Thesis for the Master of Science

Enhanced Multi-Intent Detection with Blended Datasets

Yejin Yoon

Graduate School of Hanyang University

August 2024

Thesis for the Master of Science

Enhanced Multi-Intent Detection with Blended Datasets

Thesis Supervisor: Taeuk Kim

A Thesis submitted to the graduate school of
Hanyang University in partial fulfillment of
the requirements for the degree of Master of Science

Yejin Yoon

August 2024

Department of Artificial Intelligence Application
Graduate School of Hanyang University

This thesis, written by Yejin Yoon,
has been approved as a thesis for the Master of Science.

August 2024

Committee Chairman: Dong-Kyu Chae (Signature)

Committee member: Taeuk Kim (Signature)

Committee member: Dong-Jin Kim (Signature)

Graduate School of Hanyang University

Contents

List of Figures

List of Tables

ABSTRACT

Enhanced Multi-Intent Detection with Blended Datasets

Yejin Yoon
Dept. of Artificial Intelligence Application
The Graduate School
Hanyang University

Keywords: Multi-Intent Detection, Task-Oriented Dialogue, Spoken Language Understanding

Task-oriented dialogue (TOD) systems are commonly designed with the presumption that each utterance represents a single intent. However, this assumption may not accurately reflect real-world situations, where users frequently express multiple intents within a single utterance. While there is an emerging interest in multi-intent detection (MID), existing in-domain datasets such as MixATIS and MixSNIPS have limitations in their formulation. To address these issues, we present BlendX, a suite of refined datasets featuring more diverse patterns than their predecessors, elevating both its complexity and diversity. For dataset construction, we utilize both rule-based heuristics as well as a generative tool —OpenAI's ChatGPT— which is augmented with a similarity-driven strategy for utterance selection. To ensure the quality of the proposed datasets, we also introduce three novel metrics that assess the statistical properties of an utterance related to word count, conjunction use, and pronoun usage. Extensive experiments on BlendX reveal that state-of-the-art MID models struggle with the challenges posed by the new datasets, highlighting the need to reexamine the current state of the MID field.

1. Introduction

The successful implementation of task-oriented dialogue (TOD) systems begins with the precise recognition of user intents. By accurately discerning the queries embedded in user inputs and routing them to the relevant components, the systems can adeptly respond, thereby effectively fulfilling user requests. Generally, these systems are constructed on the assumption that each user utterance is exclusively linked to a single intent, which often diverges from practical scenarios.

Contrary to the conventional setting, the task of **Multi-Intent Detection (MID)** presents a more nuanced and comprehensive challenge for TOD systems, permitting users to express multiple intentions simultaneously. The problems posed by MID are not only more demanding but also more realistic—for reference, Gangadharaiah and Narayanaswamy (2019) reported that over half of the total instances (52%) from Amazon's in-house dialogue dataset contain multiple intents, underscoring the practical significance of the task.

Despite the ongoing interest in MID, we find it surprisingly notable that resources supporting this research direction are quite limited. Most studies on MID rely on two representative datasets, i.e., *MixATIS* and *MixSNIPS* (Qin, Xu, et al. 2020). They serve as extensions of the classic single-intent detection datasets—*ATIS* (Mansour and Haider 2021) and *SNIPS* (Coucke et al. 2018)—modified to include scenarios that involve multiple intents.

Unfortunately, in spite of its pervasive adoption within the domain, **MixX**, which includes both MixATIS and MixSNIPS (Qin, Xu, et al. 2020), has faced

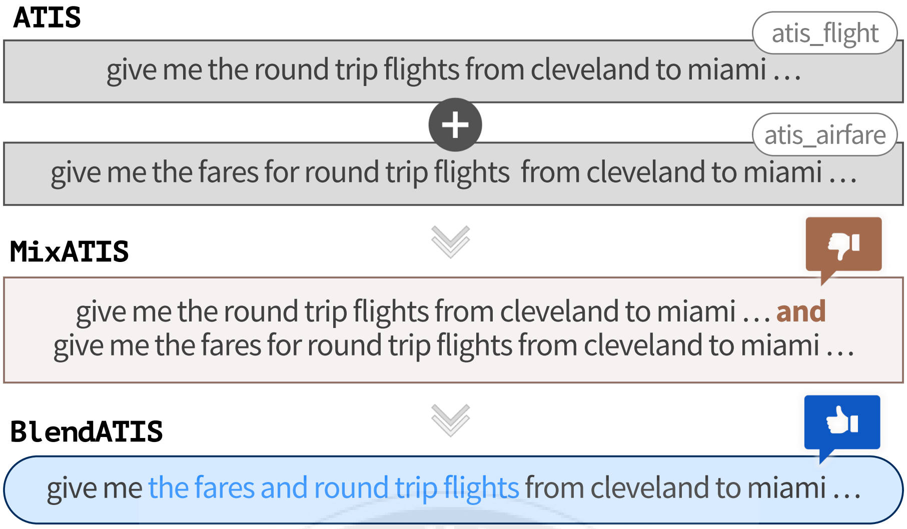
ATIS
atis_flight
give me the round trip flights from cleveland to miami ...
+
atis_airfare
give me the fares for round trip flights from cleveland to miami ...
MixATIS
give me the round trip flights from cleveland to miami ... and
give me the fares for round trip flights from cleveland to miami ...
BlendATIS
give me the fares and round trip flights from cleveland to miami ...

Figure 1.1: An example that underscores the distinct features of MixX and BlendX.

criticism for the simplicity inherent in their construction. Larson and Leach (2022a) highlighted the insufficient diversity in the connectives used to merge multiple utterances into a unified expression in the construction of MixX. That is, MixX features merely four types of coordinating conjunctions: '`and`', '`and then`', '`and also`', and '`,(comma)`', patterns that are susceptible to detection by cutting-edge models. For instance, a smart model may exploit the naïve patterns to identify the number of intents in an utterance without grasping the utterance's overall semantics. Consequently, this casts doubt on the validity of evaluations related to recent MID approaches, especially since the principal components of these evaluations generally lean on the aforementioned datasets, MixX.

In this context, we argue that obvious and urgent needs exist for establishing a more rigorous testbed for MID, as shown in Figure 1.1. Remarkably, this comes despite the minimal effort noted in the literature to address the issue. While recent work on MID largely focuses on devising new methodological schemes—evaluated within fixed, simple environments—we aim to offer orthogonal enhancement to the field by introducing a suite of upgraded datasets, dubbed **BlendX** (Yoon et al. 2024). In contrast to MixX, which relies on simple concatenations, BlendX steps beyond by simulating more realistic and complex cases often found in real-world conversations.

We explore the limitations present in the current form of MID datasets and propose curated datasets featuring more complex and varied patterns. Initially, we propose pragmatic rules for manually merging single-intent utterances, utilizing an expanded array of connectors that allow us to diverge from the simple heuristics employed in MixX.

Moreover, we consider the automated concatenation of utterances, facilitated by leveraging OpenAI's ChatGPT, `gpt-3.5-turbo-0613`. We find that, although ChatGPT is versatile, its naïve utilization struggles to merge given utterances while preserving their original intents. To maximize its efficacy, we introduce a similarity-based strategy for utterance selection, aiding the model to operate in more realistic settings without being excessively challenging.

In addition, we propose three intuitive metrics designed to assess the quality of the constructed datasets. Our analysis with these metrics demonstrates that BlendX significantly outperforms its predecessors in terms of complexity and diversity.

Lastly, we revisit state-of-the-art MID models, i.e., TFMN (Lisung Chen et al. 2022) and SLIM (Cai et al. 2022), as well as ChatGPT to evaluate their performance on BlendX. We discover that MID models struggle to adapt to the distinctive patterns present in BlendX, prompting a re-evaluation of the current state in MID literature. We also provide extensive analysis of BlendX's attributes, shedding light on its unique contributions.

2. Related Work

Single-Intent Detection Datasets We present datasets for single-intent detection, which serve as the foundation for more complex settings. One of the classic resources in the field of intent detection is the ATIS dataset (Mansour and Haider 2021), which includes utterances about 26 airline-related intents (excluding 8 of the original 26 intents in ATIS that are multi-intent classes). Meanwhile, the SNIPS dataset (Coucke et al. 2018) consists of utterances with 7 intents. These two datasets have extensions in MID settings, i.e., MixATIS and MixSNIPS Qin, Xu, et al. 2020.

Besides ATIS and SNIPS, there exist many other datasets for (single-)intent detection. We aim to expand upon these datasets by introducing them to the MID setting, similar to the cases of MixATIS and MixSNIPS. One of the candidates within the scope of our study is Banking77 (Casanueva et al. 2020), a dataset that comprises 13,083 customer service queries with 77 labeled intents specific to the banking domain. Another target of our study is CLINC150 (Larson, Mahendran, et al. 2019). This dataset encompasses 23,700 examples distributed across 150 intents within 10 domains, and offers additional distinct out-of-domain (OOD) instances. For our purpose, we exclude data instances with the out-of-domain intent, having a total of 150 intents.

In summary, our work centers on four single-intent datasets—ATIS, SNIPS, Banking77, and CLINC 150—along with their extension into MID environments. We plan to explore additional datasets such as HWU64 (Liu et al. 2019), SLURP (Bastianelli et al. 2020), and RedWood (Larson and Leach

2022b), as future work. In the following, we illustrate the current status and limitations of existing MID datasets.

Multi-Intent Detection Datasets Larson and Leach (2022a) indicate the scarcity of resources tailored for multi-intent detection. Notably, MixATIS and MixSNIPS (Qin, Xu, et al. 2020) have played a pivotal role in supporting nearly every experiment in MID. Since MixATIS and MixSNIPS were both proposed by the same group of researchers (Qin, Xu, et al. 2020), they share common characteristics. The datasets consist of utterances with up to three intents, and the distribution of inputs with different intents (1:2:3) is maintained at a ratio of 3:5:2. Furthermore, they consistently utilize the term `‘and’` (along with its variations) to merge multiple utterances into a unified one. We also note that the `‘,(comma)’` is used in the datasets exclusively when concatenating three utterances in a row.

These explicit patterns can provide strong cues for models. For example, a model might learn to identify the number of intents by either (1) counting the occurrences of the conjunction `‘and’` or (2) recognizing the presence of a `‘,(comma)’` indicating three intents. If this hypothesis holds true, models trained on MixX may encounter notable performance drops when evaluated on datasets lacking or having fewer clues. We thus intend to verify our conjecture by introducing a novel suite of datasets equipped with more diverse patterns.

Lastly, the recently introduced dataset named DialogUSR (Meng et al. 2022) stands out as a significant resource for MID research. This dataset is characterized by its provision of pairs consisting of a multi-intent utterance and its corresponding single-intent sub-queries, all annotated by humans. As a result, it enables models to learn the process of dissecting a multi-intent utterance and accurately extracting the resulting single-intent sub-queries. However, its dependence on human annotations presents a clear drawback due to the associated costs. Furthermore, the dataset is built on the assumption that multi-intent utterances can be completely segmented, which may not hold true in real-world scenarios.

3. Dataset Construction

We introduce a framework to construct a novel suite of datasets tailored for multi-intent detection, termed **BlendX** (Yoon et al. 2024), as illustrated in Figure 3.1.

Furthermore, we explore methods to connect utterances, including rule-based ones plus generative models, i.e., ChatGPT. We also propose simple but effective metrics to validate the quality of the generated datasets.

Initially, we preprocess four source datasets: ATIS, Banking77, CLINC150, and SNIPS. Inspired by MixX (Qin, Xu, et al. 2020), our approach merges utterances from single-intent datasets. We broaden the research scope by incorporating datasets such as Banking77 and CLINC150 and by utilizing diverse conjunctions. We then select single-intent utterances from these datasets. These utterances are combined using both Manual and Generative approaches. It is important to note that utterances are kept separate and not mixed across datasets. Furthermore, we explore methods to connect utterances, including rule-based ones plus generative models, i.e., ChatGPT. Following the merging process, all resultant datasets are compiled to form BlendX. We particularly highlight non-trivial combinations, such as omissions, which are indicated within the blue rounded box on the rightmost side of the framework. Finally, BlendX is evaluated using three methods: custom metrics, baseline evaluation, and visualization.

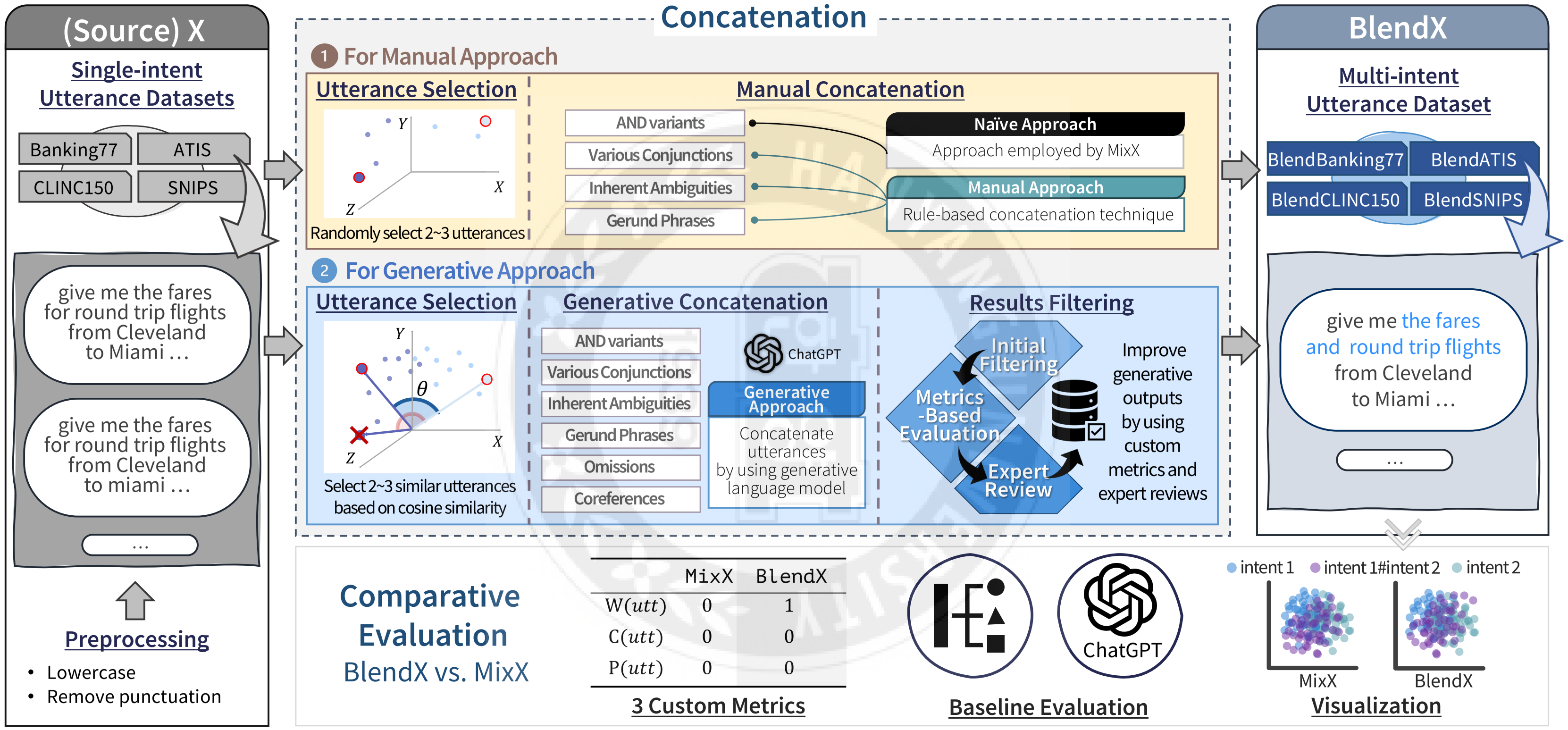

Figure 3.1: An overview of the BlendX construction framework.

3.1 Concatenation

MID datasets are typically created by merging two utterances using connectives. However, this fails to encompass the full range of ways people express multiple intents, as they often employ varied connectives or omit them entirely. To assemble multi-intent utterances with nuanced patterns and to improve upon the rule-based approach suggested by MixX, we view concatenation from two distinct aspects, as in Figure 3.2: **the complexity of concatenation** (explicit or implicit) and **the methodology of performing concatenation** (whether conducted manually or through tools such as ChatGPT.)

The complexity aspect We introduce two merging methods, each varying in complexity. These approaches target different aspects of the possible variations arising in the process of concatenation. For more details, see the middle of Figure 3.2.

1. **Explicit Concatenation:** Conjunctions are explicitly used to concatenate two or more utterances, specifically: `‘and’`, `‘and then’`, `‘and also’`, `‘,(comma)’`, `‘;(semi-colon)’`, `‘or’`, `‘before’`, `‘after’`, `‘additionally’`, `‘finally’`. We refer to the use of the four connectives as outlined in Qin, Xu, et al. (2020)—`‘and’`, `‘and then’`, `‘and also’`, `‘,(comma)’`—as the **AND variants** setting. If other conjunctions are employed, we denote it as **various conjunctions** (see Figure 3.2).

2. **Implicit Concatenation:** This approach pursues a seamless blend of utterances, minimizing the apparent usage of conjunctions. Meng et al. (2022) discovered that 62.5% of the follow-up queries in the dialogues

they gathered were either incomplete or incorrectly formulated. This inspires us to consider the four following implicit merging patterns. **Inherent ambiguity (conjunction removal)** refers to cases where conjunctions are simply removed from their original positions. Despite its simplicity, it effectively reflects the intuition that speakers tend to favor shorter utterances. In line with the same philosophy, we also consider **omissions** and **coreferences**, where redundant expressions are either eliminated or substituted with pronouns. Lastly, we employ **gerund phrases** (the `-ing` form of verbs) which are useful for emphasizing concurrency. For a clearer understanding of readers, we provide examples of each case in Table 3.3.

The methodology aspect The remaining issue is how we implement the phenomena we have specified. While we permit minor adjustments to source sentences during merging, like omissions and coreferences, we aim to retain the sentences' original structure to the greatest extent possible. In alignment with our mission, we propose two implementation strategies: manual rule-based heuristics and the utilization of ChatGPT.

1. **Naïve Approach:** It follows the original practices proposed in Qin, Xu, et al. (2020). It is Explicit Concatenation with the AND variants setting.

2. **Manual Approach:** It expands the Naïve Approach with further rule-based techniques.

3. **Generative Approach**: Facing the non-trivial challenge of implicit concatenations, we attempt to circumvent this issue with the aid of large

language models, especially ChatGPT. We instruct the tool to merge utterances while preserving their original intents and structures as much as possible, and we specifically encourage it to avoid using conjunctions such as `’and’`. Figure 3.3 shows the prompt used in the process. We provide three few-shot samples each of both successful and unsuccessful merges. The demonstration section showcases N $(= 3)$ examples of combining k $(= 2$ or $3)$ utterances, featuring both a successful and a failed case. The [QUERY] lists the sentences to be merged. ChatGPT performs the merging process by filling in the [ANSWER] part. Blue comments are for illustrative purposes only and are not part of the actual prompt. More detailed information about the prompt is available in A.1.

In our preliminary experiments, we discovered that ChatGPT often fails to combine two given sentences with the expected level of creativity and naturalness. Consequently, contrary to our expectations, the initial results from ChatGPT closely reproduced those from the Manual Approach, which heavily relies on directly using conjunctions. For reference, Table 3.1 displays a few examples of failed and successful outputs from ChatGPT.

Nevertheless, we noted some instances where ChatGPT excels in producing high-quality utterances (e.g., the bottom part of Table 3.1) that could not be simply achieved through rule-based heuristics. We highlight that manually designing such examples requires an exceptional level of expertise and effort, and they are not readily achievable even by human annotators. Therefore, in the subsequent section, we investigate techniques in terms of utterance selection to better condition ChatGPT for creating more reliable data samples.

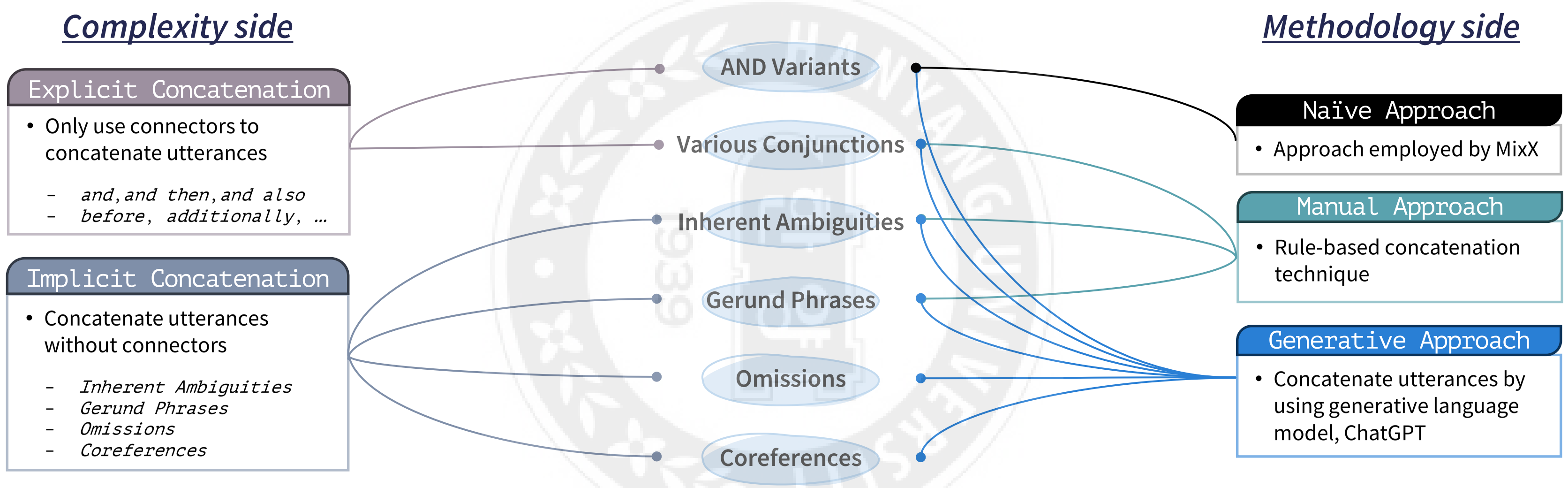

Figure 3.2: Illustration of the complexity (Left) and methodology (Right) aspects of concatenation.

Success	Utterance 1	Utterance 2	Concatenation	Implication
X	weather for frankfort (`GetWeather`)	3 out of 6 for the last album (`RateBook`)	frankfort's weather gets a 3 out of 6 rating for the last album	Intent (`RateBook`) removed
	show me the fares for a first class ticket from baltimore to dallas (`atis_airfare`)	i would like the time of all flights from san francisco to pittsburgh on sunday (`atis_flight_time`)	show me the fares for a first class ticket from baltimore to dallas and i would like to know all flights from san francisco to pittsburgh on sunday	Intent changed (from `atis_flight_time` to `atis_flight`)
	what is mci (`atis_abbreviation`)	list la (`atis_city`)	combine the sentences: "what is mci?" and "list la".	Failed to merge
O	play the playlist funtime activity (`AddToPlaylist`)	add adele onto funtime activity playlist (`PlayMusic`)	play the playlist, adding adele to the funtime activity playlist	Gerund phrase
	how do i freeze my bank account (`account_blocked`)	do you know why my bank account is frozen (`freeze_account`)	how do i freeze my bank account, and do you know why it is frozen	Coreference

Table 3.1: Failed and successful results of the Generative Approach and their implications.

Instructions: You are a native English speaker. Combine the following sentences as one single sentence naturally.

# *N* iterations

Example N**:** [(utt$_1$, intent$_1$), ..., (utt$_k$, intent$_k$)]

- **Good Answer:** [SAMPLE ANSWER]
  # An ideal result of concatenating example utterances
- **Bad Answer:** [SAMPLE ANSWER]
  # An unwanted or incorrect result of concatenating example utterances

Query: [QUERY]

Answer: [ANSWER]

Figure 3.3: Prompt design for the Generative Approach.

3.2 Utterance Selection

In the original MixX setting, source utterances for concatenation are randomly chosen without specific criteria. Our study introduces an additional selection approach rooted in utterance embedding similarity. For a given pair of utterances, we employ SBERT (Reimers and Gurevych 2019) to compute the cosine similarity of their sentence embeddings (i.e., the `[CLS]` token embeddings). We only include the sentence pair in data construction only if their score exceeds a certain threshold τ. τ is empirically set to 0.7. For some cases, we set it to 0.4 to achieve the proper training-dev-test split ratio. Table 3.2 ensures that we use semantically similar utterances rather than random samples even when τ=0.4. When combining three utterances into one, we choose the set where all possible pairs surpass the threshold.

Metric	SNIPS		ATIS		Banking77		CLINC150	
	Random	**Sim.**	**Random**	**Sim.**	**Random**	**Sim.**	**Random**	**Sim.**
Cosine sim.	0.105	0.746	0.214	0.758	0.212	0.748	0.093	0.749
Error rate (↓)	16%	**14%**	41%	**10%**	22%	**9%**	19%	**13%**
W(utt, 2)(↑)	27.38%	**44.87%**	10.17%	**27.78%**	**34.62%**	30.77%	30.86%	**31.03%**
C(utt, 2)(↑)	**8.33%**	1.28%	3.39%	**4.44%**	**28.21%**	15.38%	**25.93%**	3.45%
P(utt, 2)(↑)	3.57%	**10.26%**	1.69%	**12.22%**	10.26%	**20.88%**	3.70%	**14.94%**

Table 3.2: Comparison of Random and Similarity-Based (Sim.) utterance selection across datasets when applied to ChatGPT.

Consequently, we propose using the two following methods for utterance selection.

1. **Random Selection:** As in MixX, utterances are randomly chosen without any specific rule. Random Selection accounts for the potential scenarios in spoken language, where inputs are often noisy or ungrammatical.

2. **Similarity-Based Selection:** Utterance pairs are selected based on their similarity scores.

We utilize Random Selection for the Naïve and Manual Approach, while the Gerative Approach is combined with the Similarity-Based Selection. Prior experiments suggest the Manual Approach is not significantly impacted by utterance selection. To justify our approach, Table 3.2 presents an experiment for demonstrating the effectiveness of Similarity-Based Selection when integrated with ChatGPT. We also find that Sim. leads to a reduced error rate in ChatGPT's data generation.

3.3 Three Custom Metrics

Beyond formulating the data construction process for MID, our goal also includes developing a method to quantitatively assess the quality of the generated datasets. To achieve this, we introduce three custom metrics, based on the hypothesis that the complexity and diversity of merged instances can be measured by analyzing variances in word count, conjunction usage, and pronoun frequency before and after concatenation.

For instance, consider a scenario in which the utterances in Table 3.3, `‘play my 88 keys playlist’` and `‘add another song to my 88 keys playlist’`

Utterance 1	play my 88 keys playlist (`PlayMusic`)
Utterance 2	add another song to my 88 keys playlist (`AddToPlaylist`)

Strategies	**Concatenation Results**	W($utt, 2$)	C($utt, 2$)	P($utt, 2$)
Explicit Concatenation	play my 88 keys playlist **and also** add another song to my 88 keys playlist	0	0	0
Implicit Concatenation				
Inherent Ambiguities	play my 88 keys playlist add another song to my 88 keys playlist	1	1	0
Gerund Phrases	add another song to my 88 keys playlist playing it	1	1	1
Omissions	play my 88 keys playlist and add another song	1	0	0
Coreferences	play my 88 keys playlist and add another song to it	1	0	1

Table 3.3: Various concatenation classes, accompanied by their examples and respective metric values.

are merged to form `‘add another song to my 88 keys playlist playing it’`(Gerund Phrase). The number of words of the two original utterances are 5 and 8, respectively. After concatenation, the number becomes 10. We can apply the same calculation for conjunctions and pronouns, realizing the fact that the concatenated utterance has three fewer words, no conjunctions, one more pronoun than the original utterances. This example provides insight into numerically estimating the degree of linguistic transformations resulting from concatenation.

Suppose a merged utterance utt, which is the concatenation of n utterances out of a total of m candidate utterances, denoted as $utt_1, utt_2, \cdots, utt_m$, where each individual utterance utt_i is labeled with an intent $intent_i$. Let $|utt|_x$ be a function which counts the number of x in utt. The element x can represent any linguistic feature within the utterance, such as words, conjunctions, or pronouns, depending on the context. Here, n represents the number of utterances that were actually concatenated to form utt, and m represents the total number of candidate utterances available for concatenation.

Let $\mathrm{W}(utt, n)$ be a function to count and subtract the number of words in utt after and before concatenation, and indicate if the value is less than 1 or not. The exact definition of $\mathrm{W}(utt, n)$ is as follows:

$$\mathrm{W}(utt, n) \overset{\text{def}}{=} \mathbf{1}_{\mathbb{Z}-\mathbb{N}}\Big(|utt|_{word} - \sum_{i=1}^{n} |utt_i|_{word}\Big). \tag{1}$$

$\mathrm{C}(utt, n)$ is similar to $\mathrm{W}(utt, n)$, but it counts conjunctions instead:

$$\mathrm{C}(utt, n) \overset{\text{def}}{=} \mathbf{1}_{\mathbb{Z}-\mathbb{N}}\Big(|utt|_{conj} - \sum_{i=1}^{n} |utt_i|_{conj}\Big). \tag{2}$$

$\mathrm{P}(utt, n)$ counts and subtracts the number of pronouns in *utt* after and before concatenation, and indicates if the value is more than 0 or not:

$$\mathrm{P}(utt, n) \overset{\text{def}}{=} \mathbf{1}_{\mathbb{N}}\Big(|utt|_{pron} - \sum_{i=1}^{n} |utt_i|_{pron}\Big). \tag{3}$$

These metrics help us identify patterns in the integrated utterances, offering insights into the complexity of the concatenation procedure. If each metric becomes 1, it indicates an omission of some words, the inclusion of conjunctions, and the presence of pronouns. As such, we expect that in an ideal scenario, all metrics would converge to one.

While our metrics can serve as indicators of the complexity inherent in a concatenation process, they are not perfect; they cannot ensure a fully accurate and complete concatenation that adheres to both semantic and syntactic norms. Still, we claim that the proposed metrics can effectively serve as a proxy for measuring the quality of the merging process, as evidenced in the following sections.

Analysis of utterance selection with ChatGPT Let us first revisit the prior experiment associated with Table 3.2. We select 100 pairs of utterances using each utterance selection strategy, resulting in a notable disparity in the average cosine similarities, as shown in the first row of Table 3.2. The **Error Rate** correlates with the concept of *intent distortion*, which refers to the occurrence of either removal or alteration of the original intentions during the merging process. This rate is calculated by taking the number of merged utterances that have *intent distortion* and dividing it by the total number of utterances that were reviewed. The row for Error Rate in the table indicates that, when combined with Similarity-Based Selection, ChatGPT makes significantly fewer errors (ranging from 2% to 31%) in merging two given sentences.

Again in Table 3.2, we provide analysis with our novel metrics. When comparing the results of Similarity-Based Selection with those of Random Selection, we note an increase in the usage frequency of pronouns across datasets and a general decline in utterance word count. Exceptionally, the trend of reduced word count does not apply to Banking77, where longer, multi-sentence utterances often resulted in the use of simple `'and'` for concatenation. These findings imply that Similarity-Based Selection results in more omissions and coreferences which are desirable.

Notably, utterances that feature omissions or coreferences tend to be more ambiguous. As such, the use of conjunctions, including `'and'`, becomes essential to maintain semantic clarity. Consequently, it is expected that the percentage of $\mathrm{C}(utt, 2)$ will decrease more with Similarity-Based Selection than with Random Selection. Note that higher values for all metrics are indicative of a higher likelihood of achieving a desirable merging process. Indeed, for connections of greater complexity, it is anticipated that all metrics will exhibit increased values. The sole exception to this rule is in instances of coreference or omission, where the value may decrease to $\mathrm{C}(utt, 2)$.

Analysis of concatenation approaches Finally, we compare the effectiveness of three concatenation techniques, i.e., Naïve, Manual, and Generative, with the new metrics. For evaluation, we produce 100 instances using each approach. Results are listed in Table 3.3. We discover that only the Manual and

Metric	SNIPS			ATIS			Banking77			CLINC150		
	Naïve	**Manual**	**Generative**	**Naïve**	**Manual**	**Generative**	**Naïve**	**Manual**	**Generative**	**Naïve**	**Manual**	**Generative**
W$(utt, 2)$(↑)	0%	**37%**	29%	0%	**36%**	18%	0%	**46%**	37%	0%	**48%**	28%
C$(utt, 2)$(↑)	0%	**56%**	10%	0%	**52%**	15%	0%	**50%**	27%	0%	**56%**	32%
P$(utt, 2)$(↑)	0%	0%	**7%**	0%	0%	**8%**	0%	0%	**13%**	0%	0%	**6%**

Table 3.4: Comparative analysis of the three concatenation approaches: Naïve, Manual, and Generative.

Generative methods enable implicit concatenation. Additionally, we find that Naïve concatenation, unsurprisingly, does not result in shorter concatenated utterances; it neither introduces pronouns nor omits conjunctions. As shown in Table 3.4, Manual demonstrates its effectiveness, consistently reducing the length of concatenated utterances by 1.2 to 2 times compared to ChatGPT. Furthermore, Manual tends to use fewer conjunctions. Surprisingly, we observe that ChatGPT favors more conjunctions than expected. Upon examining metric C$(utt, 2)$ in the table, it is evident that ChatGPT employs the conjunction `'and'` in 40% to 72% of its concatenations, indicating a possible bias towards simpler concatenation strategies. Interestingly, even when explicitly instructed to avoid using `'and'`, as detailed in A.1, ChatGPT often disregards this directive.

3.4 Dataset Details

To summarize this section, we introduce **BlendX**, a collection of enhanced multi-intent datasets, shaped by our concatenation strategies and further validated using our three newly developed metrics. We apply our framework to four single-intent datasets: ATIS, SNIPS, Banking77, and CLINC150—resulting in BlendATIS, BlendSNIPS, BlendBanking77, and BlendCLINC150. The statistics of BlendX are listed in Table 3.5.

In the preprocessing step, similar to MixX (Qin, Xu, et al. 2020), BlendX ensures that each intent has equal number of utterances in the datasets. BlendX may also necessitate duplicating certain utterances, maintaining a ratio of single-, double-, and triple-intent utterances at 3:5:2. We use single-intent

Dataset	Intents #	Training	Dev	Test	Total
BlendSNIPS	7	50,625	2,613	2,615	55,853
BlendATIS	18	20,250	1,125	1,125	22,500
BlendBanking77	77	36,390	2,009	2,021	40,420
BlendCLINC150	147	54,896	2,889	2,977	60,762

Table 3.5: Statistics of the constituents of BlendX.

datasets without rectifying their internal errors, and exclude utterances that do not correspond to specific requests. (e.g., the utterances in CLINC150 whose intents are `yes`, `no`, or `maybe`.) For the case of ATIS, each subset of data (training, dev, and test) contains unique types of intents. When adapting this to BlendATIS, we retain these characteristics, ensuring it remains relatively more challenging.

We develop the final version of BlendX using both the Manual and Generative concatenation approaches. For the Manual approach, we keep a balanced ratio for the data instances corresponding to the settings of AND variants, various conjunctions, inherent ambiguities, and gerund phrases, each at 1:1:1:1. Additionally, using the Generative approach, we create data instances that amount to half of those produced with the AND variants configuration. They encompass all potential variations of concatenation, with the expectation that they foster natural and intuitive constructions, potentially featuring omissions and coreferences.

Enhancing data quality through data filtering Through the filtering process, we successfully eliminate errors similar to those highlighted in Table 3.1. This process actively exploits the three metrics proposed in §3.3. Initially, we remove clear failures generated by ChatGPT, such as explicit mentions of an intent label or unnecessary punctuation.

Subsequently, we check all concatenated sentences using the proposed metrics, paying special attention to and filtering out instances where there is a significant discrepancy before ($\sum_{i=1}^{n} |utt_i|_x$) and after concatenation ($|utt|_x$). For example, a substantial increase in word count may indicate unnecessary paraphrasing by ChatGPT, while a significant decrease might suggest overlooked utterances during concatenation. Both $\mathrm{C}(utt, n)$ and $\mathrm{P}(utt, n)$ are also subjected to similar filtering logic.

Lastly, these filtered instances are reviewed by three human experts, with a focus on excluding instances only if they compromise the intended meaning. Instances categorized under *intent removed*, *intent changed*, and *failed to merge* were identified and removed due to their significant deviation from the original intent, as detailed in Table 3.1. This filtering process significantly enhances the quality of our datasets.

4. Experiments and Analysis

4.1 Evaluation of Our Datasets on State of the Art Models

We conduct an evaluation of several MID methods based on BlendX, as well as the original MixX.

Methods In recent years, the joint learning of intent-detection and slot filling has emerged as the *de facto* standard for these tasks. Given the MixX framework's dual provision of multi-intent and slot information for each utterance, this trend has similarly permeated MID research. (Cheng et al. 2023; Tu et al. 2023; Lisung Chen et al. 2022; Xing and Tsang 2022; Lisong Chen et al. 2022; Qin, Wei, et al. 2021; Jiang et al. 2023) However, given that our emphasis in this study is solely on (multi-)intent detection, we make modest modifications to the two most predominant supervised methods to suit our objective:

- **TFMN** (Lisung Chen et al. 2022; Cheng et al. 2023): This approach first predicts the number of intents, denoted as k, in a multi-intent utterance. Subsequently, it yields the top-k options based on the predicted probability distribution. We used a variant of the original TFMN method that considers only utterance-level supervision instead of token-level, as annotating utterances from the Generative approach with token-level labels is challenging. Since we focus on (multi-)intent detection, we find it reasonable to consider only sentence-level supervision.

- **SLIM** (Cai et al. 2022): This method decomposes multi-label classifi-

Instructions: You are an Intent Detection Model on single utterance. Detect single or more intent(s) of each utterance UP TO 3.

Example: [k pairs of a multi-intent utterance and its corresponding intents]

Query: [query]

Answer: [answer]

Figure 4.1: Prompt design for solving MID with Single-prompt Strategy.

cation into a series of binary classifications. Specifically, it gauges the likelihood of each intent using the `sigmoid` function and then collects the intents whose probability exceeds a given threshold.

Furthermore, we also adopt ChatGPT (`gpt-3.5-turbo-0613`) as an extra baseline. Figure 4.1 illustrates how we design prompts for facilitating in-context learning for MID. We employ the few-shot setting where k multi-intent utterances are provided, each associated with up to three intents. The [Answer] part is filled in to predict the intent of [Query]. A.2 presents detailed illustrations. Note that the utilization of ChatGPT contrasts with the predominantly supervised fine-tuning approaches. This sheds light on the potential of generative approaches to tackle MID with little to no examples provided.

Results The main results are listed in Table 4.1. The reported numbers represent the averages and standard deviations from five distinct executions. The symbol $*$ indicates numbers derived from our re-implementation, where we have specifically excluded joint learning with slot filling.

Model	**Split**		**Dataset (Metric: Accuracy)**			
	Training	**Test**	**SNIPS**	**ATIS**	**Banking77**	**CLINC150**
TFMN	MixX	MixX	95.68* ±0.57	77.98* ±0.57	76.61 ±1.17	85.88 ±1.03
	MixX	BlendX	52.51 ±1.86	42.51 ±1.48	37.31 ±0.81	42.45 ±2.40
	BlendX	BlendX	94.93 ±0.85	76.50 ±0.83	63.99 ±0.81	77.96 ±0.82
SLIM	MixX	MixX	95.97* ±0.23	77.10* ±0.28	83.71 ±0.88	88.67 ±0.56
	MixX	BlendX	93.51 ±0.18	72.80 ±1.48	69.89 ±0.46	73.39 ±2.46
	BlendX	BlendX	95.73 ±0.86	76.92 ±0.84	75.30 ±0.71	85.62 ±0.51
gpt-3.5-turbo	-	MixX	81.68	40.30	30.90	49.22
	-	BlendX	76.18	38.84	22.67	37.55

Table 4.1: Evaluation of three competitive MID models on MixX and BlendX.

First, we reaffirm the widely recognized observation that current supervised methods show reasonable performance when both trained and evaluated with MixX. Yet, the narrative shifts significantly when these models are evaluated on our new datasets. In this configuration (training: MixX & test: BlendX), where the test data distribution deviates from the training distribution, every model shows a significant performance drop, with some declining by up to 40%. This outcome suggests that the original MixX may lack the complexity required to comprehensively evaluate the abilities of MID methods.

On the other hand, while transitioning the training data from MixX to BlendX does lead to some performance recovery on the test set, the results do not match the original performance observed when evaluated on MixX. This implies that BlendX intrinsically possesses greater complexity, making it more challenging to master.

Additionally, we find that ChatGPT's performance on MID datasets is subpar. This indicates that despite ChatGPT's adaptability, more work is needed to optimize its application for this task.

4.2 Visualization

Figure 4.2 shows data samples from SNIPS and ATIS. They are embedded with SBERT and projected into a 2-dimensional space via t-SNE. In the upper part of the figure, the green and pink dots correspond to utterances whose intents are `PlayMusic` and `RateBook`, respectively. Meanwhile, the blue dots represent multi-intent utterances with `PlayMusic#RateBook`, sourced from MixSNIPS and BlendSNIPS. The two graphs show nearly identical distributions, implying that although BlendX utterances are expected to be more diverse and noisy, they still occupy a space where data points capture the semantics of both intents.

4.3 Ablation study for concatenation methods

To evaluate the impact of different merging methods within BlendX, we categorize subsets based on the concatenation approach and measured their accuracy using TFMN. These results are presented in Table 4.2. It is observed that the Manual method contributes the most complexity and difficulty to BlendX. We observe that the subset generated by Manual is consistently more complex than the one created using the Naïve method. Comparatively, the subset merged via the Generative approach displays greater complexity than the case of Naïve but is less intricate than the case of Manual approach. These findings indicate that the Manual approach is a highly effective method of construction, facilitating both explicit and implicit concatenation. On the other hand, the Generative method tends to yield utterances akin to explicit concatenation, albeit with occasional instances of creativity.

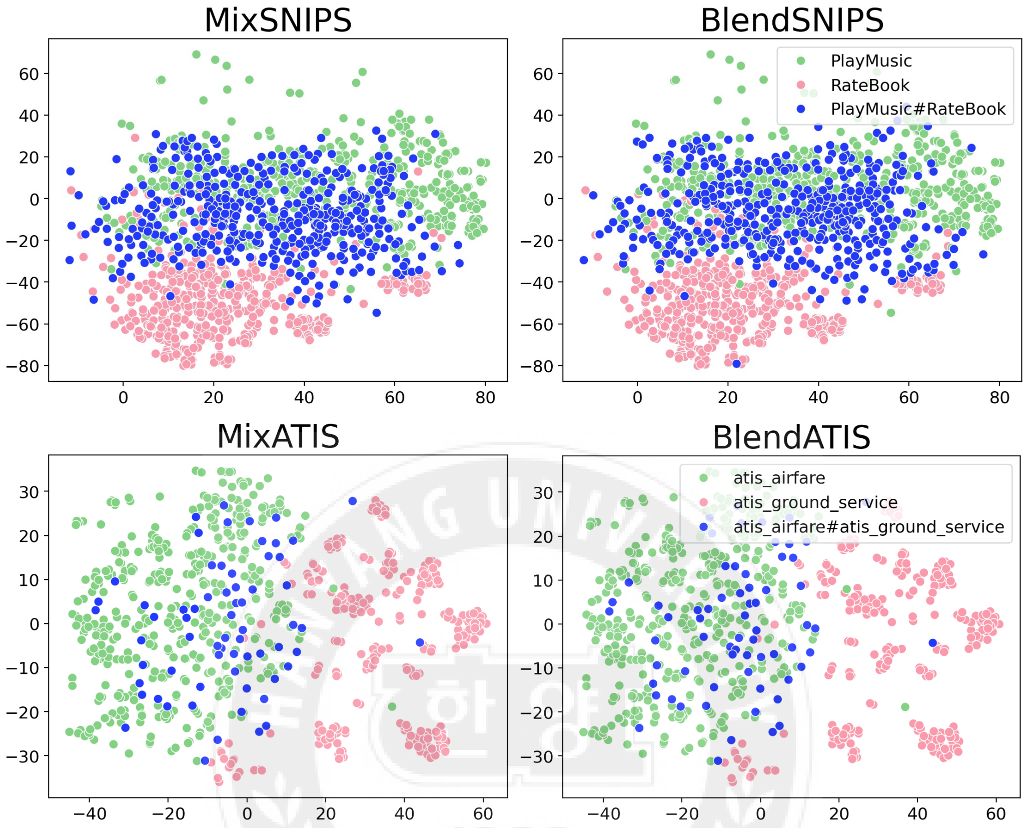

Figure 4.2: Visualization of MixX (Left) and BlendX (Right) in the $\mathbb{R}^2$ space (datasets: SNIPS, ATIS).

Data Split	Testset: BlendX (Metric: Accuracy)			
Generated By	SNIPS	ATIS	Banking77	CLINC150
Naïve	95.32	73.23	62.30	80.73
Manual	25.32	42.40	8.05	25.73
Generative	81.58	53.93	27.95	60.17

Table 4.2: Experiments with different subsets of BlendX, grouping data instances by the method used for their creation.

5. Conclusion

We highlight the disparity between utterances in class MID datasets and ones existing in more complex, real-world scenarios. To bridge this gap, we introduce BlendX, a novel suite of multi-intent datasets. The dataset is available at `https://github.com/HYU-NLP/BlendX`. We believe that the release of our multi-intent dataset framework, known as BlendX, represents a significant advancement in the field of multi-intent detection.

We present three key contributions. First, we have transcended the traditional approach by presenting 3 novel concatenation approaches: Naïve, Manual, and Generative. Second, we exhibit the effectiveness of a similarity-based strategy for sentence selection, especially when using this to augment the generative quality of ChatGPT. Third, we have devised 3 statistical metrics to validate the quality of BlendX. With extensive experiments, we verify that BlendX provides more challenging environments for MID.

However, we have identified several limitations in this work's current state and outline potential future directions to address these limitations. A prominent issue in our work is our concentration on the MID problem, overlooking the task of slot filling. The challenge becomes particularly critical when utterances are modified after concatenation, emphasizing the need for more in-depth exploration in future research to accommodate the slot-filling task. Challenges seen in single-intent datasets, such as those in Banking77 with overlapping intents (Ying and Thomas 2022), continue to exist and highlight the need for more comprehensive dataset refinement strategies. In Manual approach, we employed a variety of conjunctions; however, there is potential to improve creating more natural utterances by taking into account the relationships between sentences. The metrics we have developed, focusing on utterance length, conjunctions, and pronouns, do not fully represent linguistic complexity, necessitating further enhancements.

In conclusion, while our dataset marks a notable advancement in MID, several challenges persist. Addressing these will not only amplify our contributions but also serve the broader TOD field. We believe BlendX would facilitate more principled future research in the field.

References

[1] Rashmi Gangadharaiah and Balakrishnan Narayanaswamy. “Joint Multiple Intent Detection and Slot Labeling for Goal-Oriented Dialog”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*. 2019. URL: https://aclanthology.org/N19-1055 (cit. on p. 1).

[2] Libo Qin, Xiao Xu, Wanxiang Che, and Ting Liu. “AGIF: An Adaptive Graph-Interactive Framework for Joint Multiple Intent Detection and Slot Filling”. In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. 2020. URL: https://aclanthology.org/2020.findings-emnlp.163 (cit. on pp. 1, 5, 6, 8, 10, 11, 23).

[3] Saab Mansour and Batool Haider. *ATIS - Seven Languages*. 2021. DOI: 10.35111/g9h5-0p74 (cit. on pp. 1, 5).

[4] Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. “Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces”. In: *arXiv preprint arXiv:1805.10190* (2018) (cit. on pp. 1, 5).

[5] Stefan Larson and Kevin Leach. “A Survey of Intent Classification and Slot-Filling Datasets for Task-Oriented Dialog”. In: *arXiv preprint arXiv :2207.13211* (2022) (cit. on pp. 2, 6).

[6] Yejin Yoon, Jungyeon Lee, Kangsan Kim, Chanhee Park, and Taeuk Kim. “BlendX: Complex Multi-Intent Detection with Blended Patterns”. In: *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*. Ed. by Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue. Torino, Italia: ELRA and ICCL, May 2024, pp. 2428–2439. URL: https://aclanthology.org/2024.lrec-main.218 (cit. on pp. 3, 8).

[7] Lisung Chen, Nuo Chen, Yuexian Zou, Yong Wang, and Xinzhong Sun. “A Transformer-based Threshold-Free Framework for Multi-Intent NLU”. In: *Proceedings of the 29th International Conference on Computational*

Linguistics (COLING). 2022. URL: `https://aclanthology.org/2022.coling-1.629` (cit. on pp. 3, 26).

[8] Fengyu Cai, Wanhao Zhou, Fei Mi, and Boi Faltings. "Slim: Explicit Slot-Intent Mapping with Bert for Joint Multi-Intent Detection and Slot Filling". In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2022. DOI: `10.1109/ICASSP43922.2022.9747477` (cit. on pp. 3, 26).

[9] Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. "Efficient Intent Detection with Dual Sentence Encoders". In: *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*. 2020. URL: `https://aclanthology.org/2020.nlp4convai-1.5` (cit. on p. 5).

[10] Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. "An Evaluation Dataset for Intent Classification and Out-of-Scope Prediction". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, 1311–1316. DOI: `10.18653/v1/D19-1131`. URL: `https://aclanthology.org/D19-1131` (cit. on p. 5).

[11] Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. *Benchmarking Natural Language Understanding Services for building Conversational Agents*. 2019. arXiv: `1903.05566 [cs.CL]` (cit. on p. 5).

[12] Emanuele Bastianelli, Andrea Vanzo, Pawel Swietojanski, and Verena Rieser. "SLURP: A Spoken Language Understanding Resource Package". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2020. URL: `https://aclanthology.org/2020.emnlp-main.588` (cit. on p. 5).

[13] Stefan Larson and Kevin Leach. *Redwood: Using Collision Detection to Grow a Large-Scale Intent Classification Dataset*. 2022. arXiv: `2204.05483 [cs.CL]` (cit. on p. 5).

[14] Haoran Meng, Zheng Xin, Tianyu Liu, Zizhen Wang, He Feng, Binghuai Lin, Xuemin Zhao, Yunbo Cao, and Zhifang Sui. "DialogUSR: Complex Dialogue Utterance Splitting and Reformulation for Multiple Intent

Detection". In: *Findings of the Association for Computational Linguistics: EMNLP 2022*. 2022. URL: https://aclanthology.org/2022.findings-emnlp.234 (cit. on pp. 6, 10).

[15] Nils Reimers and Iryna Gurevych. "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 2019. URL: https://aclanthology.org/D19-1410 (cit. on p. 15).

[16] Lizhi Cheng, Wenmian Yang, and Weijia Jia. "A Scope Sensitive and Result Attentive Model for Multi-Intent Spoken Language Understanding". In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)* (2023). URL: https://doi.org/10.1609%2Faaai.v37i11.26493 (cit. on p. 26).

[17] Nguyen Anh Tu, Hoang Thi Thu Uyen, Tu Minh Phuong, and Ngo Xuan Bach. "Joint Multiple Intent Detection and Slot Filling with Supervised Contrastive Learning and Self-Distillation". In: *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2023. URL: https://doi.org/10.3233%2Ffaia230538 (cit. on p. 26).

[18] Bowen Xing and Ivor Tsang. "Co-guiding Net: Achieving Mutual Guidances between Multiple Intent Detection and Slot Filling via Heterogeneous Semantics-Label Graphs". In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2022. URL: https://aclanthology.org/2022.emnlp-main.12 (cit. on p. 26).

[19] Lisong Chen, Peilin Zhou, and Yuexian Zou. "Joint Multiple Intent Detection and Slot Filling Via Self-Distillation". In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2022. DOI: 10.1109/ICASSP43922.2022.9747843 (cit. on p. 26).

[20] Libo Qin, Fuxuan Wei, Tianbao Xie, Xiao Xu, Wanxiang Che, and Ting Liu. "GL-GIN: Fast and Accurate Non-Autoregressive Model for Joint Multiple Intent Detection and Slot Filling". In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*. 2021. URL: https://aclanthology.org/2021.acl-long.15 (cit. on p. 26).

[21] Sheng Jiang, Su Zhu, Ruisheng Cao, Qingliang Miao, and Kai Yu. “SPM: A Split-Parsing Method for Joint Multi-Intent Detection and Slot Filling”. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*. 2023. URL: `https://aclanthology.org/2023.acl-industry.64` (cit. on p. 26).

[22] Cecilia Ying and Stephen Thomas. “Label Errors in BANKING77”. In: *Proceedings of the Third Workshop on Insights from Negative Results in NLP*. 2022. URL: `https://aclanthology.org/2022.insights-1.19` (cit. on p. 31).

Appendix

A. Prompt Details

A.1 For Concatenation

Table A.1 depicts the detailed prompts of concatenating multiple utterances using ChatGPT. When two or three sentences are given as input, the prompt works to combine them into a single sentence while maintaining the essence of each sentence, mirroring the exemplary good answers. Despite numerous constraints, the generative approach often struggles to achieve concatenation, as shown in Table 3.1. We experimented with a variety of prompts, including adding or excluding components, in an effort to optimize the concatenation process. The term [Bad Answer] indicates an answer is not recommended, not necessarily incorrect. However, the variations in prompts yielded negligible differences in the quality of the results.

A.2 For Evaluating Multi-intent Detection

Table A.2 is the prompt for evaluating MID using ChatGPT in our study. Each dataset offers its distinct set of labels and examples to detect multiple intents of the given query. In this study, we expanded the dataset up to 150 single intents into multi-label settings, significantly expanding the label space. This expansion limited to the capacity to provide all examples for each prompt. The lack of a standardized approach to these few-shot prompts would be a notable challenge.

You are a native English speaker.

[**Task Definition**] Combine 2 or 3 utterances as one single utterance.

[**Goal**] The focus is on creating a single utterance that captures the essence of both ideas without unnecessary redundancy.

[**Instructions**]

- Avoid adding just punctuation.
- Don't paraphrase.
- Don't compromise the meaning of each utterance.
- Don't replace numbers with radix.
- Maintain the intent of each utterance.
- Don't forget that if a utterance starts with a verb, it's a statement.
- Do NOT use conjunctions like 'and'.
- Don't print intent directly.

# *N* iterations

[**Example 1**]

> play my 88 keys playlist (`PlayMusic`) + add another song to my 88 keys playlist (`AddToPlaylist`)
>
> [Good Answer] while playing my 88 keys playlist, add another song to it.
>
> [Bad Answer] Play my 88 keys playlist and also add another song to my 88 keys playlist.

...

[**Query**] Combine the following utterances naturally.
Inside the parentheses is the intent of each utterance: utt_0 (intent_0) + utt_1 (intent_1)

Table A.1: Specification of the prompt used for the Generative Concatenation Approach with $N = 3$.

You are an Intent Detection Model on single utterance.

[**Task Definition**] Detect single or more intent(s) of each utterance, but you can only classify UP TO 3 most plausible intents on 1 utterance.

[**Intents**] `atis_airport, atis_ground_service, atis_abbreviation, atis_city, atis_aircraft, atis_ground_fare, atis_flight, ...`

[**Answer format**] If more than one, concatenate with '#', such as {Intent}#{Intent}.
e.g. `atis_ground_fare#atis_distance`

[**Example 1**]

> [Utterance] does delta aircraft fly dc10
>
> [Answer] `atis_aircraft`

[**Example 2**]

> [Utterance] which airline has more business class flights than any other airline and what city is the airport mco in
>
> [Answer] `atis_airline#atis_city`

[**Example 3**]

> [Utterance] what does the fare code qx mean, what is the distance between pittsburgh airport and downtown pittsburgh and what is restriction ap80
>
> [Answer] `atis_abbreviation#atis_distance#atis_restriction`

[**Query**] Detect a single or up to 3 intent(s) on this following utterance. : utt

Table A.2: Specification of the prompt used for addressing MID within the in-context learning framework. (Dataset: MixATIS).

국문 요지

핵심 단어: 다중 의도 감지, 목적 지향 대화, 음성 언어 이해

목적 지향 대화 (Task-oriented dialogue; TOD) 시스템은 일반적으로 각 발화가 단일 의도를 나타낸다는 가정하에 설계된다. 그러나 이러한 가정은 실제의 다양한 상황에서 사용자들이 단일 발화 내에서 여러 의도를 한 번에 표현하는 경우를 정확하게 반영하지 못할 수 있다. 이처럼 다중 의도 감지(Multi-intent Detection; MID)에 대한 관심과 중요성이 증가하는 한편, 기존의 데이터셋인 MixATIS와 MixSNIPS는 그 구성에서 한계를 갖는다. 이러한 문제를 해결하기 위해, 본 논문에서는 개선된 데이터셋 묶음, BlendX를 제안한다. 이 데이터셋은 선행 데이터셋보다 더 다양한 패턴의 발화 병합을 특징으로, 그 복잡성과 다양성을 높이는 데에 집중했다. 데이터셋 구축을 위해, 본 논문에서는 규칙 기반 휴리스틱 접근법(Manual Approach)과 발화 선택을 위한 유사도 기반 전략이 적용된 생성형 언어모델인 OpenAI의 ChatGPT를 활용한 접근법(Generative Appraoch)을 제안한다. 제안된 데이터셋의 품질을 보장하기 위해, 단어 수, 접속사 사용 및 대명사 사용과 관련된 발화의 통계적 특성을 평가하는 세 가지 새로운 지표를 도입했다. BlendX에 대한 실험은 최첨단 성능의 MID 모델들이 새로운 데이터셋이 제기하는 도전에 어려움을 겪는다는 사실을 보여주며, MID 분야의 현재 상태를 재검토할 필요성을 강조한다.

Acknowledgment

학위 과정 동안 새로운 연구와 배움으로 가득했던 시간은, 연구실 안팎에서 함께한 동료들과 지도 교수님의 지원과 응원 덕분에 더욱 소중하게 느껴졌습니다. 이 자리를 빌어 깊은 감사의 말씀을 전합니다.

지난 석사 과정 동안 학문적으로 크게 성장할 수 있도록 치열하게 지도해주신 교수님께 진심으로 감사드립니다. 교수님 덕분에 더 높은 이상과 목표를 설정할 수 있게 되었고, 이를 향해 나아가는 길에 자신감을 가질 수 있었습니다. 학문의 길을 걸으면서 교수님의 조언과 지도가 제 연구에 깊이와 방향을 제시해 주었고, 덕분에 더 큰 목표를 향해 계속 나아갈 수 있는 용기를 얻었습니다. 앞으로도 연구를 지속하면서 교수님의 가르침을 되새기고, 끊임없는 배움을 통해 더 훌륭한 연구자로 성장하겠습니다. 바쁘신 중에도 심사에 시간을 내어주신 채동규 교수님과 김동진 교수님께도 깊은 감사를 표합니다.

연구실에서 함께 의지가 되어준 동료들에게도 감사의 말씀을 전합니다. 특히, 항상 옆에서 든든하게 자리해준 정연 덕분에 마음이 어지러운 시간을 딛고 앞으로 나아갈 수 있었습니다. 연구실 생활에 빨리 적응할 수 있게 항상 먼저 챙겨준 지이, 성실함과 열정이 모두의 귀감이 되는 지수, 신중하고 꼼꼼해서 든든한 영우, 중요한 부분을 잘 짚어내는 기정, 특유의 개성이 연구실을 밝혀준 정민, 함께 어려움을 겪으면서도 포기하지 않은 강산, 모두 같이 연구할 수 있어서 기뻤습니다. 항상 진지한 태도로 연구에 임하는 덕영, 날카로운 시선으로 한 발 앞서서 생각해보게 해준 동건, 차분한 태도로 복잡한 문제도 쉽게 풀게 도와주는 강민, 항상 진실된 모습을 보여주는 창현, 적극적인 태도와 배우려는 자세가 인상깊었던 승희, 모두에게 많이 배웠고, 감사했습니다. 바로 어제 함께했던 것 같은 진현, 영현, 태준, 성훈에게도 감사 인사를 전합니다. 여러분들과 함께 연구할 수 있어서 큰 행운이었습니다.

마음 다해 응원해주신 가족들, 특히 어머니, 아버지, 동생 상진이와 항상 제게 기댈 수 있는 소중한 친구들에게도 감사합니다. 마음이 혼란스러웠던 순간에 위로가 되어주셔서, 따뜻한 말 한마디가 큰 힘이 되었습니다. 저도 제 소중한 사람들에게 힘이 되어줄 수 있는 존재가 되고 싶습니다.

마지막으로, 학교로 돌아와 새로운 도전을 향한 부족하고 서툰 제 모습을 격려해주신 모든 분들의 지지와 배려로 성장할 수 있었습니다. 다시 한 번 진심으로 감사드립니다.

Declaration of Ethical Conduct in Research

I, as a graduate student of Hanyang University, hereby declare that I have abided by the following Code of Research Ethics while writing this dissertation thesis, during my degree program.

"First, I have strived to be honest in my conduct, to produce valid and reliable research conforming with the guidance of my thesis supervisor, and I affirm that my thesis contains honest, fair and reasonable conclusions based on my own careful research under the guidance of my thesis supervisor.

Second, I have not committed any acts that may discredit or damage the credibility of my research. These include, but are not limited to : falsification, distortion of research findings or plagiarism.

Third, I need to go through with Copykiller Program(Internet-based Plagiarism-prevention service) before submitting a thesis."

JUNE 13, 2024

Degree :	Master
Department :	DEPARTMENT OF ARTIFICIAL INTELLIGENCE APPLICATION
Thesis Supervisor :	KIMTAEUK
Name :	YOON YE JIN (Signature) Yoon

연구 윤리 서약서

본인은 한양대학교 대학원생으로서 이 학위논문 작성 과정에서 다음과 같이 연구 윤리의 기본 원칙을 준수하였음을 서약합니다.

첫째, 지도교수의 지도를 받아 정직하고 엄정한 연구를 수행하여 학위논문을 작성한다.

둘째, 논문 작성시 위조, 변조, 표절 등 학문적 진실성을 훼손하는 어떤 연구 부정행위도 하지 않는다.

셋째, 논문 작성시 논문유사도 검증시스템 "카피킬러"등을 거쳐야 한다.

2024년06월13일

학위명 : 석사

학과 : AI응용학과

지도교수 : 김태욱

성명 : 윤예진 (서명) YJ Yoon

한양대학교대학원장귀하