Unsupervised dialogue intent detection via hierarchical topic model

Artem Popov[1,2], **Victor Bulatov**[1], **Darya Polyudova**[1], **Eugenia Veselova**[1]
[1]Moscow Institute of Physics and Technology
[2]Lomonosov Moscow State University
popov.artem.s@yandex.ru, viktor.bulatov@phystech.edu,
darya.polyudova@phystech.edu, veselova.er@phystech.edu

Abstract

One of the challenges during a task-oriented chatbot development is the scarce availability of the labeled training data. The best way of getting one is to ask the assessors to tag each dialogue according to its intent. Unfortunately, performing labeling without any provisional collection structure is difficult since the very notion of the intent is ill-defined.

In this paper, we propose a hierarchical multimodal regularized topic model to obtain a first approximation of the intent set. Our rationale for hierarchical models usage is their ability to take into account several degrees of the dialogues relevancy. We attempt to build a model that can distinguish between subject-based (e.g. medicine and transport topics) and action-based (e.g. filing of an application and tracking application status) similarities. In order to achieve this, we divide set of all features into several groups according to part-of-speech analysis. Various feature groups are treated differently on different hierarchy levels.

1 Introduction

One of the most important goals of task-oriented dialogue systems is to identify the user intention from the user utterances. State-of-the-art solutions like (Chen et al., 2017) require a lot of labeled data. User's utterances (one or several for a dialogue) have to be tagged by the intent of the dialogue.

This is a challenging task for a new dialogue collection because the set of all possible intents is unknown. Giving a provisional hierarchical collection structure to assessors could make the intent labeling challenge easier. The resulting labels will be more consistent and better suitable for model training.

Simple intent analysis is based on empirical rules, e.g. "question" intent contains phrase "what is # of #" (Yan et al., 2017). More universal and robust dialogue systems should work without any supervision or defined rules. Such systems can be implemented with automatic extraction of the semantic hierarchy from the query by multi-level clustering, based on different semantic frames (capability, location, characteristics etc.) in sentences (Chen et al., 2015). In our work intents represent a more complex entity which combine all intentions and objectives.

Many previous works take advantage of hierarchical structures in user intention analysis. In paper (Shepitsen et al., 2008) automatic approach through hierarchical clustering for document tagging is used. However, this approach does not take advantage of peculiar phrase features, such as syntax or specific words order. Syntactic parsing of intention was applied in (Gupta et al., 2018) to decompose client intent. This hierarchical representation is similar to a constituency syntax tree. It contains intentions and objects as tree elements and demands deep analysis of every sentence. Attempt to extract subintents along with main intent can be found in paper (Tang et al., 2018), but as proved below it is not necessary to apply neural networks for precise and efficient retrieval of multi-intent, especially in unsupervised task.

We propose a hierarchical multimodal regularized topic model as a simple and efficient solution for accurate approximation of the collection structure. The main contribution of this paper is the construction of a two-level hierarchical topic model using different features on the first and second levels. To the best of our knowledge, this is the first work that investigates that possibility. We

Proceedings of Recent Advances in Natural Language Processing, pages 932–938,
Varna, Bulgaria, Sep 2–4, 2019.
https://doi.org/10.26615/978-954-452-056-4_108

introduce a custom evaluation metric which measures the quality of hierarchical relations between topics and intent detection.

The hierarchy structure helps to make a provisional clustering more interpretative. Namely, we require first level topics to describe the dialogue subject and the second level topics to describe the action user is interested in. We accomplish this by incorporating information about part-of-speech (PoS) tags into the model.

This paper is organized as follows. Section two describes popular approaches to an unsupervised text classification. Section three describes our reasoning behind our choices of model architecture. Section four briefly reviews our preprocessing pipeline and introduces several enhancements to the existing NLP techniques. We demonstrate the results of our model in section five. We conclude our work in section six.

2 Text clustering approaches

2.1 Embeddings approaches

The simplest way to build a clustering model on a collection of text documents includes two steps. On the first step, each document is mapped to a real-valued vector. On the second step, one of the standard clustering algorithms is applied to the resulting vectors.

There are many methods to build an embedding of a document. The simplest way is the tf-idf representation. Logistic regression on the tf-idf representation is quite a strong algorithm for the text classification problem. This algorithm is respectable baseline even in deep neural networks research (Park et al., 2019). However, the direct use of the tf-idf representation leads to poor results in the clustering problem because of the curse of dimensionality. Dimensionality reduction methods could be used to improve clustering quality: PCA or Uniform Manifold Approximation and Projection (UMAP, McInnes et al. (2018)).

Another popular approach makes use of different word embeddings (Esposito et al., 2016). First of all, each word is mapped to a real-valued vector. Then the document representation is derived from the embeddings of its words. The most popular embedding models belong to the word2vec family (Mikolov et al., 2013b): CBOW, Skip-gram and their modifications (Mikolov et al. (2013a)). For correct representation word2vec models should be trained on a large collection of documents, for example, Wikipedia. Further improvement in quality of clustering models with embeddings can be achieved through fine-tuning. Similar to the tf-idf approach dimensionality reduction is often employed for the clustering problem (Park et al., 2019). Several averaging schemes can be used to aggregate word embeddings: mean, where all words contribute equally to the document, or idf-weighted, where rare words have a greater contribution than frequent words.

2.2 Topic modeling

Another approach to text clustering problem is topic modeling. The topic model simultaneously computes words and document embeddings and perform clusterization. It should be noted that in some cases topic model-based embeddings outperform traditional word embeddings, (Potapenko et al., 2017). The probability of the word w in the document d is represented by formula below:

$$p(w \mid d) = \sum_{t \in T} p(w \mid t) p(t \mid d) = \sum_{t \in T} \phi_{wt} \theta_{td}$$

where matrix Φ contains probabilities ϕ_{wt} of word w in topic t, matrix Θ contains probabilities θ_{td} of topic t in document d.

Probabilistic Latent Semantic Analysis (pLSA) (Hofmann, 2000) is the simplest topic model which describes words in documents by a mixture of hidden topics. The Φ and Θ distributions are obtained via maximization of the likelihood given probabilistic normalization and non-negativity constraints:

$$L(\Phi, \Theta) = \sum_{d \in D} \sum_{w \in W} n_{dw} \log p(w|d) \;\to\; \max_{\Phi, \Theta}$$

$$\sum_{w \in W} \phi_{wt} = 1, \;\; \phi_{wt} \geq 0$$

$$\sum_{t \in T} \theta_{td} = 1, \;\; \theta_{td} \geq 0$$

This optimization problem can be effectively solved via EM-algorithm or its online modifications (Kochedykov et al., 2017).

Latend Dirichlet Allocation (LDA) (Blei et al., 2003) model is an extension of pLSA with a prior estimation of the Φ and Θ, widely used in topic modelling. However, as a solution for both pLSA and LDA optimization problem is not unique, each solution may have different characteristics.

Additive Regularization of Topic Models (ARTM) (Vorontsov and Potapenko, 2015) is non-bayesian extension of likelihood optimization task, providing robustness of the solution by applying different regularizers. Each regularizer is used to pursue different solution characteristics. For example, many varieties of LDA can be obtained from ARTM model by using certain smoothing regularizer; pLSA model is an ARTM model without regularizers. Furthermore, documents can contain not only words but also terms of other modalities (e.g. authors, classes, n-grams), which allow us to select specific for our task language features. In this case, instead of a single Φ matrix, we have several Φ_m matrices for each modality m. Resulting functional to be optimized is the sum of weighted with α_m coefficients modalities likelihoods with regularization terms:

$$\sum_m \alpha_m L(\Phi_m, \Theta_m) + R(\cup_m \Phi_m, \Theta) \rightarrow \max_{\Phi,\Theta}$$

3 Multilevel clustering

Our goal is to build a topic model with topics corresponding to the user's intents. We use the following operational definition of intent: two dialogues (as represented by user's utterances) are said **to have the same intent** if both users would be satisfied with the essentially same reaction by the call centre operator. This definition, while inherently problematic, allows us to highlight several important practical problems:

- Simple bag-of-words (BoW) approach isn't sufficient. Compare: "I want my credit card to be blocked. What should I do¿" and "My credit card is blocked, what should I do¿".
- In some cases, the intent of conversation is not robust to a single word change. "I want to make an appointment with cardiologist" and "I want to make an appointment with neurologist" are considered to have the same intent since they require the user to perform a virtually identical set of actions. However, "Payment of state duty for a passport" and "Payment of state duty for vehicle" are vastly different.

To account for the BoW problem we add an n-gram modality and p_{tdw} smoothing regularizer (Skachkov and Vorontsov, 2018) for all tokens. The p_{tdw} smoothing regularizer respects the sequential nature of text, making the distributions $p(t|d,w)$ more stable for w belonging to a same local segment. In a way, $p(t|d,w)$ distribution could be interpreted as the analogue for context embeddings in topic modeling world. $p(t|d,w)$ distribution isn't used directly for topic representation, but it is used on the E-step of EM-algorithm for ϕ_{wt} and θ_{td} recalculation.

In order to obtain more control over intent robustness we propose to use a two-level hierarchical topic model. The first level is responsible for coarse-grained similarity, while the second one could take into account less obvious but important differences.

The hierarchical ARTM model consists of two different ARTM models for each level, which are linked to each other. The first level of the hierarchical model can be any ARTM model. The second level is built using regularizer from (Chirkova and Vorontsov, 2016) which ensures that each first-level topic is a convex sum of second-level topics. Various methods could be employed to ensure that each parent topic is connected to only a handful of relevant children topics: one can use either interlevel sparsing regularizer (Chirkova and Vorontsov, 2016) or remove "bad" edges according to EmbedSim metric (Belyy, 2018).

3.1 Distinct hierarchy levels

Building a two-level clustering model is a difficult task due to the inaccuracy of clustering algorithms. Provided that documents in the model first-level clusters are already similar to each other (as they should be), further separation could be complicated (especially if we attempt to subdivide each cluster by the same algorithm). In practice, the second-level clusters tend to repeat first-level clusters at smaller scale instead of demonstrating some meaningful differences. In order to make our model able to distinguish new dissimilarities in clusters on the second level, we adjust algorithm at the second level: in broad strokes, we base the second level of model on different features.

In the context of our problem, separation based on the functional purpose of the model tokens is proposed. We divide all words and n-grams into two groups based on the PoS analysis: "thematic" and "functional". The "functional" group consists of the verb words and n-grams that contain at least one verb. The "thematic" group consists of the nouns and adjectives and n-grams that contain at least one noun and have no verbs. Inspired by

multi–level (Tang et al., 2018) and multi–syntactic (Gupta et al., 2018) phrases annotation, among with hierarchical partition, our approach is essential for client goal and subgoals extraction.

The purpose of the first hierarchy level is to determine the conversation subject (the entities the dialogue is about). Hence, at the first level of the hierarchy thematic tokens should have a noticeably higher weight than functional tokens. The purpose of the second level of hierarchy, by contrast, is to determine client intent concerning particular objects (e.g. what action the client is trying to perform). Functional tokens should have higher impact over thematic ones. The tokens unrelated to these two groups are used on both levels and serve as a connection between the layers.

4 Preprocessing

We use standard preprossessing pipeline consisting of tokenization, lemmatization, part-of-speech tagging, n-grams extracting, named entity recognition and spell checking. In this section we describe some details of preprocessing algorithms since the preprocessing is very important for any morphologically rich languages such as Russian.

Data prepossessing pipeline consists of many parts, therefore each part must be relatively fast. That is why we don't use some great powerful approaches such as (Devlin et al., 2018) for NER.

4.1 N-grams extracting

Conventional approach in surpassing the bag-of-word hypothesis of the model is by adding n-grams or collocations into the model. To extract n-grams we use TopMine algorithm (El-Kishky et al., 2014) based on a words co-occurrences statistics.

However, we found it beneficial to implement some modifications. First change alters gathering and usage of word co-occurrences statistics: TopMine differentiates between sequences (w_1, w_2) and (w_2, w_1), which is not desirable for synthetic languages with less strict word order compared to English. To make it better suited to the Russian language, we use multisets as containers for collocations instead of sequences. Second change modifies the extraction process: while the original version of TopMine extracts only disjoint collocations and won't detect sub-collocations (e.g. if n-gram "support vector machines" is extracted, n-gram "support vector" will not be extracted), our modification will extract every high-scoring collocation at the cost of increased memory usage.

4.2 Named entity recognition

There are a lot of references to the speakers' names, company/product names, streets, cities in the dialogue collection. It makes sense to take into account some entities in a special way.

For the named entity recognition problem (NER) different methods are commonly used: rule-based, machine-learning-based or neural-networks-based. We used neural network from Arkhipov et al. (2017) pretrained on a PERSONS-1000 (Vlasova et al. (2014)) for our experiments. We replace all person related tokens by the $\langle \mathrm{PERSON} \rangle$ tag.

4.3 Spell checking

Errors and typos in client utterances are common in the dialogue collection. The simplest way to deal with this problem is to apply a spell checking algorithm. We use Jamspell[1] algorithm for spell checking since its fastness.

We make some modifications to adapt the Jamspell model to our case. First of all, the language model used to select the best correction candidates should be trained on the collection for clustering. This modification takes into account the collection specificity and collection specific words won't be treated as unknown.

Also the set of candidates can be extended. According to the statistics of Yandex search engine[2] word merging error is one of the most popular typos in the dialogues. Hence, we add candidates that are obtained via splitting a word in two.

5 Experiments

We use two dialogue datasets from the Russian call-centres ($\sim 90K$ dialogues in each) in our experiments. The first dataset is collected from client dialogues with various public services. The second dataset is conversation logs of ISP tech support. All dialogues are between a user and a call agent, mean length of a single dialogue is six utterances. Both datasets are proprietary.

5.1 Scoring metric

There are several approaches for measuring the quality of topic model, especially its interpretabil-

[1] Jamspell github

[2] Yandex search errors statistics (on Russian)

ity. The usual procedure involves evaluating the list of most frequent topic words by human experts. However, this approach suffers from several fundamental limitations (Alekseev, 2018). Therefore we choose to employ a different method.

For each dataset, we collect a set of dialogue pairs to score our model. Following the reasoning outlined in the section 3, we generated a number of (d_1, d_2) pairs (where d_i is a dialogue) and asked three human experts to label them. To measure the quality of the model, we compare these labels to the labels predicted by model.

The following list summarizes our approach for model estimation and labeling guidelines for human experts:

- `0`: d_1 and d_2 have nothing in common. Such objects should correspond to the different first-level topics.
- `1`: both d_1 and d_2 are related to the same subject, but there are significant differences. Such dialogues should correspond to the same first-level topic, but to the different second-level topics.
- `2`: d_1 shares an intent with d_2. Such dialogues should correspond to the same first-level and second-level topics.
- `?`: it is impossible to determine the intent for at least one of the dialogues.

We select the best model according to the accuracy metric on a given labeled pairs. Three sets of pairs are used for the estimation ($\sim 12K$ and $\sim 1.5K$ for the first dataset, $\sim 1.5K$ for the second dataset). All model hyperparameters are tuned according to the accuracy on a $12K$ dataset ("1-big"). Two other sets are used to control overfitting ("1-small" and "2-small"). Notably, the good performance on 2-small dataset implies that the model generalizes beyond the initial training dataset.

The same preprocessing procedures are used for both datasets. All tokens are lemmatized, stoptokens are deleted, simple entities (e-mails, websites e.t.c) are replaced by their tags. Operator utterances are deleted from the dialogue document (they are not informative in our datasets; for example, there are many cases where operator fails to reply at all). Finally, each document is a concatenation of one dialogue user utterances from a single dialogue.

5.2 Baselines

As one of the baselines, we use the following procedure. First, we convert raw texts into real-valued vectors using pretrained embeddings or tf-idf scores in a way described in 2.1. Second, we cluster this dataset via K-Means algorithm. Third, we treat each cluster as a separate collection and perform K-Means algorithm again. As a result, we obtain both first-level and second-level clusters.

Another baseline models are hierarchical topic model without any additional regularizers and hierarchical topic model with Φ and Θ smoothing for both levels. For K-Means based algorithms we tune embeddings dimensionality and both level cluster number. For topic modeling based algorithms we tune both level topics number. As shown in table 1 regularized topic model outperforms K-Means approaches at two out of the three pair sets.

	1-big	1-small	2-small
hKmeans (tf-idf)	0.568	0.593	**0.649**
hKmeans (emb.)	0.615	0.638	0.641
hPLSA	0.603	0.675	0.633
hARTM	**0.636**	**0.683**	0.631

Table 1: Baselines accuracy

5.3 Proposed model perfomance

We use several NLP-based techniques described in 4 to improve main model quality. We start with the hPLSA model. For each problem we test a few approaches and choose the best one. We add all main features one by one, e.g. we choose the best method for extracting n-grams and use it on the next step. We conduct all the experiments in the following order:

1. including additional n-gram modality, choosing between the based and modified n-grams extracting methods, tuning modality weights and topics number;
2. adding p_{tdw} smoothing at the first model level for all tokens, tuning regularizer coefficient and topics number;
3. replacing person related named entities, choosing between the dictionary-based and rnn-based methods;
4. typo correction, choosing between the base and modified algorithm

	1-big	1-small	2-small
hPLSA	0.603	0.675	0.633
+ n-grams base	0.612	0.634	0.633
+ n-grams mod.	**0.635**	**0.674**	**0.655**
+ ptdw smooth.	**0.64**	**0.678**	**0.66**
+ NER dict.	0.634	0.661	0.635
+ NER NN	**0.64**	**0.68**	**0.662**
+ Jamspell	0.635	0.674	0.655
+ mod. Jamspell	**0.657**	**0.686**	**0.663**

Table 2: NLP techniques quality improvement

As the table 2 demonstrates our n-grams extraction method outperforms traditional TopMine algorithm in this task. Replacing persons by a tag does not lead to a great improvement of the quality. Our analysis of hPLSA cluster top-tokens shows that only 3% of the top-tokens are related to persons. After the NER preprocessing the proportion of named entities in top tokens reduces to 0.3%. And at the same time spellchecking improves the performance on all three pair sets. It should be noted that standard Jamspell algorithm leads to a quality decrease.

Finally, we apply feature grouping schemes proposed in 3.1. The results (table 3) turned out to be reassuring. There is a noticeable performance boost for all of the pair sets.

	1-big	1-small	2-small
featured hARTM	0.657	0.686	0.663
+ groups	**0.667**	**0.715**	**0.672**

Table 3: Grouping feature quality improvement

Further, we represent some examples of the model performance. All example texts from examples were translated from Russian to English. In the table 4 all subtopics of the topic "Tariff plan" are presented. Each subtopic described by the characteristic question.

In the table 5 we demonstrate top documents corresponding to the "How do I switch from credit to advance payment?" subtopic.

6 Conclusion

In this paper, we report a success in formalizing the clustering process suitable for unsupervised inference of user intents.

The realization that any intent consists of two crucial parts: the entity relevant to the user's request and the action user wishes to perform helped us to choose a two-level hierarchical model as our main tool. This leads us to design a custom quality metric which takes into account several degrees of the dialogues relevancy.

Our next step was to devise a PoS-based feature separation and to leverage n-grams, named entities and spellchecking. This allowed us to construct a hierarchical multimodal regularized topic model which outperforms all baseline models.

Tariff plan
How to change the tariff plan?
When did the tariff change happen?
How often can I change my tariff plan?
When will the changes take effect when the tariff is changed?
Why can't I change the tariff?
Why was the tariff plan changed without my knowledge?
Why there are no available tariff plans for the transition?

Table 4: Subtopics of topic "Tariff plan"

How do I switch from credit to advance payment?
How do I switch from credit to advance payment?
Hi. Tell me can we change the credit system of payment to advance? Well thanks!
I need to change my payment from credit to advance.
How to disable credit payment system?
Hello. Change the payment system from credit to advance!
Good morning. How to change the payment system from credit to normal?
Disable the credit payment system.

Table 5: Top documents of subtopic "How do I switch from credit to advance payment?"

Acknowledgments

We thank our colleagues Alexey Goncharov and Konstantin Vorontsov from Machine Intelligence Laboratory who provided expertise that greatly assisted the research. We thank Evgeny Egorov for comments that greatly improved the manuscript.

References

V.A. Bulatov V.G. Vorontsov K.V. Alekseev. 2018. Intra-text coherence as a measure of topic models' interpretability. In *Computational Linguistics and Intellectual Technologies: Papers from the Annual International Conference Dialogue*. pages 1–13.

Mikhail Y Arkhipov, Mikhail S Burtsev, et al. 2017. Application of a hybrid bi-lstm-crf model to the task of russian named entity recognition. In *Conference on Artificial Intelligence and Natural Language*. Springer, pages 91–103.

A.V. Seleznova M.S. Sholokhov A.K. Vorontsov K.V. Belyy. 2018. Quality evaluation and improvement for hierarchical topic modeling. In *Computational Linguistics and Intellectual Technologies: Papers from the Annual International Conference Dialogue*. pages 110–123.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3(Jan):993–1022.

Yun-Nung Chen, William Yang Wang, and Alexander I Rudnicky. 2015. Learning semantic hierarchy with distributed representations for unsupervised spoken language understanding. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Zheqian Chen, Rongqin Yang, Zhou Zhao, Deng Cai, and Xiaofei He. 2017. Dialogue act recognition via crf-attentive structured network. *CoRR* abs/1711.05568.

Nadezhda Chirkova and Konstantin Vorontsov. 2016. Additive regularization for hierarchical multimodal topic modeling. *Journal Machine Learning and Data Analysis* 2(2):187–200.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR* abs/1810.04805.

Ahmed El-Kishky, Yanglei Song, Chi Wang, Clare R Voss, and Jiawei Han. 2014. Scalable topical phrase mining from text corpora. *Proceedings of the VLDB Endowment* 8(3):305–316.

Fabrizio Esposito, Anna Corazza, and Francesco Cutugno. 2016. Topic modelling with word embeddings. In *Proceedings of the Third Italian Conference on Computational Linguistics CLiC-it 2016)*. pages 129–134.

Sonal Gupta, Rushin Shah, Mrinal Mohit, Anuj Kumar, and Mike Lewis. 2018. Semantic parsing for task oriented dialog using hierarchical representations. In *EMNLP*.

Thomas Hofmann. 2000. Learning the similarity of documents: An information-geometric approach to document retrieval and categorization pages 914–920.

Denis Kochedykov, Murat Apishev, Lev Golitsyn, and Konstantin Vorontsov. 2017. Fast and modular regularized topic modelling. In *2017 21st Conference of Open Innovations Association (FRUCT)*. IEEE, pages 182–193.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426* .

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* .

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. pages 3111–3119.

Jinuk Park, Chanhee Park, Jeongwoo Kim, Minsoo Cho, and Sanghyun Park. 2019. Adc: Advanced document clustering using contextualized representations. *Expert Systems with Applications* .

Anna Potapenko, Artem Popov, and Konstantin Vorontsov. 2017. Interpretable probabilistic embeddings: bridging the gap between topic models and neural networks. In *Conference on Artificial Intelligence and Natural Language*. Springer, pages 167–180.

Andriy Shepitsen, Jonathan Gemmell, Bamshad Mobasher, and Robin Burke. 2008. Personalized recommendation in social tagging systems using hierarchical clustering. In *Proceedings of the 2008 ACM conference on Recommender systems*. ACM, pages 259–266.

Nikolay Skachkov and Konstantin Vorontsov. 2018. Improving topic models with segmental structure of texts. In *Computational Linguistics and Intellectual Technologies: Papers from the Annual International Conference Dialogue*. pages 652–661.

Da Tang, Xiujun Li, Jianfeng Gao, Chong Wang, Lihong Li, and Tony Jebara. 2018. Subgoal discovery for hierarchical dialogue policy learning. In *EMNLP*.

Nataliya Vlasova, Elena Syleymanova, and Igor Trofimov. 2014. The russian language collection for the named-entity recognition task. *Language semantics: models and technologies* pages 36–40.

Konstantin Vorontsov and Anna Potapenko. 2015. Additive regularization of topic models. *Machine Learning* 101(1-3):303–323.

Zhao Yan, Nan Duan, Peng Chen, Ming Zhou, Jianshe Zhou, and Zhoujun Li. 2017. Building task-oriented dialogue systems for online shopping. In *Thirty-First AAAI Conference on Artificial Intelligence*.