to misclassify the phrase as negative. This suggests that the model is overfitting to entity-level associations rather than recognizing general sentiment cues. A potential solution is to refine the BK generation prompt to avoid explicit mentions of named entities, ensuring the generated knowledge remains applicable.

6 Conclusions and Future Works

We introduced BACKGEN, a framework that leverages Frame Semantics to generate structured Background Knowledge (BK) as a principled alternative to example-based prompting. Instead of selecting individual examples, BACKGENclusters semantically coherent instances, identifies shared patterns through Semantic Frames, and synthesizes generalized knowledge using an LLM. This structured BK enhances model reasoning without relying on specific instances. Applied to Sentiment Phrase Classification (SPC), where sentiment is often implicit and context-dependent, BACKGEN significantly outperforms few-shot prompting, achieving up to a 29.94% error reduction.

Future work will explore the broader applicability of structured Background Knowledge (BK) beyond sentiment analysis, particularly in tasks that require commonsense reasoning, such as commonsense question answering, where external knowledge is crucial. Finally, we aim to investigate potential biases and stereotypes in the generated BK and the underlying data. Developing automated methods to detect and mitigate such biases would enhance the reliability of knowledge prompting.

Furthermore, integrating BK into an explainability framework could enhance both sentiment classification and reasoning transparency. Frame Semantics, which models how conceptual knowledge is shared and activated in language, provides a structured basis for generalization. By linking LLM predictions to underlying frames and role structures, this approach could improve the coherence and interpretability of AI-generated explanations, making them more aligned with human cognitive representations of meaning.

Limitations

The applicability of the BK database produced in this study is currently limited to the environmental sustainability (ES) domain, and its effectiveness in other sentiment analysis tasks remains to be explored. Additionally, as BACKGEN relies on a frame parser, the quality of the generated BK is inherently dependent on the accuracy of the parser.

Another limitation is the lack of automatic analysis of the collected BK statements, which may unintentionally introduce biases or stereotypes. Since BK is generated based on clustered instances, it is possible that certain perspectives are overrepresented, reinforcing pre-existing biases in the data. Future work should focus on developing methods for detecting and mitigating such biases, ensuring that the generated BK remains neutral and representative across different domains. Moreover, investigating how BK influences model reasoning, i.e., particularly in tasks requiring explainability, could provide insights into its broader applicability beyond sentiment analysis.

Ethical Reflections

It is important to consider the potential risks of NLP tools like BACKGEN, particularly the possibility of generating biased or misleading background knowledge (BK). Without proper safeguards, BACKGEN could produce inaccurate, overly generalized, or even harmful statements that misrepresent real-world contexts, especially in sensitive areas like environmental sustainability (ES). To mitigate this risk, prompt design should be carefully refined to encourage neutral and well-grounded knowledge generation. Additionally, a verification step should be implemented to detect and filter out problematic BK, ensuring that the generated content remains accurate, unbiased, and contextually appropriate.

References

Jinheon Baek, Alham Fikri Aji, and Amir Saffari. 2023. Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. In *Proceedings of the First Workshop on Matching From Unstructured and Structured Data (MATCHING 2023)*, pages 70–98, Toronto, ON, Canada. Association for Computational Linguistics.

Valerio Basile, Roque Lopez Condori, and Elena Cabrio. 2018. Measuring frame instance relatedness. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 245–254, New Orleans, Louisiana. Association for Computational Linguistics.

Cristina Bosco, Muhammad Okky Ibrohim, Valerio Basile, and Indra Budi. 2023. How green is sentiment analysis? environmental topics in corpora at the university of turin. In *The 9th Italian Conference on Computational Linguistics (CLiC-it)*, volume 3596. CEUR-WS.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Danilo Croce, Alessandro Moschitti, and Roberto Basili. 2011. Structured lexical similarity via convolution kernels on dependency trees. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1034–1046, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Inderjit S. Dhillon, Yuqiang Guan, and Brian Kulis. 2004. Kernel k-means: spectral clustering and normalized cuts. In *KDD*, pages 551–556. ACM.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. 2024. A survey on in-context learning. *Preprint*, arXiv:2301.00234.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, and 516 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Simone Filice, Giuseppe Castellucci, Giovanni Da San Martino, Alessandro Moschitti, Danilo Croce, and Roberto Basili. 2018. Kelp: a kernel-based learning platform. *Journal of Machine Learning Research*, 18(191):1–5.

Charles J. Fillmore. 1985. Frames and the semantics of understanding. *Quaderni di semantica*, 6:222–254.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Abdullatif Köksal, Timo Schick, and Hinrich Schuetze. 2023. MEAL: Stable and active learning for few-shot prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 506–517, Singapore. Association for Computational Linguistics.

Itay Levy, Ben Bogin, and Jonathan Berant. 2023. Diverse demonstrations improve in-context compositional generalization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1401–1422, Toronto, Canada. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Hongfu Liu and Ye Wang. 2023. Towards informative few-shot prompt with maximum information gain for in-context learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15825–15838, Singapore. Association for Computational Linguistics.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022a. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.

Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. 2022b. Generated knowledge prompting for commonsense reasoning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3154–3169, Dublin, Ireland. Association for Computational Linguistics.

Yixin Liu, Kejian Shi, Katherine S He, Longtian Ye, Alexander R. Fabbri, Pengfei Liu, Dragomir Radev, and Arman Cohan. 2024. On learning to summarize with large language models as references. *Preprint*, arXiv:2305.14239.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.

Alessandro Moschitti. 2006. Efficient convolution kernels for dependency and constituent syntactic trees. In *Machine Learning: ECML 2006*, pages 318–329, Berlin, Heidelberg. Springer Berlin Heidelberg.

Branislav Pecher, Ivan Srba, and Maria Bielikova. 2024a. A survey on stability of learning with limited labelled data and its sensitivity to the effects of randomness. *ACM Comput. Surv.* Just Accepted.

Branislav Pecher, Ivan Srba, Maria Bielikova, and Joaquin Vanschoren. 2024b. Automatic combination of sample selection strategies for few-shot learning. *Preprint*, arXiv:2402.03038.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Filipe Rodrigues, Francisco Pereira, and Bernardete Ribeiro. 2014. Sequence labeling with multiple annotators. *Machine Learning*, 95(2):165–181.

Avijit Shah, Valerio Basile, Elena Cabrio, and Sowmya Kamath S. 2017. Frame instance extraction and clustering for default knowledge building. In *CEUR Workshop Proceedings*, volume 10, pages 1–10.

Chengyu Song, Fei Cai, Mengru Wang, Jianming Zheng, and Taihua Shao. 2023. Taxonprompt: Taxonomy-aware curriculum prompt learning for few-shot event classification. *Knowledge-Based Systems*, 264:110290.

Zhiyong Wu, Yaoxiang Wang, Jiacheng Ye, and Lingpeng Kong. 2023. Self-adaptive in-context learning: An information compression perspective for in-context example selection and ordering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1423–1436, Toronto, Canada. Association for Computational Linguistics.

Patrick Xia, Guanghui Qin, Siddharth Vashishtha, Yunmo Chen, Tongfei Chen, Chandler May, Craig Harman, Kyle Rawlins, Aaron Steven White, and Benjamin Van Durme. 2021. LOME: Large ontology multilingual extraction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 149–159, Online. Association for Computational Linguistics.

Yiming Zhang, Shi Feng, and Chenhao Tan. 2022. Active example selection for in-context learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9134–9148, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

A Prompts for Background Knowledge Generation

The BACKGEN framework employs two prompts to generate Background Knowledge (BK) from clusters of semantically similar instances. These clusters are formed by grouping examples that evoke the same semantic frames and share a common sentiment polarity, either positive or negative. Each cluster is then processed using the appropriate prompt:

- Clusters of positive instances use the **Positive Sentiment Knowledge Prompt** (Figure A).

- Clusters of negative instances use the **Negative Sentiment Knowledge Prompt** (Figure B).

Each prompt follows a standardized structure to ensure consistency in BK generation:

1. **Task Definition**: The prompt begins with an explicit instruction, guiding the model to generate a single sentence that captures general background knowledge from the clustered examples. This instruction specifies that the output should reflect a stereotypical generalization, either positively or negatively framed, depending on the sentiment of the cluster.

2. **Example Cluster**: The prompt includes an example cluster of semantically related instances, where each input sentence is annotated with its corresponding *frame-semantic structure*. This includes:
   - The **Lexical Unit (LU)** evoking the frame.
   - The **Frame Elements (roles)** present in the sentence.
   - The **Frame Definitions** to provide contextual understanding.

3. **Example BK Statement**: A correctly structured BK statement is provided as a reference, demonstrating the level of abstraction and generalization expected from the model. This serves as a guideline to ensure that the output captures high-level conceptual knowledge rather than instance-specific details.

4. **Target Cluster for BK Generation ("Your Turn")**: The final section of the prompt presents a new set of sentences from a different cluster (all sharing the same sentiment polarity and evoking similar frames). This part of the prompt contains placeholders (e.g., `{frame_n}`, `{text_n}`, `{LU}`, `{arguments_of_frame}`) that are dynamically populated based on the actual instances and frame annotations of the current cluster. The model is then instructed to generate a single BK statement that generalizes the semantic properties of these instances, mirroring the structure of the provided example.

Both prompts are designed to ensure that the model generates reliable, structured commonsense knowledge that can be effectively injected into prompts for downstream NLP tasks. Additionally, the framework supports variations of these prompts where the instruction is modified to generate a short paragraph instead of a single sentence, allowing for more detailed knowledge synthesis.

B Prompts for Sentiment Phrase Classification (SPC)

To evaluate different prompting strategies in Sentiment Phrase Classification (SPC), we employed three approaches:

- **Zero-shot** (Figure C): The model classifies the sentiment polarity (*positive* or *negative*) of a target phrase within a given text without additional context. The prompt explicitly instructs the model to provide a classification and a brief explanation.

- **Few-shot** (Figure D): The model is given one or two labeled examples (*1-shot* or *2-shot*) before classifying the target phrase. The examples are either selected randomly ($_{\text{Rand}}$) or based on text similarity ($_{\text{TSim}}$) with the input instance. The model cannot explicitly reference these examples in its explanation.

Write one sentence expressing general background knowledge that reflects stereotypical information, based on the input sentences provided. These sentences are grouped by shared situations (or frames) modeled according to Frame Semantics Theory. Each input sentence explicitly indicates the Lexical Unit (evoking the frames) and the corresponding role. Definitions of the frames will also be provided to guide the generation. Ensure that the generated text conveys a positive sentiment and the reason for the sentiment should be made explicit.

Example:

Here are the definitions of the involved frame(s):
- Cause_change_of_position_on_a_scale: This frame consists of words that indicate that an Agent or a Cause affects the position of an Item on some scale (the Attribute) to change it from an initial value (Value_1) to an end value (Value_2).

Here are the input texts:
1. if the tourism sector is serious about reducing its footprint they should choose real emission reductions and biodiversity protection even airlines are starting to move away from offsets fornature 4
    - Cause_change_of_position_on_a_scale:
        - Lexical Unit (LU): reducing
        - Roles: Attribute(its footprint)
2. moving away from capitalism green washing is not easy under the current systems political allegiances we live within so i commend for being bold enough to try but let us not forget that redistributing wealth and reducing consumerism must remain 1 priorities
    - Cause_change_of_position_on_a_scale:
        - Lexical Unit (LU): reducing
        - Roles: Attribute(consumerism)
3. india reduced emission intensity of its gdp by 24 per cent in 11 yrs through 2016 un via official pollution
    - Cause_change_of_position_on_a_scale:
        - Lexical Unit (LU): reduced
        - Roles: Agent(India),Attribute(emission intensity of its GDP),Difference(by 24 per cent), Speed(in 11 yrs),Time(through 2016),Means(un via official pollution)

Answer: Reducing material that is bad for the environment is a positive act.

Your Turn:

Here are the definitions of the involved frame(s):
- `{frame_1}` : `{definition_of_frame_1}`.
  ...
- `{frame_n}` : `{definition_of_frame_n}`

Here are the input texts:
1. `{text_1}`
    - `{frame_label_of_text_1}`:
        - Lexical Unit (LU): `{LU_span_of_frame_label_of_text_1}`
        - Roles: `{arguments_of_frame_label_of_text_1}`

   ...

n. `{text_n}`
    - `{frame_label_of_text_n}`:
        - Lexical Unit (LU): `{LU_span_of_frame_label_of_text_n}`
        - Roles: `{arguments_of_frame_label_of_text_n}`

Answer:

Figure A: Prompt for generating positive sentiment background knowledge. The input sentences are clustered based on shared semantic frames, and the model is instructed to generate a generalized knowledge statement that reflects a positive sentiment.

- **BK-shot** (Figure E): Instead of example-based prompting, the model receives background knowledge (BK) statements generated by BACKGEN. These statements, selected using either frame similarity ($_{\text{Kernel}}$) or text similarity ($_{\text{TSim}}$), provide generalizable knowledge to guide sentiment classification.

Each prompt follows a structured format, including:

- **Task Definition**: the goal is to classify the sentiment polarity of a given target phrase.
- **Instructions**: Constraints are provided, including the requirement for a polarity label and an explanation, without explicit reference to examples or BK.
- **Input Information**: The given text and target phrase are explicitly stated.

Write one sentence expressing general background knowledge that reflects stereotypical information, based on the input sentences provided. These sentences are grouped by shared situations (or frames) modeled according to Frame Semantics Theory. Each input sentence explicitly indicates the Lexical Unit (evoking the frames) and the corresponding role. Definitions of the frames will also be provided to guide the generation. Ensure that the generated text conveys a negative sentiment and the reason for the sentiment should be made explicit.

Example:

Here are the definitions of the involved frame(s):
- Causation: A `Cause` causes an `Effect`.
- Destroying: A `Destroyer` (a conscious entity) or `Cause` (an event, or an entity involved in such an event) affects the `Patient` negatively so that the `Patient` no longer exists.
- Cause_to_end: An `Agent` or `Cause` causes a `Process` or `State` to end.
- Cause_to_amalgamate: These words refer to an Agent joining Parts to form a Whole.

Here are the input texts:
1. water pollution is putting our health at risk unsafe water kills more people each year than war and all other forms of violence combined here are six causes of water pollution as well as what we can do to reduce it
    - Causation:
        - Lexical Unit (LU): putting
        - Roles: Cause(water pollution),Effect(our health),Cause(at risk unsafe water kills more people each year than war and all other forms of violence combined)
2. i hope izzy one day understands that we can be against pollution in all it s forms which truly is destroying our environment and health but also be smart enough to see through the carbon emissions global warming shenanigans
    - Destroying:
        - Lexical Unit (LU): destroying
        - Roles: Cause(pollution in all it s forms),Cause(which),Patient(our environment and health)
3. extinction is forever amp for all we know we have lost what we will need to fix things when it becomes obvious we have to do something technology will not end pollution of the air water soil or the contamination of our food earth cycles themselves will be the only way out of it
    - Cause_to_end:
        - Lexical Unit (LU): end
        - Roles: Cause(technology),State(pollution of the air water soil)
4. water pollution is putting our health at risk unsafe water kills more people each year than war and all other forms of violence combined here are six causes of water pollution as well as what we can do to reduce it
    - Cause_to_amalgamate:
        - Lexical Unit (LU): combined
        - Roles: Parts(all other forms of violence)

Answer: The existence of pollution and other materials that cause damage and destroy our environment is very negative.

Your Turn:

Here are the definitions of the involved frame(s):
- `{frame_1}` : `{definition_of_frame_1}`.
  ...
- `{frame_n}` : `{definition_of_frame_n}`

Here are the input texts:
1. `{text_1}`
    - `{frame_label_of_text_1}`:
        - Lexical Unit (LU): `{LU_span_of_frame_label_of_text_1}`
        - Roles: `{arguments_of_frame_label_of_text_1}`

   ...

n. `{text_n}`
    - `{frame_label_of_text_n}`:
        - Lexical Unit (LU): `{LU_span_of_frame_label_of_text_n}`
        - Roles: `{arguments_of_frame_label_of_text_n}`

Answer:

Figure B: Prompt for generating negative sentiment background knowledge. The model generates a background knowledge statement that reflects the negative sentiment conveyed by the clustered examples.

- **Additional Context**: In few-shot prompting, examples are included; in BK-shot prompting, relevant background knowledge statements are injected instead.

- **Expected Output**: The model generates a classification followed by a justification.

Figures C, D, and E illustrate the complete templates for the zero-shot, few-shot, and BK-shot prompts.

```
Task: Determine the polarity (either 'positive' or 'negative') of the target phrase from the provided text. Then, provide a short explanation for your classification. The explanation should be clear and helpful for the user to understand the choice.

Instructions:
  - The polarity output can only be 'positive' or 'negative'.
  - The first word of your answer should be your final polarity classification, then followed by your explanation.

Input:
  - Text: {text}
  - Target Phrase: {target_phrase}

Answer:
```

Figure C: Prompt `zero-shot` for SPC.

```
Task: Determine the polarity (either 'positive' or 'negative') of the target phrase from the provided text. Then, provide a short explanation for your classification. You are also provided with some examples. The explanation should be clear and helpful for the user to understand the choice.

Instructions:
  - Use the examples to help determine the polarity.
  - Note the sentiment of each example as it may assist in your reasoning.
  - The polarity output can only be 'positive' or 'negative'.
  - The first word of your answer should be your final polarity classification, then followed by your explanation.
  - The user is not aware of the examples, so you cannot refer to them explicitly in your explanation.

Input:
  - Text: {text}
  - Target Phrase: {target_phrase}

Examples:
  1. {example_text_1}. Target Phrase: {example_target_phrase_1}. Sentiment: {example_polarity_1}
     ...
  n. {example_text_n}. Target Phrase: {example_target_phrase_n}. Sentiment: {example_polarity_n}

Answer:
```

Figure D: Prompt `few-shot` for SPC.

```
Task: Determine the polarity (either 'positive' or 'negative') of the target phrase from the provided text. Then, provide a short explanation for your classification. You are also provided with potentially useful sentences reflecting background knowledge. The explanation should be clear and helpful for the user to understand the choice.

Instructions:
  - Use the background knowledge to help determine the polarity.
  - Note the sentiment of each background sentence as it may assist in your reasoning.
  - The polarity output can only be 'positive' or 'negative'.
  - The first word of your answer should be your final polarity classification, then followed by your explanation.
  - The user is not aware of the background knowledge, so you cannot refer to it explicitly in your explanation.

Input:
  - Text: {text}
  - Target Phrase: {target_phrase}

Examples:
  1. {bk_text_1}. {bk_polarity_1}
     ...
  n. {bk_text_n}. {bk_polarity_n}

Answer:
```

Figure E: Prompt BK injection shot (`bk-shot`) for SPC.