



Московский государственный университет имени М. В. Ломоносова

Факультет вычислительной математики и кибернетики

Кафедра математических методов прогнозирования

## Построение обобщенного графа диалога

### КУРСОВАЯ РАБОТА

**Выполнил:**

студент 3 курса 317 группы

*Штыков Павел Дмитриевич*

**Научный руководитель:**

д.ф.-м.н., профессор

*А. Г. Дьяконов*

Москва, 2022

# Содержание

<b>1</b>	<b>Введение</b>	<b>2</b>
<b>2</b>	<b>Существующие подходы</b>	<b>3</b>
2.1	Мотивация . . . . .	3
2.2	VRNN и SVRNN . . . . .	3
2.3	TSCAN . . . . .	4
<b>3</b>	<b>Постановка задачи</b>	<b>4</b>
<b>4</b>	<b>Предложенный метод</b>	<b>6</b>
4.1	Алгоритм построения графа диалога в пространстве эмбедингов . . . . .	6
<b>5</b>	<b>Эксперименты</b>	<b>8</b>
5.1	Датасеты . . . . .	8
5.2	Оценка качества кластеризации . . . . .	9
5.3	Визуализация графов . . . . .	11
<b>6</b>	<b>Заключение и будущая работа</b>	<b>13</b>
<b>A</b>	<b>Приложение</b>	<b>15</b>

# 1 Введение

Обработка естественного языка (*NLP*) является ключевой задачей в машинном обучении, а обработка диалогов является важной ее частью. Одной из слабо изученных областей в обработке диалогов является проблема построения и представления общей структуры диалога.

Естественно предположить, что у диалогов из одной области может быть некоторая общая структура. Так же естественно представлять эту структуру в виде графа. Такой граф позволяет представить информацию о корпусе однородных диалогов в сжатой форме, подходящей как для визуализации, так и для встраивания в более сложные диалоговые системы. В данной работе предложена формализация понятия обобщенного графа диалога и базовые способы его построения и визуализации.

## 2 Существующие подходы

### 2.1 Мотивация

Базовая идея построения графа диалога состоит в поиске некоторой общей структуры диалогов в однородной выборке. Например, известно, что у операторов в колл-центрах есть четкий план вопросов и ответов, которому необходимо следовать. Соответственно большинство их диалогов похожи друг на друга, т.е. имеют общую структуру. Такую структуру чаще всего представляют в виде некоторого ориентированного графа, вершины которого отражают текущую тему диалога, а ребра — канву диалога. Следовательно общая цель исследований в данной области — построение такого графа и его дальнейшее изучение.

Однако на сегодняшний день количество актуальных работ по данной теме мало. Приведем некоторые из них.

### 2.2 VRNN и SVRNN

Самой цитируемой является серия статей [14], [10], в которой предлагается два способа построения графа диалога. Первый основан на использовании вариационных автокодировщиков в паре с рекуррентной нейронной сетью [4]. Во второй статье авторы добавили в данную архитектуру механизм внимания [17], что позволило им улучшить качество. Результат работы их алгоритма представлен на Рис. 1.

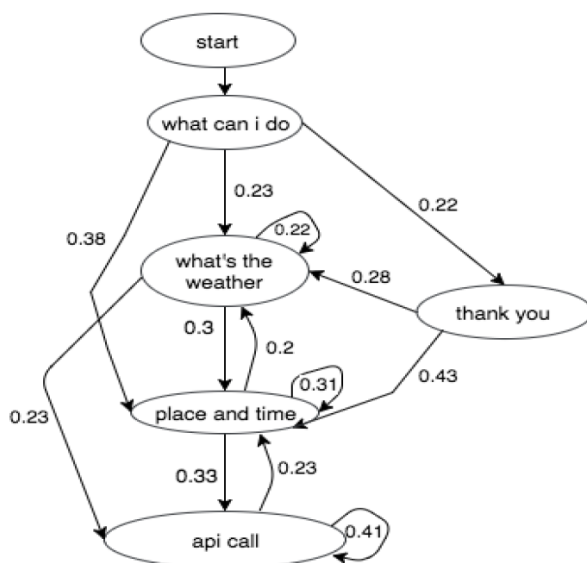


Рис. 1: Пример обобщенного графа диалога взятого из статьи [14]

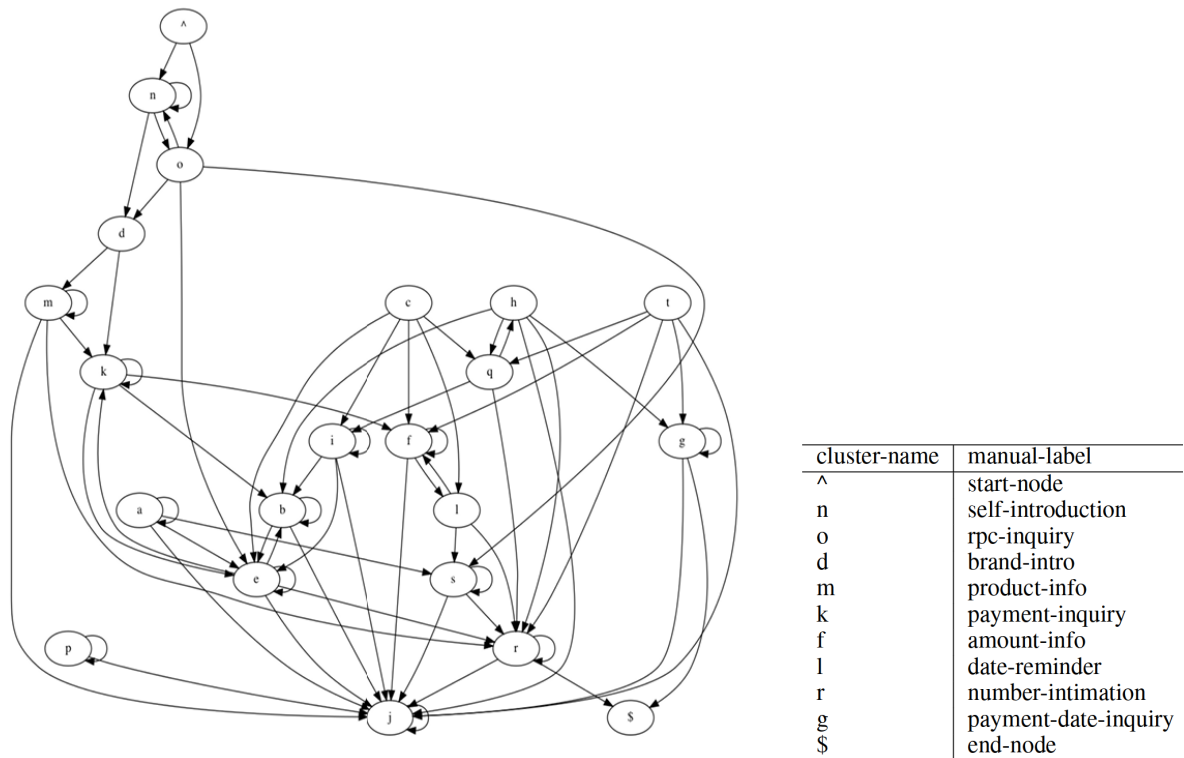


Рис. 2: Пример обобщенного графа диалога взятого из статьи [9]

## 2.3 TSCAN

В своей же работе мы будем ориентироваться на следующую статью [9]. В ней авторы применяют для построения графа диалога алгоритм классификации изображений с самообучением — SCAN [16]. Авторы адаптировали данный алгоритм для работы с текстами, используя в качестве эмбединга (для pretext задачи) стандартный BERT [6]. В данной работе будет исследована применимость эмбединга, более подходящего для семантической кластеризации — SBERT [11]. Граф полученный авторами представлен на рис. 2. Однако в своей работе авторы используют закрытый датасет для сравнения алгоритма с более простыми методами (в частности k-средних). Кроме этого авторы не предоставляют ни подробного алгоритма, ни его программного кода. В данной работе будет приведен пример такого алгоритма и проведены эксперименты с разными методами кластеризации. Также в работе дополнено определение обобщенного графа диалога для его более простого дальнейшего анализа и визуализации.

## 3 Постановка задачи

Введем определение обобщенного графа диалога.

**Определение 1.** Назовем обобщенным графом диалога пару  $T = (G, p(u|v))$ , где:

- $G = (V, E)$  — направленный взвешенный граф. С каждым ребром в графе  $G$  ассоциирована вероятность перехода по этому ребру:  $e_{i,j} \sim p(v_i|v_j)$ . При этом сумма вероятностей ребер выходящих из каждой вершины равна 1:  $\sum_i p(v_i|v_j) = 1$
- $u \in U$  — единичное высказывание, а  $U$  — пространство всех высказываний во всех диалогах;
- $p(u|v)$  — плотность вероятности (либо функция вероятности в случае дискретного пространства  $U$ ) отнесения высказывания  $u$  к текущей вершине  $v$ .

Такое определение не ограничивает нас в выборе модели для его построения. Дополнительное требование наличия функции  $p(u|v)$  позволит нам вычислять статистики полезные для визуализации и дальнейшего использования графа (например самое вероятное предложение или самые частотные слова среди предложений ассоциированных с текущей вершиной).

Также такой граф достаточно просто обобщается на случай персонализированных диалогов (например диалогов вида: «пользователь»-«система») — введением раскраски вершин, т.е. дополнительной функции  $\phi(v)$ , ставящей в соответствие каждой вершине некоторый персональный идентификатор пользователя ( $ID$ ). Однако для большей корректности необходимо ввести дополнительные ограничения. Так как высказывания пользователей чередуются, то логично потребовать, чтобы вершины разных цветов не были инцидентны. Заметим также, что основное определение не запрещает петли. В персонализированном графе их стоит запретить. В данной работе мы будем строить простой граф диалога, без дополнительной раскраски.

Пусть  $D = \{d_1, d_2, \dots, d_{|D|}\}$  — выборка диалогов, где  $d_i$  — один диалог. Каждый диалог является набором из нескольких высказываний:  $d_i = \{d_i^1, d_i^2, \dots, d_i^n\}$ ,  $d_i^j \in U$ . В данной работе мы будем работать с неразмеченными корпусами диалогов. В общем же случае нет ограничения на использование разметки.

Добавим к каждому диалогу  $d_i$  технические высказывания *начала* и *конца* диалога ( $BEGIN$  и  $END$ ). Аналогичные вершины добавим и в граф диалога. Вероятность  $p(u|v)$  для этих вершин будет вырождена в соответствующих точках в пространстве высказываний  $U$ . Это необходимо для более ясной конструкции графа и соблюдения ограничения на сумму

вероятностей ребер, исходящих из вершины:  $\sum_i p(v_i|v_j) = 1$ .

## 4 Предложенный метод

Для решения pretext задачи воспользуемся эмбедингом:

$$Embedding : U \rightarrow M = \mathbb{R}^n, \quad \text{для некоторого } n \in \mathbb{N}$$

В качестве эмбединга мы использовали предобученную сиамскую нейронную сеть [11] с разными базовыми сетями (подробнее в разделе 5). В дальнейшем, если не оговорено другого, под высказыванием  $u$  мы будем подразумевать его эмбединг:  $Embedding(u)$ .

Пространство  $M$  равно либо  $\mathbb{R}^{768}$ , либо  $\mathbb{R}^{384}$  в зависимости от использованной базовой сети в SBERT. В пространстве  $M$  введена косинусная мера, отражающая семантическую близость высказываний. Это позволяет использовать в пространстве  $M$  простые методы кластеризации для объединения близких по смыслу высказываний.

### 4.1 Алгоритм построения графа диалога в пространстве эмбедингов

Теперь приведем алгоритм построения обобщенного графа диалога  $T = (G, p(u|v))$ , для высказываний в пространстве эмбедингов  $M$ .

► Пусть выбран некоторый алгоритм кластеризации  $a$ :

$$a : M \rightarrow V$$

В данном случае множество вершин графа  $V$  есть множество кластеров. Соответственно может быть вычислена *дискретная* вероятность принадлежности каждого высказывания к каждой вершине:

$$a(u) = p(v|u)$$

При этом кластеризация может быть как жесткой (например методом  $k$ -средних ( $k$ -means)), так и мягкой (например смесью гауссиан ( $GMM$ )).

Зная  $p(v|u)$ , можно вычислить  $p(u|v)$ , используя теорему Байеса:

$$p(u|v) = \frac{p(v|u)p(u)}{\sum_{i=1}^{|U|} p(v|u_i)p(u_i)},$$

где  $p(u)$  частота встречаемости высказывания  $u$  во всем корпусе диалогов. Заметим, что вероятность  $p(u)$  не одинакова для всех высказываний, так как в корпусе могут встречаться диалоги с повторяющимися высказываниями.

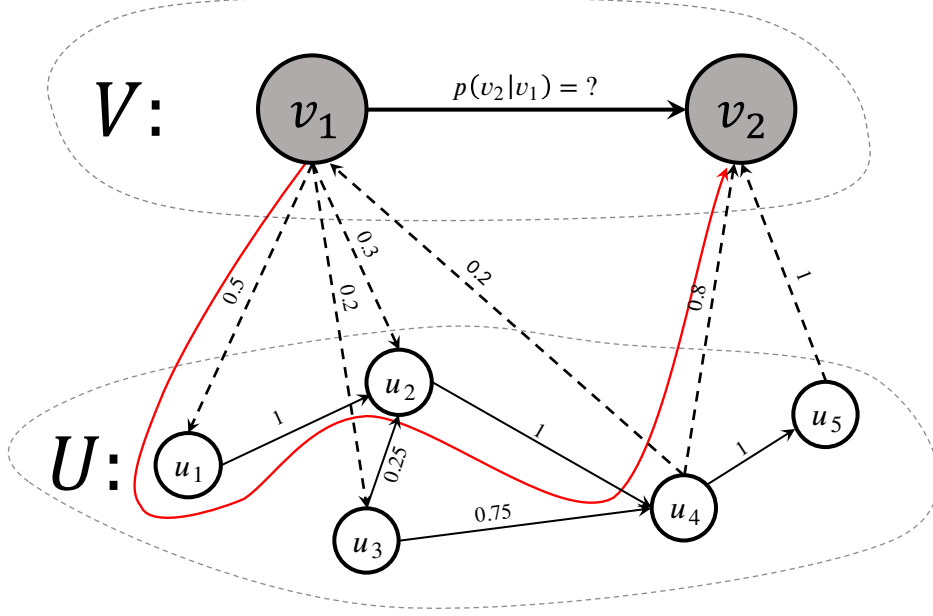


Рис. 3: Пример двух графов:  $G$  (сверху) в пространстве вершин  $V$  и  $\hat{G}$  (снизу) в пространстве высказываний  $U$ .

Нам осталось построить в графе  $G$  ребра и найти вероятности, ассоциированные с ними. Введем в пространстве высказываний  $U$  граф  $\hat{G}$ , подобный графу основному  $G$ , т.е. ориентированный взвешенный граф с вероятностями, ассоциированными с ребрами:

$$\hat{G} = (\hat{V}, \hat{E}), \quad \hat{V} \subset M, \quad \hat{E} \subset \hat{V} \times \hat{V}, \quad \hat{e}_{i,j} \sim p(u_j|u_i).$$

Данный граф строится напрямую по выборке диалогов, и ребра в нем имеют смысл апостериорной вероятности встретить ответ  $u_j$  на высказывание  $u_i$ . Схему совместного размещения обоих графов  $G$  и  $\hat{G}$  можно увидеть на рис. 3. Соответственно матрица смежности  $\hat{A}$  графа  $\hat{G}$  определяется как:

$$\hat{A} = (\hat{a}_{ij}), \quad \hat{a}_{ij} = p(u_j|u_i), \quad i, j = \overline{1, |U|}$$

Теперь, зная вероятности  $p(u|v)$ ,  $p(u|u)$  и  $p(v|u)$ , становится просто вычислить вероятности ребер  $p(v|v)$  в графе  $G$ :

$$p(v_j|v_i) = \sum_{\alpha, \beta}^{|U|} p(u_\alpha|v_i) p(u_\beta|u_\alpha) p(v_j|u_\beta)$$



То есть вероятность перехода из вершины  $v_i$  в вершину  $v_j$  графа  $G$  равна сумме вероятностей всех путей из  $v_i$  в  $v_j$ , проходящих через всевозможные пары высказываний  $(u_\alpha, u_\beta)$  в графе  $\hat{G}$ . Пример такого пути выделен на рис. 3 красной кривой.

Соответственно матрица смежности графа  $G$  имеет вид:

$$A = (a_{ij}), \quad a_{ij} = p(v_j|v_i), \quad i, j = \overline{1, |V|}$$

Так как в нашем случае совместные распределения  $p(u|v)$  и  $p(v|u)$  дискретны, то они могут быть представлены в виде матриц. Следовательно способ вычисления матрицы смежности  $A$  может быть представлен в короткой матричной форме:

$$A = p(u|v) \cdot \hat{A} \cdot p(v|u).$$

Мы закончили построение обобщенного графа диалога  $T$ . ■

Заметим, что данный алгоритм построения обобщенного графа применим не только в случае использования кластеризации в пространстве эмбеддингов, но и в случае использования любого другого алгоритма способного оценить апостериорные вероятности  $p(v|u)$  (например с помощью латентного размещения Дирихле (*LDA*) или нейронной сети, решающей задачу от начала до конца без промежуточного использования эмбеддингов). Кластеризация была выбрана, как наиболее простой метод.

## 5 Эксперименты

### 5.1 Датасеты

Нам не известен датасет полностью соответствующий поставленной задаче, т. е. датасет, состоящий из диалогов, имеющих некоторую общую *известную* структуру. Поэтому для проведения экспериментов нами было выбрано два стандартных датасета, для которых, на наш взгляд, можно предположить наличие общей структуры. Первый датасет — *Customer Support on Twitter* [1], в котором собраны ответы официальных аккаунтов технической поддержки крупных компаний. Мы выбрали из него подмножество сообщений аккаунтов шести разных авиакомпаний США, чтобы сделать датасет более однородным. Вторым датасетом — *DailyDialog* [7], в котором собраны обычные диалоги из повседневной жизни на разные темы. Для экспериментов мы взяли диалоги на тему работы, как наиболее однородные и узкие. Датасеты имеют размер 8081 диалога и 1924 диалога для Customer Support

on Twitter и DailyDialog соответственно. Среднее количество высказываний в одном диалоге равно 3.6 и 7.5 соответственно. Примеры диалогов из обоих датасетов приведены на рис. 4.

Twitter Customer Support	DailyDialog
— @AlaskaAir it says you open at 5:15 @317258 where is everyone? helloooooo <a href="https://t.co/WePfUANLsZ">https://t.co/WePfUANLsZ</a>	— Everything's gone wrong.
— @429415 @317258 Ticket counter opens at 615 is what I see on our website.	— I know, it's not as I had planned.
— @AlaskaAir @317258 all good! They just showed up thanks Andre	— What are we going to do now?
— @429415 That is good news	— I'll speak to Bob, he'll be able to help us

Рис. 4: Примеры диалогов из датасетов Twitter Customer Support и DailyDialog

Видно, что датасет Twitter Customer Support очень зашумлен. Так как переписка в Twitter публична, то в нее часто вмешиваются третьи лица. Следовательно мы оставили в корпусе только те диалоги между компаниями и пользователями, которые соответствуют схеме: «система» – «пользователь  $N$ » – «система» – «пользователь  $N$ » и т.д. Из высказываний были убраны все идентификаторы пользователей, а идентификаторы компаний были заменены на единый токен *companyname*. После чего была применена стандартная предобработка текста: приведение к нижнему регистру, удаление слов с цифрами, лемматизация, выбрасывание стоп-слов и т.д.

Датасет DailyDialog имеет более хорошее изначальное качество текста, поэтому к нему была применена только стандартная предобработка.

## 5.2 Оценка качества кластеризации

Для начала рассмотрим визуализацию пространства эмбедингов (в качестве эмбединга использовался SBERT с базовой сетью Distill-RoBERTa[8]). Мы понизили размерность с помощью t-SNE с перплексией равной 50. Результат представлен на рис. 6 и 5. Для диалогов из обоих датасетов заметна кластерная структура. Это дает нам некоторое подтверждение наличия общей структуры у корпуса диалогов. Однако кластеры небольших размеров и между ними много шума, это может привести к отсутствию четких границ между вершинами графа.

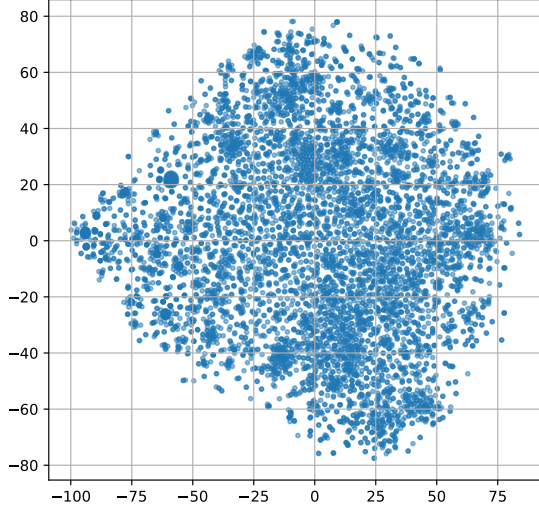


Рис. 5: Пространство эмбедингов для датасета DailyDialog

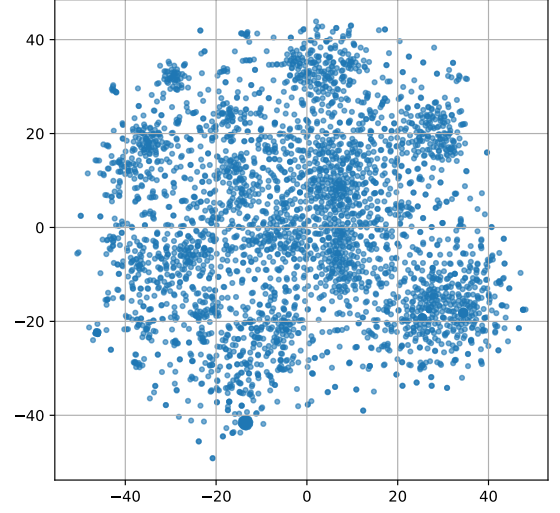


Рис. 6: Пространство эмбедингов для датасета Twitter Customer Support

		Twitter Customer Support				DailyDialog			
Model		Silh.	C.-H.	D.-B.	Entr.	Silh.	C.-H.	D.-B.	Entr.
$ V =5$	MPNet_KMeans	<b>0.052</b>	1043.9	3.912	0.535	0.011	<b>292.3</b>	5.166	0.712
	RoBERTa_KMeans	0.043	1003.3	4.43	0.606	<b>0.024</b>	281.8	5.176	0.719
	MiniLM_KMeans	0.045	<b>1054.0</b>	3.869	0.523	0.023	286.5	5.287	0.724
	MPNet_GMM	0.036	940.4	4.152	0.524	-0.001	253.3	5.517	0.694
	RoBERTa_GMM	0.034	988.7	4.544	0.617	0.022	278.1	5.156	0.71
	MiniLM_GMM	0.044	1046.8	<b>3.856</b>	<b>0.519</b>	0.01	273.4	<b>4.74</b>	<b>0.663</b>
	MPNet_SCAN	0.042	1117.515	4.169	0.576	0.017	230.752	5.992	0.717
	RoBERTa_SCAN	0.037	947.797	4.569	0.625	0.024	258.829	5.563	0.718
	MiniLM_SCAN	0.036	901.678	4.493	0.624	0.021	247.22	5.596	0.705

Таблица 1: Результаты сравнения качества кластеризации для графа с 5-ю вершинами для двух датасетов: Twitter Customer Support и DailyDialog

Измерим качество кластеризации. В таблице 1 переставлены результаты работы алгоритма с 5-ю вершинами в графе в зависимости от следующих параметров:

- Базовая модель в SBERT: *MPNet* [15], *DistillRoBERTa* [8], [13] и *MiniLM* [18]
- Кластеризатор: *k*-средних (*k-means*), смесь гауссиан (*GMM*) и SCAN (метод предложенный в статье 2)

Так как авторы статьи 2 не указали оптимальные гиперпараметры для SCAN, то он обу-

чался для каждой конфигурации с нуля со следующими стандартными гиперпараметрами:

- Количество голов классификатора: 1
- Темп обучения:  $1e^{-5}$
- Количество эпох: 15

Для графов с 10-ю и 15-ю вершинами в приложении А прикреплена аналогичная сравнительная таблица.

В качестве метрик использовались следующие базовые метрики качества кластеризации на неразмеченных данных: коэффициент силуэта (*Silh.*) [12], индекс Калински-Харабаса (*C.-H.*) [3] и индекс Дэвиса-Болдина (*D.-B.*) [5].

Также введем дополнительную метрику качества для структуры графа. Нам хотелось бы, чтобы граф был более определенным, т.е. было больше ребер с большой вероятностью и меньше ребер с меньшей вероятностью. Для этого будем измерять среднюю нормализованную энтропию (*Entr.*):

$$H(G) = \frac{1}{|V|} \sum_{i=1}^{|V|} \frac{\sum_{j=1}^{|V|} p(v_j|v_i) \ln p(v_j|v_i)}{\log |V|}$$

Соответственно, чем меньше энтропия, тем более определенный граф.

Из таблицы 1 видно, что мы не смогли повторить результат авторов статьи TSCAN 2. В нашем случае SCAN кластеризатор не смог превзойти стандартные k-средних и смесь гауссиан ни по одной метрике. Вероятно, мы не смогли подобрать необходимые гиперпараметры.

### 5.3 Визуализация графов

Наконец построим и визуализируем графы. Для всех графов использовалась лучшая (по результатам 1) модель для данного количества вершин и для данного датасета. В качестве маркировки вершин будем использовать 4-е слова с самым большим значением Tf-Idf. Tf-Idf представления строились для двухсот наиболее вероятных для данной вершины высказываний. Чтобы не засорять рисунок значениями вероятности ребер, мы заменили их разной толщиной ребер: чем толще ребро, тем больше вероятность перехода по нему. Также были убраны ребра с вероятностью меньше 0.1:  $p(u_j|u_i) < 0.1$ . Визуализация графов

производилась с помощью пакета GraphViz [2].

На рис. 7 и 8 представлены графы с 5-ю вершинами, составленные по обоим датасетам. Графы для 10-и и 15-и вершин находятся в приложении А.

В целом графы с 5-ю вершинами выглядят читабельными. Ребра с разной толщиной позволяют визуальную оценить наиболее вероятную канву диалога. Хотя Tf-Idf представление и является простым инструментом маркировки вершин, они позволяют приблизительно понять тему вершины. Графы же с большим количеством вершин становятся почти полностью связными. Никакой общей канвы диалога в них не наблюдается. Интерпретировать такой граф крайне сложно.

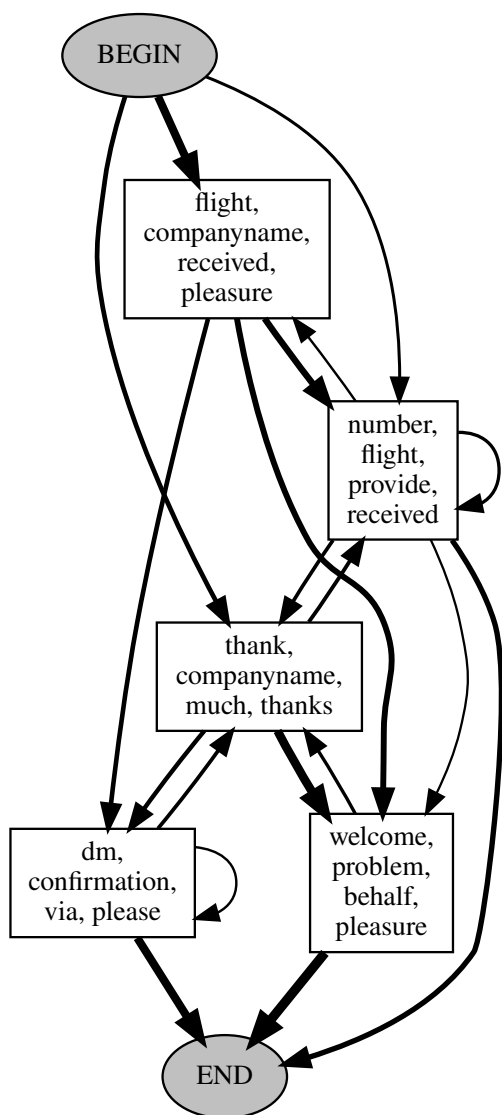


Рис. 7: Граф диалога с 5-ю вершинами для датасета Twitter Customer Support

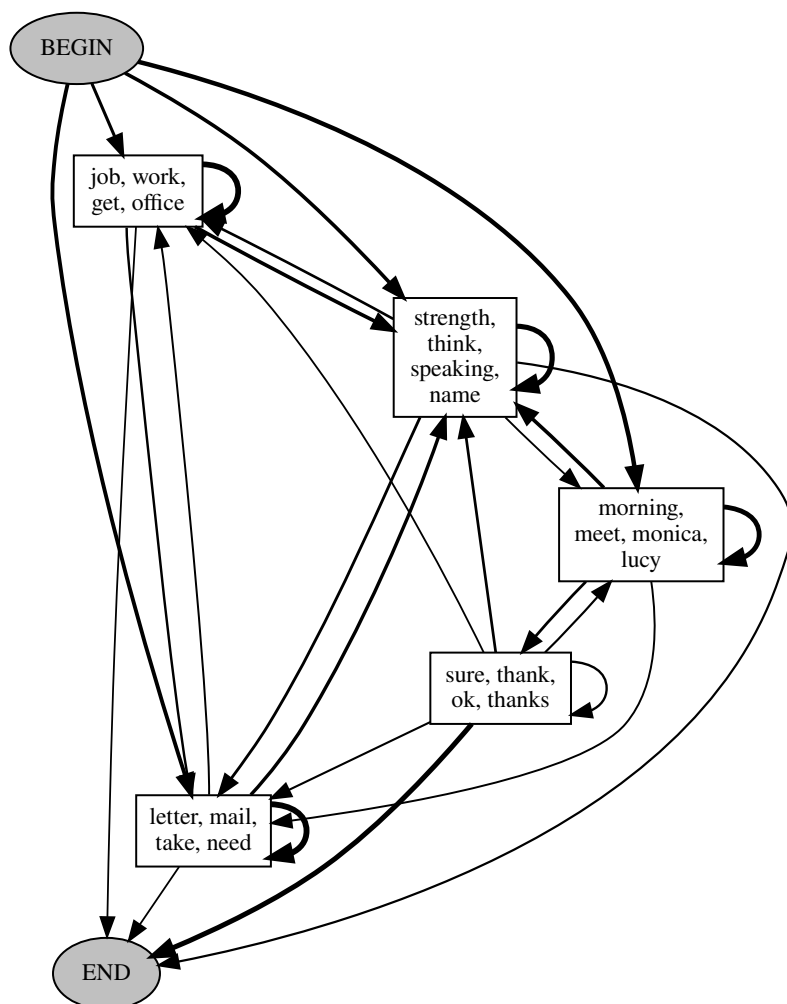


Рис. 8: Граф диалога с 5-ю вершинами для датасета DailyDialog

## 6 Заключение и будущая работа

В данной работе был приведен простой алгоритм построения графа диалога с помощью кластеризации в пространстве эмбедингов SBERT. Были проведены сравнения простых методов кластеризации со SCAN. Также предложено уточнение определения самого графа диалога.

Однако остается большое количество открытых вопросов:

- Отсутствует подходящий датасет, собранный для данной задачи;
- С ростом количества вершин граф быстро становится почти полносвязным, что усложняет его читаемость;
- Нет общепринятого принципа маркировки вершин и в целом визуализации графа.

## Список литературы

- [1] Customer support on twitter. <https://www.kaggle.com/datasets/thoughtvector/customer-support-on-twitter>. Доступ: 05.05.2022.
- [2] Graphviz: open source graph visualization software. <https://graphviz.org/>. Доступ: 07.05.2022.
- [3] Tadeusz Caliński and Harabasz JA. A dendrite method for cluster analysis. *Communications in Statistics - Theory and Methods*, 3:1–27, 01 1974.
- [4] Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron C. Courville, and Yoshua Bengio. A recurrent latent variable model for sequential data. *CoRR*, abs/1506.02216, 2015.
- [5] David L. Davies and Donald W. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227, 1979.
- [6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. 2018.
- [7] Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. DailyDialog: A manually labelled multi-turn dialogue dataset. pages 986–995, November 2017.
- [8] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019.
- [9] Apurba Nath and Aayush Kubba. TSCAN : Dialog structure discovery using SCAN. *CoRR*, abs/2107.06426, 2021.
- [10] Liang Qiu, Yizhou Zhao, Weiyan Shi, Yuan Liang, Feng Shi, Tao Yuan, Zhou Yu, and Song-Chun Zhu. Structured attention for unsupervised dialogue structure induction. *CoRR*, abs/2009.08552, 2020.
- [11] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. 11 2019.
- [12] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

- [13] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108, 2019.
- [14] Weiyan Shi, Tiancheng Zhao, and Zhou Yu. Unsupervised dialog structure learning. *CoRR*, abs/1904.03736, 2019.
- [15] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mpnnet: Masked and permuted pre-training for language understanding. 2020.
- [16] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. 2020.
- [17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. 2017.
- [18] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. 2020.

## **А Приложение**



		Twitter Customer Support				DailyDialog			
Model		Silh.	C.-H.	D.-B.	Entr.	Silh.	C.-H.	D.-B.	Entr.
$ V  = 10$	MPNet_KMeans	0.04	672.1	4.018	0.659	0.016	<b>210.4</b>	4.422	0.778
	RoBERTa_KMeans	0.041	634.1	4.147	0.675	<b>0.021</b>	195.1	4.394	0.758
	MiniLM_KMeans	<b>0.054</b>	<b>693.4</b>	3.827	0.668	0.015	198.3	4.482	0.759
	MPNet_GMM	0.037	652.9	<b>3.669</b>	<b>0.597</b>	-0.002	195.5	4.899	0.776
	RoBERTa_GMM	0.036	617.6	3.992	0.611	0.018	185.1	4.724	0.767
	MiniLM_GMM	0.026	668.6	3.807	0.623	0.009	182.8	<b>4.373</b>	<b>0.73</b>
	<i>MPNet_SCAN</i>	0.032	626.664	4.569	0.663	0.014	176.676	5.117	0.809
	<i>RoBERTa_SCAN</i>	0.031	562.999	4.465	0.68	0.021	165.634	5.414	0.809
	<i>MiniLM_SCAN</i>	0.03	585.485	4.278	0.679	0.021	175.363	5.003	0.81
$ V  = 15$	MPNet_KMeans	0.038	530.2	3.704	0.659	0.018	<b>167.7</b>	4.329	0.796
	RoBERTa_KMeans	0.04	490.5	3.936	0.651	<b>0.023</b>	155.2	4.27	0.785
	MiniLM_KMeans	<b>0.049</b>	<b>533.7</b>	3.781	0.668	0.018	160.8	4.234	0.788
	MPNet_GMM	0.015	508.8	3.711	0.626	0.003	158.4	4.404	0.781
	RoBERTa_GMM	0.014	466.8	3.79	<b>0.623</b>	0.016	150.7	<b>4.14</b>	<b>0.772</b>
	MiniLM_GMM	0.03	509.1	<b>3.663</b>	0.637	0.014	155.6	4.341	0.775
	<i>MPNet_SCAN</i>	0.024	475.495	4.457	0.693	0.005	136.29	4.779	0.832
	<i>RoBERTa_SCAN</i>	0.024	447.426	4.418	0.694	0.023	135.245	4.937	0.829
	<i>MiniLM_SCAN</i>	0.024	439.935	4.43	0.696	0.022	140.048	5.132	0.826

Таблица 2: Результаты сравнения качества кластеризации для графа с 10-ю и 15-ю вершинами для двух датасетов: Twitter Customer Support и DailyDialog

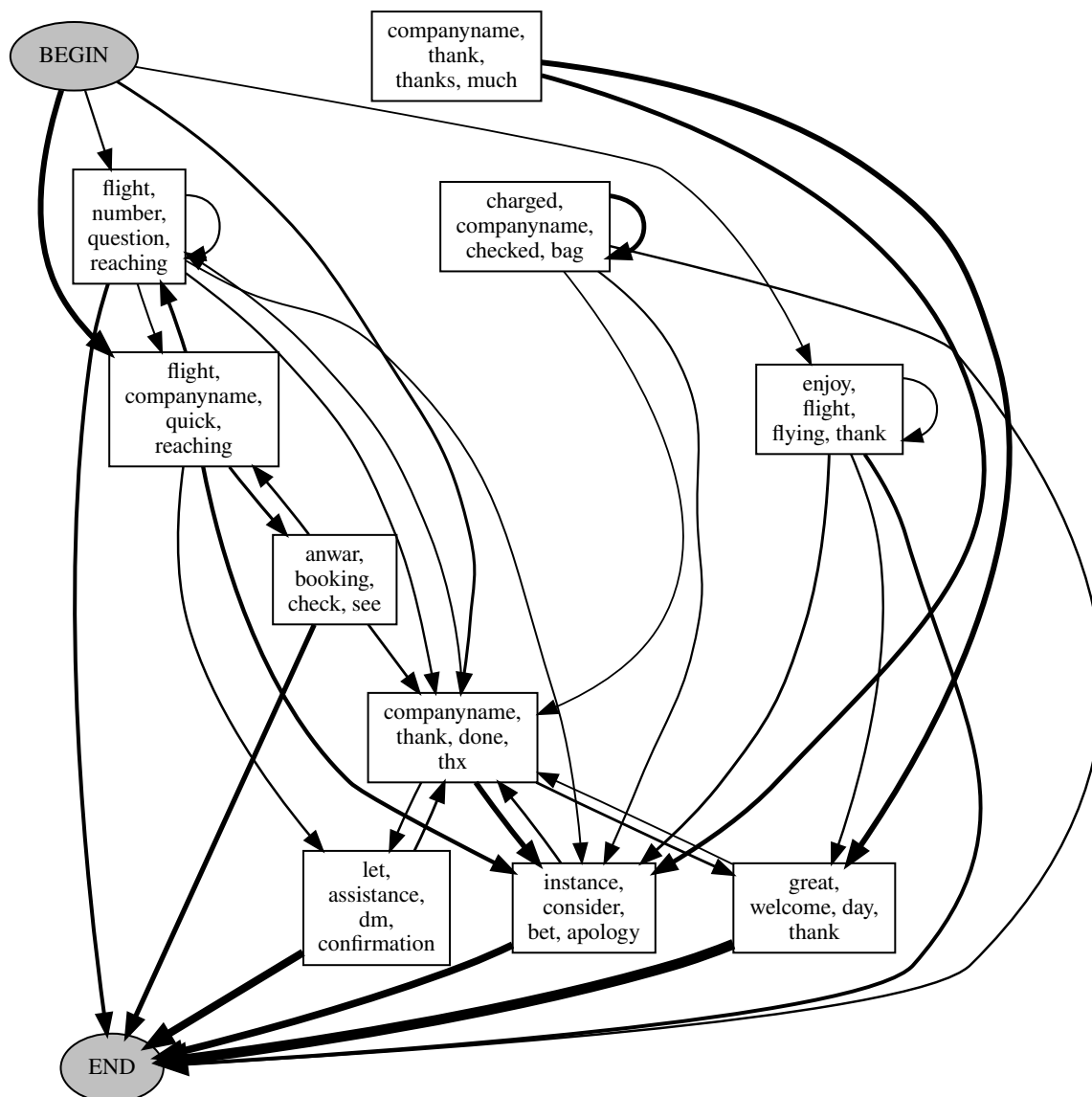


Рис. 9: Граф диалога с 10-ю вершинами для датасета Twitter Customer Support

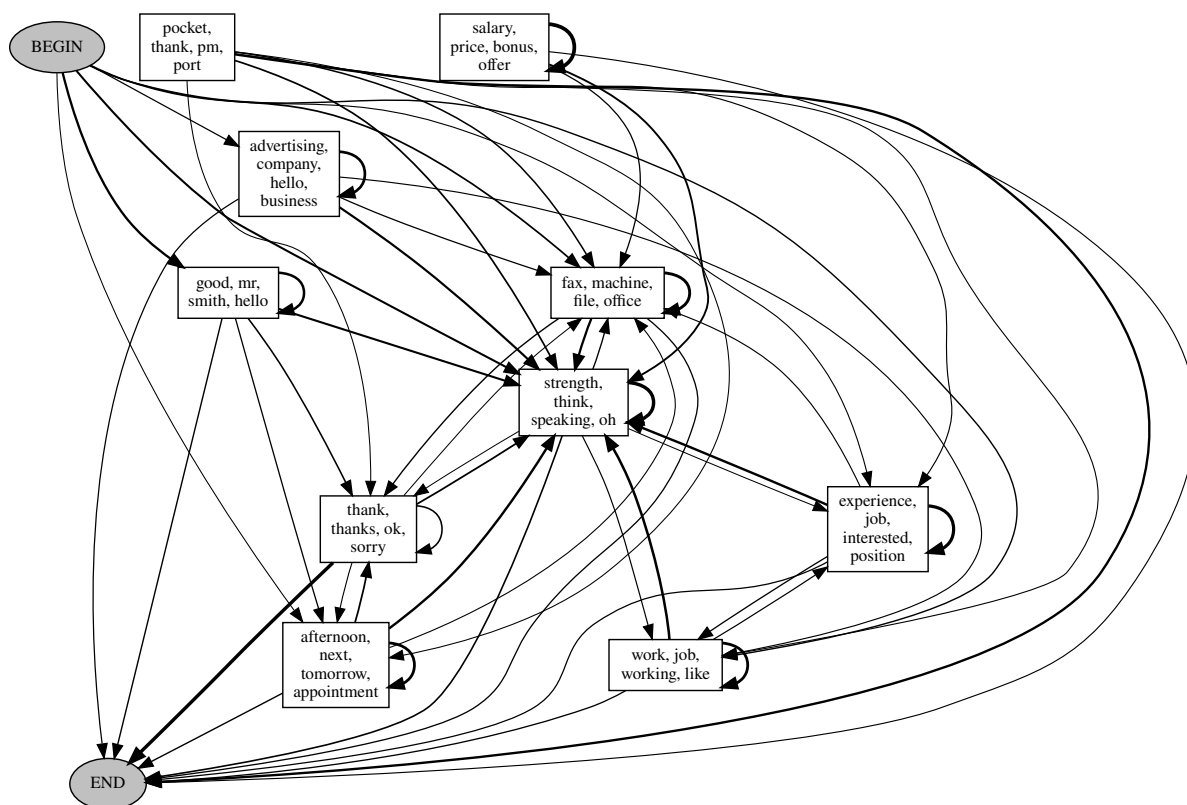


Рис. 10: Граф диалога с 10-ю вершинами для датасета DailyDialog

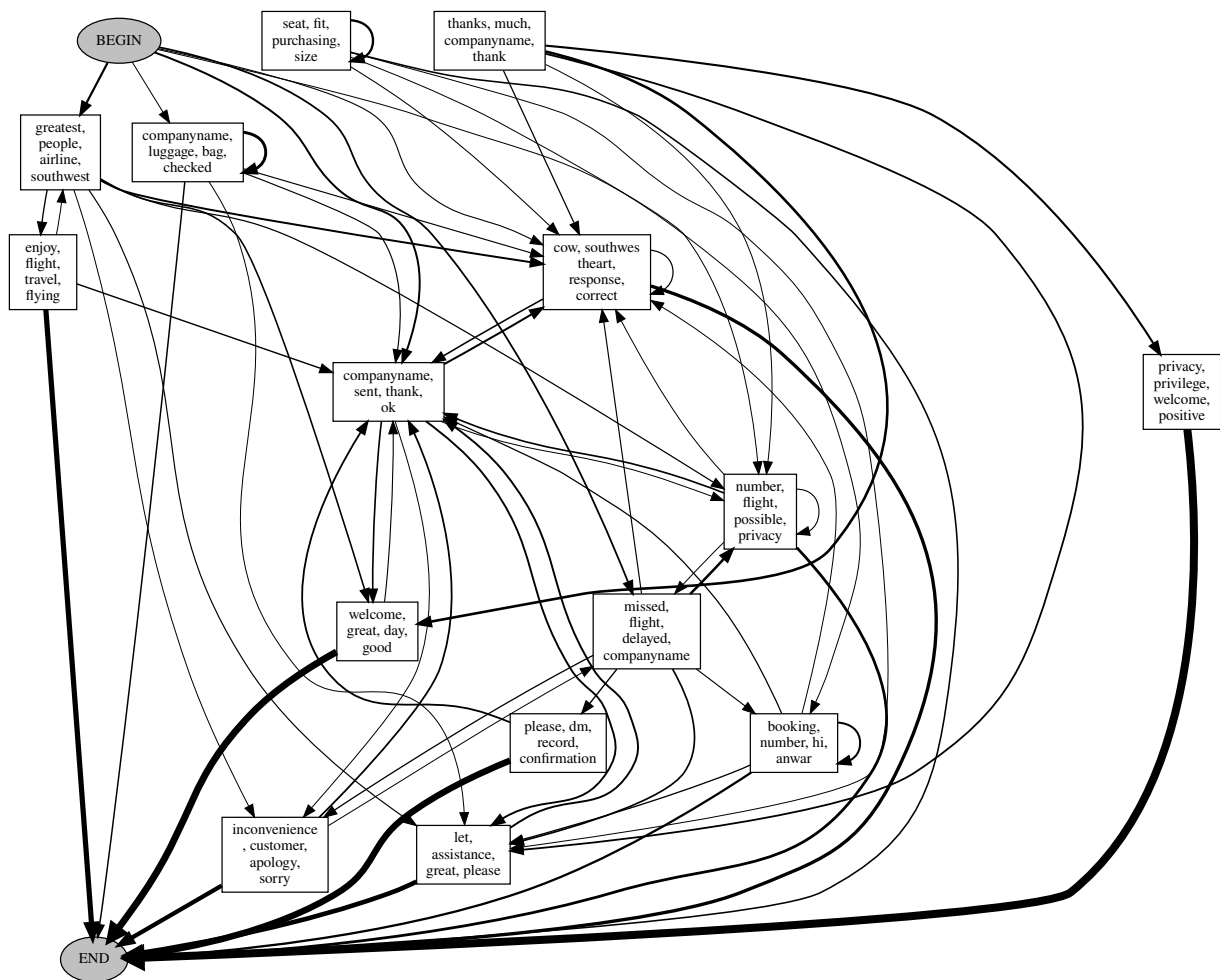


Рис. 11: Граф диалога с 15-ю вершинами для датасета Twitter Customer Support

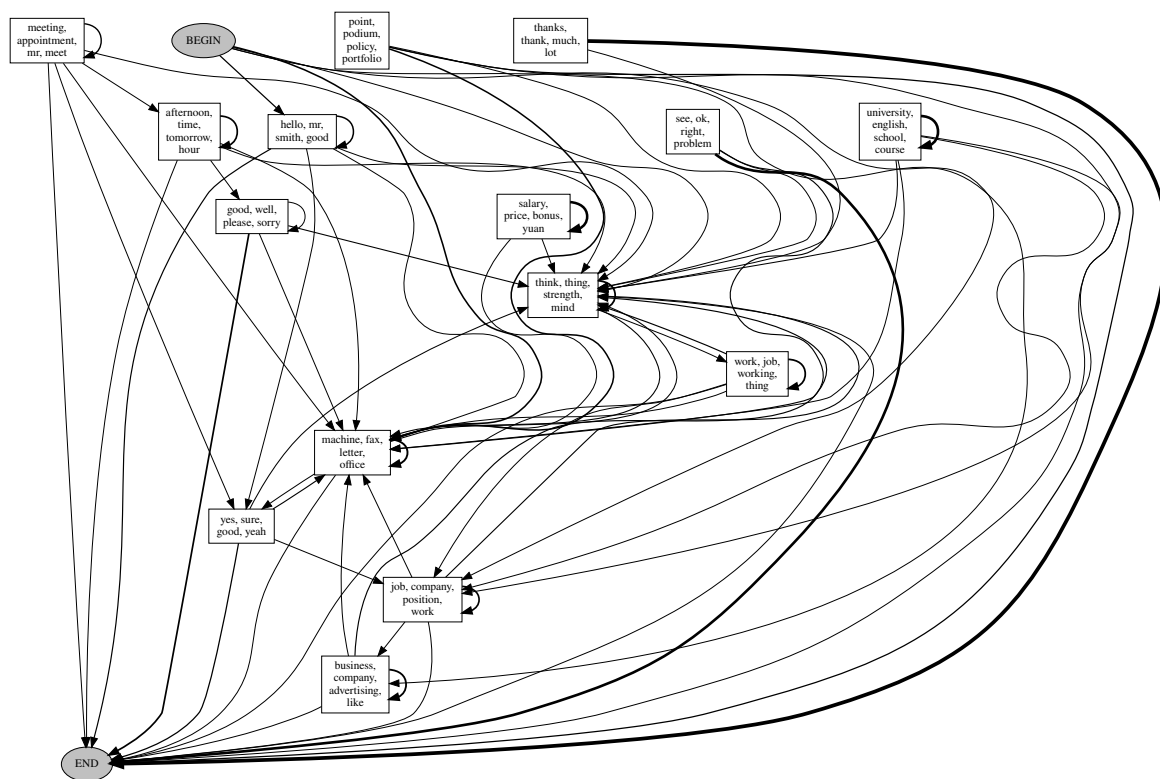


Рис. 12: Граф диалога с 15-ю вершинами для датасета DailyDialog