Generating Meaningful Topic Descriptions with Sentence Embeddings and LDA

Javier Miguel Sastre Martínez, Seán Gorman, Aisling Nugent, Anandita Pal
Accenture The Dock, R&D Global Innovation Center,
7 Hanover Quay, Grand Canal Dock, Dublin, Ireland
{j.sastre.martinez, sean.gorman, a.nugent, anandita.pal}
@accenture.com

Abstract

A major part of business operations is interacting with customers. Traditionally this was done by human agents, face to face or over telephone calls within customer support centers. There is now a move towards automation in this field using chatbots and virtual assistants, as well as an increased focus on analyzing recorded conversations to gather insights. Determining the different services that a human agent provides and estimating the incurred call handling costs per service are key to prioritizing service automation. We propose a new technique, ELDA (**E**mbedding based **LDA**), based on a combination of LDA topic modeling and sentence embeddings, that can take a dataset of customer-agent dialogs and extract key utterances instead of key words. The aim is to provide more meaningful and contextual topic descriptions required for interpreting and labeling the topics, reducing the need for manually reviewing dialog transcripts.

1 Introduction

Topic models are statistical tools for discovering the hidden semantic structure in a collection of documents/dialogs. One such widely used topic model is Latent Dirichlet Allocation (LDA, Blei et al., 2003). LDA is a hierarchical probabilistic model that represents each topic as a distribution over terms/words and represents each document/dialog as a mixture of the topics. One of the main issues with the standard LDA bag-of-words approach is that the discovered topics can be difficult to interpret, as the user is presented with only the key words per topic. Due to this, the user often needs to go through the documents/dialogs for each topic to gather more context. The ELDA (**E**mbedding based **LDA**) approach attempts to produce more interpretable topics by running the topic modeling at an utterance level. The resulting topics can be represented by the most relevant utterances per topic, giving more context to the analyst so they can better understand the topic, with little to no manual inspection of the dialogs.

Another issue with bag-of-word approaches is that they fail to capture co-reference resolution, homonymy, and polysemy. For example, the words "leave" and "depart" mean the same thing in similar contexts but will be treated as having different meanings. Conversely, one word, for example "right", can mean different things given the context but will be treated as having the same meaning. Representing text as embeddings can overcome these issues to some extent. For example, word and sentence encoders such as (Google's) Multilingual Universal Sentence Encoder (MUSE, Yang et al., 2020), Sentence-BERT (SBERT, Reimers and Gurevych, 2019), etc. can capture the meaning of sentences and words in context with no need for any text pre-processing (e.g. stop word removal, part-of-speech tagging, lemmatization etc.).

A further challenge in running LDA is that it requires to specify in advance the number of topics to generate, which can be hard to determine in cases where the domain or data is not known in detail. The ELDA approach includes a novel technique to automatically estimate the number of topics to generate for a given dataset.

We compared the topic descriptions of the ELDA approach with that of standard LDA on the MultiWOZ dataset (Han et al., 2021).

2 Related Work

Cygan (2021) employed a method of topic modeling that leverages SBERT (Reimers and Gurevych, 2019) to create rich semantic document embeddings by averaging sentence embeddings, after which documents are assigned to a cluster using HDBSCAN. Once the clusters are created, Cygan uses LDA to construct a single topic descriptor (a list of key words) over the documents of each cluster. They claim in their analysis that a small set of documents clustered together by SBERT em-

Proceedings of the SIGdial 2022 Conference, pages 244–254
Heriot-Watt University, Edinburgh, UK. 07-09, September, 2022

beddings can generate a coherent and interpretable topic, outperforming topics made from Doc2Vec (Le and Mikolov, 2014) based document embeddings. Our approach uses sentence-level topic descriptors rather than key words, and we apply a recent sentence encoder that supports multiple languages (Yang et al., 2020).

Kozbagarov et al. (2021) present another approach to generating interpretable topics by combining sentence embeddings with a topic modeling technique, though they use EM (expectation-maximization) instead of LDA and use averaged BERT word embeddings (Devlin et al., 2019) instead of a pretrained sentence encoder. Like us, they cluster the resulting sentence embeddings and estimate the probability of sentence occurrence within texts, assuming sentences within each cluster as identical. However, they apply EM on the text distribution over sentence clusters, thereby representing each topic as a probability distribution over sentence clusters. Finally, they also labeled the clusters with the closest sentence to the cluster centroid, as we do. Their experimental results show a high level of interpretability in the formed topics compared to traditional topic modeling approaches.

Moody (2016) described the lda2vec model, which builds representations over both words and documents by mixing word vectors (word2vec) with Dirichlet-distributed latent document-level mixtures of topic vectors, yielding sparse and interpretable document-to-topic proportions in the style of LDA. The topics obtained on the 20newsgroup corpus are shown to yield high mean topic coherences, correlating with human evaluations of the topics.

Dieng et al. (2020) developed an embedded topic model (ETM) which integrates topic embeddings with traditional topic models. Like in LDA, the ETM is a generative probabilistic model, where each document is a mixture of topics, and each term is assigned to one of the topics. In contrast to LDA, each term is represented by an embedding, and each topic is a point in that embedding space. The topic's distribution over terms is proportional to the exponentiated inner product of the topic's embedding and each term's embedding. The ETM claims to discover more interpretable topics even with large vocabularies that include rare words and stop words. It claims to outperform LDA in both predictive performance and topic quality and diversity as measure by topic coherence.

Our work specifically targets topic discovery in customer call conversations rather than general documents, such as news articles or publications, as in most of the related work. We have also created novel techniques in: (i) automatically deciding on the number of topics to produce and (ii) to measure the interpretability and accuracy of the produced topics.

3 Method

Given a collection of dialogs segmented into utterances, either by a speech-to-text system that includes diarization or based on metadata provided by a text-messaging system (see Table 1), ELDA applies topic modeling at the utterance level, producing topics represented by a selection of key utterances relevant to each topic. The method is split in 5 steps, namely: computing the utterance vectors (Section 3.1), clusterizing the utterance vectors (Section 3.2), auto-labeling the clusters (Section 3.3), encoding the dialogs as bags of utterance clusters (Section 3.3), and applying LDA on these bags of utterance clusters (Section 3.5), using then their corresponding cluster auto-labels as the resulting topic key items.

3.1 Utterance encoding

We first apply a sentence encoder to each utterance to obtain a vector representation. In particular, we have tested Universal Sentence Encoder (USE, Cer et al., 2018), Multilingual Universal Sentence Encoder (MUSE, Yang et al., 2020), and Sentence-BERT (SBERT, Reimers and Gurevych, 2019). Each of these embed text segments into vectors of a fixed size. In our approach, we settled on using MUSE as it supports 16 different languages and produced results comparable to the other two. Comparison was done as explained in Section 4.3, though we only present here the results obtained with MUSE to avoid repetition.

3.2 Utterance clustering

We compute groups of semantically similar utterances by clustering the set of utterance embeddings. This allows us to represent each dialog as a collection of utterance clusters/types. For the clustering, we employ a combination of k-means (MacQueen, 1967) and DBSCAN (Ester et al., 1996) algorithms in two steps. We first apply k-means to create an initial set of k clusters, with a relatively low k proportional to the total number of utterances n (e.g.,

#	Speaker	Utterance
1	CUSTOMER	Hi , I 'm looking for a train that is going to cambridge and arriving there by 20:45 , is there anything like that?
2	AGENT	There are over 1,000 trains like that . Where will you be departing from ?
3	CUSTOMER	I am departing from birmingham new street .
4	AGENT	Can you confirm your desired travel day ?
5	CUSTOMER	I would like to leave on wednesday.
6	AGENT	I show a train leaving birmingham new street at 17:40 and arriving at 20:23 on Wednesday . Will this work for you ?
7	CUSTOMER	That will , yes . Please make a booking for 5 people please
8	AGENT	I 've booked your train tickets , and your reference number is A9NHSO9Y.
9	CUSTOMER	Thanks so much .

Table 1: Sample dialog between customer and agent regarding a train booking in the MultiWOZ dataset. Note some utterances may convey more than one sentence (e.g., utterances 2, 6 and 7).

$n/5000$). Then we apply DBSCAN to the set of utterances of each initial cluster in order to avoid having to choose a final number of clusters to generate: DBSCAN creates a cluster for each set of a minimum size min_pts of transitively connected points, where 2 points are connected (or neighbors) iff they are within a maximum distance eps. Sets smaller than min_pts do not form clusters, naturally discarding rare utterances. As a drawback, DBSCAN requires to compute the distance between every pair of points, which can be time intensive for the case of large sets of utterances. By pre-clustering the set of utterances with k-means we reduce the number of distances to compute by several orders of magnitude. The two main hyperparameters of DBSCAN, eps and min_pts, have considerable impact on the quality of ELDA results. The tuning of these hyperparameters is described in Section 4.3.

3.3 Utterance cluster auto-labeling

For each utterance cluster we select the best utterance representative to serve as the cluster's label. We first compute the cluster centroid (the average of its vectors), then select the utterance whose vector is closest to the centroid. An example of the clusters and their labels can be found in appendix A.2.

3.4 Dialog encoding

To perform topic modeling on the labeled utterances clusters, we represent each document/dialog as a bag of utterance clusters (instead of a bag of words), followed by the standard LDA approach. We use a TF-IDF-like vectorizer to compute the document/dialog vectors by considering the utterance clusters as terms (i.e., we compute utterance cluster frequency-inverse document frequency). The set of document vectors form the document-cluster matrix D.

3.5 LDA topic modeling

Like k-means, the LDA algorithm requires to specify the number of topics K to compute in advance. However, it is often difficult to choose a proper value, especially for unknown domains. We propose a new approach to automatically select the number of topics by modeling the topic coverage decay using an exponential function (see Algorithm 1). The goal of this approach is to automatically discover as many real services/use cases in call center conversations as possible, at the expense of generating an excess of topics that are either redundant, subcategories of other topics, or noise.

Instead of specifying K, the algorithm requires a rough estimate of the interval $[K_{min}, K_{max}]$ comprising K. Starting from K_{min} and at step increments, an LDA topic model is computed and tested for the given document-cluster matrix D and number of topics K until a complying model is found. To test compliance of a model, each dialog is assigned to its highest probability topic, according to the model, and the coverage of each topic (proportion of total dialogs assigned to each topic) is computed. The topic coverages are sorted in descending order and an exponential function is fitted to smooth the decay curve $y = me^{tx}$ with m as the y-intercept and t as the exponent factor (refer to Figure 1). Using the inverse of the exponential function derivative, we find the frontier between

Algorithm 1 exponential_decay_LDA(D)

Input: D, document cluster matrix
Parameters: $K_{min}, K_{max}, step,$
$slope\_threshold, min\_tail\_ratio$
Output: lda_model

for each $K = K_{min}$ to K_{max} by $step$ **do**
  $lda\_model \leftarrow \text{train\_lda\_model}(D, K)$
  for each $i = 0$ to $K - 1$ **do**
    $topic\_dialogs_i \leftarrow \emptyset$
  end for
  for each dialog d **do**
    $i \leftarrow$ topic index for which d has its highest probability, according to lda_model
    $topic\_dialogs_i \leftarrow topic\_dialogs_i \cup \{d\}$
  end for
  for each $i = 0$ to $K - 1$ **do**
    $topic\_coverage_i \leftarrow \frac{|topic\_dialogs_i|}{K}$
  end for
  $X \leftarrow (0, 1, \ldots, K - 1)$
  $\text{sort\_descending}(topic\_coverage)$
  $m, t \leftarrow \text{exponential\_regression}(X, Y)$
  $x_t \leftarrow \frac{\ln(\frac{slope\_threshold}{mt})}{t}$
  $tail\_ratio \leftarrow \frac{|\{topic_i : i \geq x_t\}|}{K}$
  if $tail\_ratio \geq min\_tail\_ratio$ **then**
    break
  end if
end for

the head and the tail of the exponential function (dashed line in Figure 1), where the tail is the part of the curve with a slope below $slope\_threshold$. We then compute $tail\_ratio$, the proportion of topics in the tail, and check if it is greater than or equal to the threshold min_tail_ratio. If true, the algorithm stops and returns the corresponding model; otherwise, further LDA models for higher K values are computed until either min_tail_ratio or K_{max} is reached. By enforcing a minimum tail ratio, we expect to discover most of the relevant conversation topics while limiting the number of topics to compute. After a certain point, increasing K results in a greater number of topics in the tail region, each one covering a very small portion of the totality of dialogs.

We used the following parameter values in all our experiments: $K_{min} = 5$, $K_{max} = 60$, $step = 1$, $slope\_threshold = -0.001$ and $min\_tail\_ratio = 0.4$. For the MultiWOZ dataset, the algorithm stopped at $K = 24$ (refer to Figure 1).

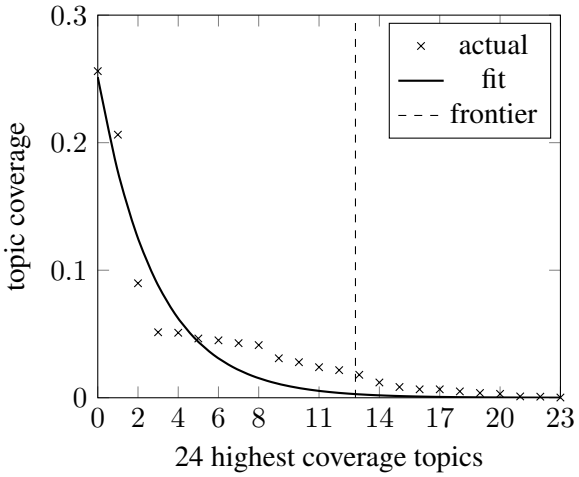

Figure 1: Plot showing the exponential decay approach for $K = 24$ (the first compliant K found) on MultiWOZ data. The $\times$'s represent the actual topic coverages, the curve denotes the best fitted exponential function and the vertical dashed line denotes the frontier between the head and the tail regions.

Each topic in the resulting model is a probability distribution of utterance clusters where each cluster is labeled with the most representative utterance. Thus, each topic can be represented by a set of key utterances, thereby providing descriptive context to the user in the process of interpreting and labeling the topics.

The ELDA result comprises a document/dialog-topic matrix (just like the standard LDA) and a topic-cluster or topic-utterance matrix (contrary to topic-word matrix of standard LDA).

4 Experiments

4.1 Data

To evaluate the quality of the ELDA approach we use the MultiWOZ dataset (Han et al., 2021), which comprises more than 10,000 annotated agent-customer dialogs across 7 domains/intents, namely: train, taxi, hotel, restaurant, attraction, police and hospital (Table 2, Figure 2). The dialogs are segmented into turns, which we use as utterances, and each dialog is annotated with the customer's intents, each dialog having at least one intent. In our case, we refer to each dialog's set of intents as its "label".

# dialogs	# utterances	# intents
10,438	224,179	7

Table 2: MultiWOZ data metrics

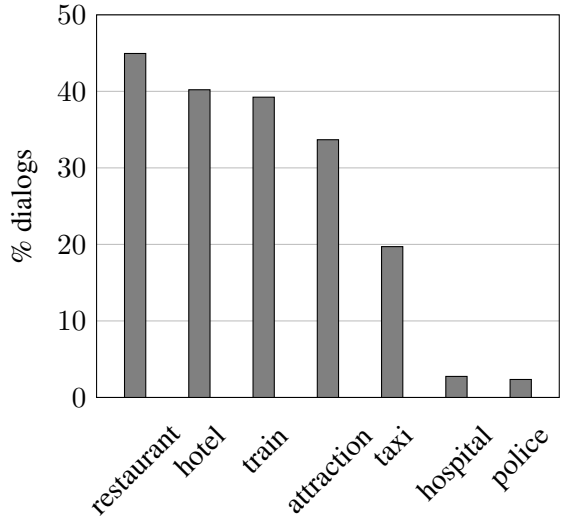

Figure 2: True intent distribution of the MultiWOZ dataset – Vertical axis denotes percentage of dialogs per intent

4.2 Evaluation methods

In this section we discuss two different aspects of evaluating ELDA. Mainly we compare ELDA's results with that of standard word-level LDA based on two evaluation criteria:

1. **Accuracy of dialog label identification**
2. **Interpretability of topic key utterances vs topic key words**

Accuracy: To measure the accuracy of a topic model, we must first manually inspect its output topics and label each topic with one of the seven MultiWOZ intents. For simplicity, we assume each topic has just one intent. For each topic, we first observe the topic key items (words for standard LDA and utterances for ELDA) and their respective scores. Giving priority to the key items with higher scores, we identify the related dominant intent and select it as the topic label (see Table 3). Topics with an equal mixture of different intents (more than one dominant intent), or those with unclear intents, are not given any label (see Table 4). We first label the bigger topics (based on topic coverage) and proceed towards the smaller ones. This strategy allows for identifying the most frequent intents first, while also considering the greatest number of dialogs in the least amount of time. Smaller topics that are subcategories of the bigger topics (e.g., Chinese restaurant booking vs restaurant booking) are given the same labels as the corresponding bigger topics.

Topic 8		
Cluster	**Cluster label**	**Score**
41	I am looking for a hotel instead of a guesthouse .	0.119
11	Is there a price range you 'd like ?	0.080
39	I need to book it for 4 people starting from saturday for 5 nights .	0.062
3	Can I get some help finding a hotel or guesthouse please ?	0.043
30	I need free parking and free wifi though .	0.034
10	I would like to book a reservation for it .	0.033
46	There are a couple of options .	0.031
23	I would like a guesthouse that is 4 stars .	0.031
34	Is there a particular area of town you 'd like to be in ?	0.029
32	I am also looking for a place to go in town .	0.027

Table 3: An example of topic with a clear dominant intent "hotel" (label given by either the oracle, annotator 1 or annotator 2 was "hotel")

Next, we assign these labeled topics to dialogs. For each dialog, we find all topics that have a probability score greater than or equal to the mean dialog-topic probability score (average of the probabilities in the dialog-topic matrix). The reason for selecting the mean as the threshold is that the topic probabilities, after being sorted for each dialog, are likely to follow a skewed distribution and the mean helps to filter out the lower probability or less frequent topics, steering the focus towards the higher probability or dominant topics for each dialog. We take the union of those dominant topics' labels as the predicted label for each dialog. Thus, each dialog will have zero, one or more of the seven MultiWOZ intents as its label. We then compare these predicted labels to the true dialog labels. Any overlap between the true and predicted dialog labels is considered a hit, and the hit rate across all dialogs

Topic 5		
Cluster	**Cluster label**	**Score**
7	Is there anything else you need ?	0.023
9	The phone number is 01223351241 .	0.023
33	Can I get the phone number , postcode , and address please ?	0.023
35	I need to book a taxi please .	0.023
44	Glad that I could help .	0.023
40	From Cambridge , which is why I asked the Cambridge TownInfo centre .	0.022
21	No , indeed .	0.022
3	Can I get some help finding a hotel or guesthouse please ?	0.022
0	I 'll take a cheap one please .	0.022
31	From where will you be departing ?	0.022

Table 4: An example of a "noisy" topic with more than one dominant intent, hence gets no label (label given by either the oracle, annotator 1 or annotator 2 was "blank")

is computed. This hit rate, or overlap score as we call it, is the accuracy of our topic model.

Interpretability: To compare interpretability of topic key words with topic key utterances, we show a few example topics obtained by running standard LDA and ELDA respectively on the MultiWOZ dataset and describe the efforts required to interpret and label them.

4.3 Experiment details

In this section, we discuss the experiments we performed to evaluate ELDA.

Baseline: For the baseline standard LDA model we start by applying a standard NLP pre-processing pipeline to the dialog words comprising lowercasing, POS tagging, lemmatization and stop word removal. We then encode the dialogs as TF-IDF vectors using the Gensim library (Řehůřek and Sojka, 2010). While encoding, we also use the inbuilt Gensim filtering utility to first remove the words that appear in more than 90% of the dialogs and in less than two dialogs, and then keep the remaining most frequent 100,000 words only. We use the described exponential decay approach to compute LDA models for different numbers of topics and for the resulting model, the topics are then manually labeled. Finally, the topics are assigned to each dialog, and an overlap score between the dialog topic labels and the MultiWOZ true labels is computed for the sake of evaluation and comparison with ELDA.

ELDA: We first run a grid search to find optimal values of the DBSCAN hyperparameters min_pts (minimum points per cluster) and eps (maximum allowed distance between neighboring points in the same cluster), computing multiple ELDA models for each combination and then calculating the overlap score between the true and predicted dialog labels. To avoid having to manually label the topics for each hyperparameter combination, we use an oracle approach: for a given topic, find the set of dialogs that are dominant using mean as the threshold, and select as topic label the most frequent MultiWOZ intent in that set of dialogs. In the case where a topic does not have dialogs above the threshold, it gets no label and will not contribute to the overlap score. We tested Gensim filtering analogous to the process used in baseline LDA on ELDA but filtering out low and high document frequency clusters barely filtered any clusters out, which in turn had little to no impact on the overlap scores. The DBSCAN density parameter values used for the grid search are as below:

- min_pts: 3 and 5
- eps: 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.6, 0.7, 0.8 and 0.9.

Comparison: In the search for optimal ELDA model, we compare the different ELDA models' overlap scores with that of baseline LDA. For fair comparison, we apply the same oracle labeling approach to both ELDA and LDA models. Then, on obtaining the optimal ELDA model, it is evaluated against the baseline LDA model using the overlap scores obtained from manual labels of two annotators. To ensure oracle labeling is consistent with manual labeling, we also compare the oracle labels and the manual labels of both the baseline LDA and optimal ELDA models.

5 Results

In this section we first report the overlap scores of the baseline LDA model and the different ELDA models, using the oracle topic labeling for all the models. Then we report the results of the comparison between the oracle labels and manual labels (from the annotators) for both the baseline LDA and optimal ELDA (best grid-search model). Next, we show a comparison of the overlap scores resulting from manual labeling obtained from the baseline LDA with the same obtained from the optimal ELDA. We also compare the true intent distribution of MultiWOZ with that produced by the manual labels of the baseline LDA and optimal ELDA. Lastly, we exhibit the topic descriptions of the three biggest topics from the baseline LDA and optimal ELDA and compare their individual level of interpretability.

5.1 ELDA optimization

The overlap score using the oracle labels of the baseline LDA model is 0.9281, showing that there is a high similarity between the predicted and the true dialog labels. This score is used as the baseline that the ELDA optimization aims to match or exceed.

The best ELDA model produced an overlap score of 0.9555 using the oracle labels for min_pts = 5 and eps = 0.5, surpassing our baseline score for LDA.

Based on these optimization results (Table 5) we expect the best ELDA model to match the baseline LDA in overlap score using the manual labels obtained from the annotators. Before that, we need to ensure that the optimization of ELDA based on oracle labels is consistent with manual labeling.

5.2 Validation of oracle labels

To validate the use of oracle labeling in optimizing the ELDA results, two annotators manually labeled the topics of the baseline LDA and the best ELDA model, and then we compared those manual labels to the oracle's labels. The results seen in Table 6 show reasonable overlaps between the oracle and manual topic labels. This validates the use of the oracle labeling as an efficient alternative to manual labeling, and so, was considered a suitable approach to enable running the ELDA optimization. Note the optimal values found for hyperparameters min_pts and eps may be extrapolable to other datasets, given that the semantic similarity distance magnitudes are given by the sentence embedding and not by the dataset. Hence, we would not need to manually annotate other datasets to repeat the tuning of the hyperparameters, which would defeat the purpose of running ELDA.

min_pts	***eps***	**# topics**	**Overlap**
3	0.2	35	0.6868
	0.25	39	0.7603
	0.3	29	0.8463
	0.35	37	0.8912
	0.4	39	0.9449
	0.45	40	0.9461
	0.475	40	0.9503
	0.5	39	0.9517
	0.525	38	0.9516
	0.6	41	0.9493
	0.7	40	0.9428
	0.8	40	0.9289
	0.9	40	0.9415
5	0.3	35	0.8818
	0.4	40	0.9488
	0.475	37	0.9538
	0.5	**38**	**0.9555**
	0.525	40	0.9303
	0.6	42	0.9428

Table 5: Overlap scores of ELDA for different values of DBSCAN hyperparameters min_pts and eps, and different number of topics, based on oracle labels (best result in bold)

Model	**# topics**	**Annotator 1**	**Annotator 2**
Baseline LDA	24	0.71	0.75
Best ELDA	38	0.71	0.76

Table 6: Average annotator overlap scores between oracle and manual topic labels

5.3 Evaluation of ELDA

As discussed earlier, we evaluate ELDA against LDA based on two aspects: accuracy and interpretability. We measure both on the best ELDA model and the baseline LDA model.

5.3.1 Accuracy

The overlap scores of the best ELDA model are evaluated against that of the baseline LDA according to the manual annotations of their respective topics obtained from the two annotators (see Table 7). From the results we observe an average of

Model	# topics	Annotator 1	Annotator 2	Annotator avg.
Baseline LDA	24	0.8921	0.9157	0.904
Best ELDA	38	0.9040	0.8827	0.8934

Table 7: Average annotator overlap scores of the baseline LDA and best ELDA models

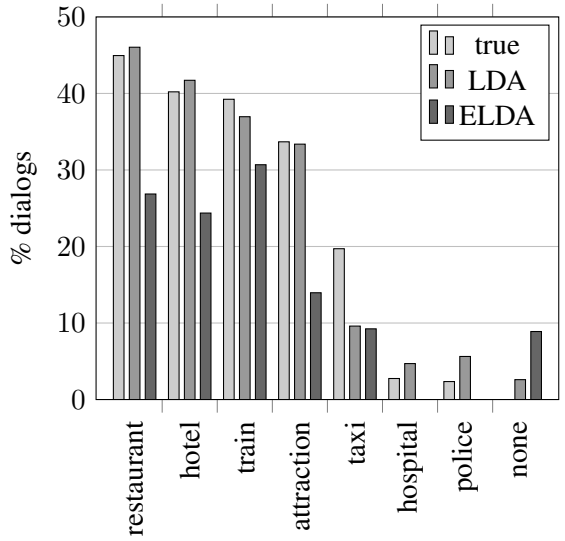

Figure 3: True vs baseline LDA vs best ELDA – Comparison of the intent ratio over the dialogs (% of dialogs per intent)

89% overlap with the best ELDA as opposed to an average of 90% overlap with the baseline LDA.

We also compare the true ratio of intents across the dialogs (% dialogs per intent) with that obtained from the best ELDA and baseline LDA models (Figure 3). For both LDA and ELDA, we show the average ratio of intents for each of the two annotators. Observing the plot, we see that ELDA has successfully identified the most frequent five of the seven intents, however LDA performs better in matching the true intent ratio. Potentially, further fine-tuning of the ELDA approach may improve these results.

5.3.2 Interpretability

In this section we analyze the top key items for the topics of the best ELDA model (along with the clusters) and the baseline LDA model (see Tables 8, 9 and 10 in the appendix). At first glance, the topic key words in Table 8 would be meaningful only to someone familiar with the MultiWOZ intents. To anyone with no knowledge of MultiWOZ, these key words lack the context required to interpret the topics, the context which can only be discovered when the same key words are used in sentences or utterances like in Table 9. For example, the highest scoring key word "train" in the largest baseline LDA topic versus the highest scoring key utterance "I need a train on thursday" in the largest ELDA topic, the key word "depart" versus the key utterance "What day and time would you like to depart", the key word "leave" versus the key utterance "I want to leave on Tuesday after 12:45", the key words "parking", "wifi", "free". versus the key utterance "I need free parking and free wifi though.", etc. show the power of utterances over words. As discussed before these key utterances are cluster labels and Table 10 provides a good idea about the quality of the clusters and validates the selection of their respective labels. Often in topic modeling evaluation, the reviewer must read the actual documents within the topics to better grasp what the topic is about, as the key words alone may not provide enough context. ELDA reduces this manual effort as the top utterances provide this context.

We ran both LDA and ELDA on an unseen, unlabeled technical helpdesk Accenture dataset (containing customer-agent dialogs resolving technical issues) with the optimal ELDA hyperparameters found for MultiWOZ and labeled the topics for both approaches. As expected, the topic key words were not descriptive enough to label the LDA topics and we had to manually review a few dialogs of each topic to understand what they were about. In contrast, the topic utterances provided the required context and meaning to understand and label the ELDA topics, with little to no need of reviewing the dialogs. For legal/privacy reasons we are not able to share these results.

6 Conclusions and future work

In this work we developed ELDA, an embedding-based LDA method, that represents each document or dialog in a dataset as a bag of utterance clusters instead of a bag of words. As a result, this approach represents each LDA topic as a probability distribution over utterance clusters which are labeled by the utterances closest to the cluster centroids. Unlike key words, the key utterances (cluster labels) provide more context to each topic, which helps to better interpret and label the topics. The ELDA and LDA approaches were evaluated and

compared using the MultiWOZ dataset. The results indicate ELDA is on par with the standard LDA in accurately identifying the existing topics or dialog intents, while producing easier-to-interpret topic descriptions that facilitate and accelerate the task of manually labeling the resulting topics.

The optimal ELDA hyperparameter values presented here may be extrapolable to other datasets, given that the semantic similarity distance magnitudes are given by the sentence embedding and not by the dataset. We continue testing ELDA with other (proprietary) datasets to verify this hypothesis.

One proposal for improving this work is to use a more stringent overlap metric in order to force the hyperparameter fine-tuning process to converge to better values. Note that the current approach considers a match between the predicted and true intents if any one of the intents match. Hence better hyperparameter values than the ones selected in this paper may be yet found.

7 Acknowledgements

We thank Paul A. Walsh, Ondřej Dušek and the SIGDIAL 2022 reviewers for their feedback.

References

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174, Brussels, Belgium. Association for Computational Linguistics.

Natalie Cygan. 2021. Sentence-BERT for interpretable topic modeling in Web browsing data. Technical Report CS224N, Department of Computer Science, Stanford University.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. 2020. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pages 226–231. AAAI Press.

Ting Han, Ximing Liu, Ryuichi Takanabu, Yixin Lian, Chongxuan Huang, Dazhen Wan, Wei Peng, and Minlie Huang. 2021. MultiWOZ 2.3: A multi-domain task-oriented dialogue dataset enhanced with annotation corrections and co-reference annotation. In *Natural Language Processing and Chinese Computing*, pages 206–218, Cham. Springer International Publishing.

Olzhas Kozbagarov, Rustam Mussabayev, and Nenad Mladenovic. 2021. A new sentence-based interpretative topic modeling and automatic topic labeling. *Symmetry*, 13(5).

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, page II–1188–II–1196. JMLR.org.

James B. MacQueen. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press.

Christopher E. Moody. 2016. Mixing dirichlet topic models and word embeddings to make lda2vec. *Computing Research Repository*, abs/1605.02019.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA.

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 and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2020. Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online. Association for Computational Linguistics.

A Appendix

A.1 Topic modeling results for baseline LDA and best ELDA models

Tables 8 and 9 list the three largest topics obtained from the baseline LDA and best ELDA models, respectively, along with the top 9 key items and their probability scores.

Topic 17		**Topic 12**		**Topic 2**	
Word	**Score**	**Word**	**Score**	**Word**	**Score**
train	0.045	hotel	0.032	hotel	0.024
leave	0.027	stay	0.020	guesthouse	0.022
arrive	0.022	guesthouse	0.020	parking	0.017
travel	0.021	parking	0.019	stay	0.016
ticket	0.021	night	0.019	free	0.016
depart	0.020	free	0.018	east	0.015
time	0.018	wifi	0.016	allenbell	0.015
cambridge	0.013	guest	0.016	north	0.013
departure	0.011	house	0.014	night	0.013

Table 8: Top 9 key words (with probability scores) for the three largest LDA topics

Topic	**Cluster**	**Cluster label**	**Score**
	25	**I need a train on thursday .**	**0.131**
	5	Train TR1526 leaves 17:40 and will get you there by 18:08 .	0.085
	14	I need to find a train leaving on Thursday going to Cambridge .	0.060
	27	I want to leave on tuesday after 12:45 .	0.048
6	38	What day and time would you like to depart ?	0.047
	31	From where will you be departing ?	0.034
	40	From Cambridge , which is why I asked the Cambridge TownInfo centre .	0.033
	43	Would you like me to book a reservation for it ?	0.027
	37	Its entrance fee is free .	0.027
	23	**I would like a guesthouse that is 4 stars .**	**0.150**
	41	I am looking for a hotel instead of a guesthouse .	0.063
	11	Is there a price range you 'd like ?	0.047
	39	I need to book it for 4 people starting from saturday for 5 nights .	0.046
37	30	I need free parking and free wifi though .	0.042
	3	Can I get some help finding a hotel or guesthouse please ?	0.041
	46	There are a couple of options .	0.034
	34	Is there a particular area of town you 'd like to be in ?	0.034
	36	Would you like any other info ?	0.030
	19	**I have your table booked for Tuesday at 15:15 .**	**0.146**
	8	I 'm looking for a moderately priced restaurant that serves chinese food .	0.095
	28	Is there a particular kind of restaurant you would like ?	0.065
	12	Your reference number is AJSQZY8R .	0.034
31	11	Is there a price range you 'd like ?	0.031
	6	It is in the centre part of town .	0.030
	42	The Booking was successful .	0.026
	22	I need the reference number please .	0.025
	10	I would like to book a reservation for it .	0.024

Table 9: Top 9 key utterances (with probability scores) for the three largest ELDA topics

A.2 Clustering results

Table 10 contains a sample of three clusters and some of their utterances. Each of these clusters is a top-scoring key item for each of the largest three topics from the best ELDA model (see the rows in bold in Table 9). To exhibit the quality of these clusters and represent them fairly, we take all the utterance embeddings within a given cluster, compute the distances to the cluster centroid, and rank them in ascending order. We display nine utterances in total, the first three are the three closest to the centroid, the next three are in the middle of the ranked list, and the last three are the three furthest from the centroid.

Cluster 25: I need a train on thursday .	**Cluster 23: I would like a guesthouse that is 4 stars .**	**Cluster 19: I have your table booked for Tuesday at 15:15 .**
Closest		
I need a train on thursday .	I would like a guesthouse that is 4 stars .	I have your table booked for Tuesday at 15:15 .
I need a train that gets me where I 'm going by 4:15 PM .	I am looking for a moderately priced hotel , that has a 4 star rating .	I would like to book a table for 6 at 15:15 on Tuesday .
I need a train that is leaving on wednesday .	I would prefer a 4 star hotel , are any of those three rated 4 stars ?	Please book a table for 7 at 15:15 on Wednesday .
Middle		
I have a number of trains leaving from london liverpool street .	yes it is 4 star	Can you book a table for seven people on Thursday at 15:00 ?
Actually yes , can you help me find a train to london liverpool street ?	Might you be willing to accept a place with 4 stars and free parking ?	Can you book me a table for 7 people on Sunday at 13:00 ?
Could I have the price for that train please ?	Yes , I would like to stay in the West area of town and I would also like it to have a 3 star rating .	Please book a table for 1 at 20:00 on friday .
Furthest		
The last train of the day will work for you .	Lucky star .	I am very sorry , our system was giving me an error , but I have managed to book your party of 5 at 16:45 on Tuesday .
There are 10 results of trains departing from Ely on Thursday .	It is four starts and it does have wifi .	You 'll find a table for 8 at Loch Fyne for 18:15 , reference number NGNNFSHD .
With your new criteria , that train wo n't work anymore , but there are other options .	The lucky star is chinese .	OK , a yellow Skoda will pick you up at the Cherry Hinton at 12:30 to get you to the restaurant in time for that 13:00 reservation .

Table 10: Sample of three utterance clusters (the highest scoring for each of the three largest topics from the best ELDA model) each with a sample of nine utterances.