Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

На правах рукописи

Зейти Бассел Файсалович

Электронное управление средствами массовой информации, понимание текста и генерация текста с использованием глубокого обучения

Направление подготовки	09.06.01	Информатика и вычислительная техника
	Код и наименование	
Направленность	09.06.01_06	Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей
	Код и наименование	

НАУЧНЫЙ ДОКЛАД

об основных результатах научно-квалификационной работы (диссертации)

Автор работы: Зейти Б.Ф.
Научный руководитель: Профессор, Профессор, Черноруцкий И.Г.

Санкт Петербург – 2021

Научно-квалификационная работа выполнена в ВШ/на кафедре Института компьютерных наук и технологий федерального государственного автономного образовательного учреждения высшего образования «Санкт-Петербургский политехнический университет Петра Великого»

Директор ВШ/зав. кафедрой: – *Дробинцев Павел Дмитриевич, кандидат технических наук, доцент*

Научный руководитель: – *Черноруцкий Игорь Георгиевич, Профессор*

Рецензент: – *Дробинцев Павел Дмитриевич, кандидат технических наук, доцент*

С научным докладом можно ознакомиться в библиотеке ФГАОУ ВО «Санкт-Петербургский политехнический университет Петра Великого» и на сайте Электронной библиотеки СПбПУ по адресу: http://elib.spbstu.ru

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность работы

В международной литературе много говорится о Четвертой промышленной революции, которая начала менять почти все в нашей жизни, об информационной революции. Несмотря на то, что мы находимся в начале этой революции, она начала влиять на все детали нашей жизни, и мы живем в то время, когда правительства и компании в равной степени переходят к оцифровке своего бизнеса, чтобы идти в ногу с развитием событий и из страха оседлать волну. катастрофических результатов, которые могут возникнуть в результате этого цифрового изменения.

Каждый исторический скачок был основой будущего. Паровые двигатели, положившие начало первой промышленной революции, проложили путь к буму в электроэнергетике, двигателях внутреннего сгорания и обработке стали в ходе так называемой второй промышленной революции, которая, в свою очередь, привела к развитию автоматизации и вычислений в последнем квартале прошлого года. XX век (третья революция). Компьютер был основой сегодняшней революции искусственного интеллекта, робототехники, Интернета вещей, блокчейна, цифровых валют и прочего, то есть четвертой промышленной революции. Единственная разница между цифровой революцией и ее предшественниками состоит в том, что ее спектр более широк и затрагивает все аспекты жизни, начиная с социальных отношений, проходя через повседневную рабочую переписку со всеми ее деталями, проходя через торговые операции и сопутствующую рекламу и реклама. И теории, которые касаются реальности, в которой мы жили, и сосредоточены на новой виртуальной среде, вызванной неограниченным развитием Интернета, и поэтому необходимо подумать об автоматизации управления этими цифровыми платформами, которые переместили людей из материальной жизни в мирную. полностью виртуальная реальность.

Традиционного маркетингового процесса уже недостаточно в свете новой реальности. Таким образом, в новой среде возник ряд вопросов, на которые необходимо ответить, чтобы добиться успеха в процессе продвижения. Вот эти вопросы:

1- Что такое контент и тип цифровой рекламы (текст, изображение, видео).

2- Кому мы адресуем это объявление? Для какой категории мы будем показывать рекламу?

3- Какую информацию можно извлечь из комментариев потенциальных клиентов, которые видели эту рекламу, и какие ответы необходимо предпринять, чтобы поддержать интерес клиентов к продукту.

Ответ на первый вопрос есть у маркетологов. Что касается второго вопроса, то я попытался изучить его в магистерской диссертации, а в данном исследовании делается попытка ответить на третий вопрос и автоматизировать работу над этим вопросом, проанализировав проблему и разделив ее на основные части:

1- Понимание комментария, который пишет пользователь

2- Пытаться автоматически сформировать соответствующий ответ без вмешательства человеческого фактора

Цель и задачи исследования

Основная цель этого исследования - разработать систему распознавания комментариев клиентов и генерации ответных сообщений.
Основная задача в понимании разговорного языка в целенаправленных системах человеко-машинного разговорного понимания состоит в том, чтобы автоматически извлекать семантические концепции или заполнять набор аргументов или «слотов», встроенных в семантический фрейм, для достижения цели в человеческом теле. -машинный диалог.

Работа была разделена на две части:
1- **первая часть работы** - разработка алгоритмов заполнения слотов и обнаружение сущностей, которые являются основной частью понимания текста.
2- **Вторая часть работы**- генерация текста, которая позволит генерировать автоматические ответы на комментарии пользователей.

Для достижения поставленной цели должны быть решены следующие задачи:
1- Исследовать архитектуры и алгоритмы сетей глубокого обучения, используемых в области обработки текстов
2- Исследовать существующие решения в области распознавания и генерации текста
3- Спроектировать новые нейронные сети для достижения лучших результатов и более высокой точности распознавания и генерации текста
4- Проверка стабильности и достоверности результатов улучшенных нейронных сетей
5- Соединить алгоритм распознавания текста с алгоритмом генерации текста, разработать чат-бота, способного понимать вопросы и генерировать ответы

Научная новизна

1- Была разработана новая структура нейронной сети для обнаружения намерений и заполнения слотов. Новая структура является объединением сети CNN с сетью RNN для получения гибридной структуры.
Новая структура нейрнной сети позволила:
2 - добиться лучшего коэффициента F1 для задачи точности «заполнения щели» 98,21%, где наилучшие опубликованные результаты были 96,43% «Модель ACJIS»;
3 - улучшение коэффициента F1 с точки зрения точности «Обнаружения намерения» в тексте и получения результатов 98,21%, где лучшим результатом был «Структура распространения стека с BERT» с 97,5%.
4- На данный момент задача генерации грамматически верного текста были решены для английского языка. Для других языков было мало экспериментов и исследований. В данной работе был собран набор данных для генерации текста на русском языке. Были проведены эксперименты с текущими методами

генерации текста, а именно VAE и seqGAN. Были подобраны параметры алгоритмов, а также произведено обучение нейронной сети для генерации грамматически и лексически корректного текста.

Теоретическая и практическая значимость

Теоретическая важность исследования заключается в обсуждении и изучении новой и очень важной области в мире бизнеса и менеджмента.
И с научной точки зрения эффективные встроенные алгоритмы были достигнуты как в области понимания письменных текстов, знания того, что требуется от текста, так и создания текстов на основе конкретных данных в качестве ответов на тексты, которые были идентифицированы.
С практической точки зрения результаты этого исследования были использованы для создания эффективного, интегрированного и самообучающегося голосового помощника.
Здесь представлен пример предложения с проиллюстрированными аннотациями домена, намерения и слота / концепции, а также типичными именованными объектами, не зависящими от предметной области. Этот пример следует популярному представлению in / out / begin (IOB), где Бостон и Нью-Йорк - это города отправления и прибытия, указанные в качестве значений слотов в высказывании пользователя, соответственно.
Когда мы начинаем изучать проблему заполнения слотов в понимании разговорной речи, лучший результат в этой области составил 93,07%, полученный структурой RNN / GRU с сетевой архитектурой dropout 0,25. Таким образом, там, где действительно необходимо улучшить предыдущие результаты и уменьшить диапазон ошибок, там, где необходимо было найти новую архитектуру сети, отличную от CNN и RNN.
Классификация большого текста с несколькими метками - это сложная проблема обработки естественного языка (NLP), которая связана с классификацией текста для наборов данных с тысячами меток. Мы решаем эту проблему в правовой области, где наборы данных, такие как JRC-Acquis и EURLEX57K, помеченные словарём EuroVoc, были созданы в правовых информационных системах Европейского Союза. Таксономия EuroVoc включает около 7000 понятий. В этой работе мы изучаем эффективность различных недавних моделей на основе трансформаторов в сочетании с такими стратегиями, как генеративное предварительное обучение, постепенное размораживание и дискриминирующая скорость обучения, чтобы достичь конкурентоспособной производительности классификации, и представляем новые современные результаты 0,661 (F1) для JRC-Acquis и 0,754 для EURLEX57K. Кроме того, мы количественно оцениваем влияние отдельных шагов, таких как точная настройка языковой модели или постепенное размораживание в исследовании абляции, и предоставляем разделение эталонных наборов данных, созданное с помощью итеративного алгоритма стратификации.

Апробация работы

Наиболее важные результаты научных исследований и разработок обсуждались на специализированных конференциях, а результаты исследований публиковались в журналах, получивших рейтинг Scopus.
На практике эти результаты были одобрены частной компанией, на базе которой был построен интегрированный голосовой помощник.

Публикации

Основные результаты по теме диссертации изложены в восьмых научных работах, в том числе три статьи в сборниках докладов конференции , входящих в перечень Scopus.

Представление научного доклада: основные положения

Основные положения защиты:
1. Методы глубокого обучения и алгоритмы понимания языка
Совместное заполнение слотов и обнаружение намерений в понимании разговорной речи с помощью гибридной модели CNN-LSTM
2. Методы глубокого обучения и алгоритмы генерации текста.
Генерация русского естественного языка: создание набора данных для языкового моделирования и оценка с использованием современных нейронных архитектур
Наша идея заключалась в разработке новой архитектуры, сочетающей в себе преимущества и мощь как CNN, так и RNN вместе.
В следующей работе мы решили улучшить результат нашей гибридной модели, поэтому мы исследуем использование гибридных сетей сверточной и долговременной краткосрочной памяти для совместного заполнения слотов и обнаружения намерений при понимании разговорной речи. Мы предлагаем новую модель, которая объединяет сверточные нейронные сети, благодаря их способности обнаруживать сложные особенности во входных последовательностях, применяя фильтры к кадрам этих входных данных, и рекуррентные нейронные сети, учитывая тот факт, что они могут отслеживать длинные и короткие временные зависимости во входных последовательностях. Мы решили построить модель для совместного заполнения слотов и обнаружения намерений, потому что считаем, что существует тесная взаимосвязь между намерениями и семантическими слотами. Совместная модель может отразить эту связь, выяснить ее и использовать для улучшения результатов прогнозирования. Мы используем набор данных Airline Travel Information System (ATIS), чтобы измерить производительность нашей модели и сравнить ее с результатами других моделей, поскольку этот набор данных стал одним из самых популярных наборов данных для проблем с пониманием разговорной речи.

СОДЕРЖАНИЕ РАБОТЫ

(по главам)

Глава 1: **Вступление**

Введение об управлении электронными медиа и новом образе жизни пользователей, а также о необходимости дополнительных исследований в области понимания текста и генерации текста.

Глава 2: **Диалоговые системы и чат-боты**

1-Chatbots

Чат-боты - это системы, которые могут вести расширенные беседы с целью имитации неструктурированного разговора или «чатов», характерных для взаимодействия человека и человека Архитектуры чат-ботов делятся на два класса.

a) Rule-based chatbots

включают ранние влиятельные системы ELIZA и PARRY. ELIZA - самая важная диалоговая система чат-бота в истории отрасли. ELIZA была разработана для имитации психолога Роджера на основе раздела клинической психологии, методы которого включают в себя выведение пациента из состояния за счет отражения им высказываний пациента. Психология Роджера - это редкий тип разговора, в котором, как указывает Вейценбаум, можно «принять позу, почти ничего не зная о реальном мире». Если пациентка говорит: «Я совершила долгую прогулку на лодке», а психиатр говорит: «Расскажите мне о лодках», вы не предполагаете, что она не знала, что такое лодка, а скорее предполагаете, что у нее была какая-то разговорная цель. Большинство чат-ботов, пытающихся пройти тест Тьюринга, выбирают домен с похожими свойствами.

В шаблоне ELIZA 0 означает Kleene *, а в правилах преобразования числа являются индексом составляющей в шаблоне. Таким образом, число 3 относится ко второму 0 в первом шаблоне. Это правило перенесет

\\Я тебе нужен\\

в:

\\Что заставляет тебя думать, что ты мне нужен\\

Каждый шаблон / правило ELIZA связан с ключевым словом, которое может встречаться в предложении пользователя. Архитектура кратко представлена на рис.1

```
function ELIZA GENERATOR(user sentence) returns response

Find the word w in sentence that has the highest keyword rank
  if w exists
      Choose the highest ranked rule r for w that matches sentence
      response ← Apply the transform in r to sentence
      if w = 'my'
          future ← Apply a transformation from the 'memory' rule list to sentence
          Push future onto memory stack
  else (no keyword applies)
    either
        response ← Apply the transform for the NONE keyword to sentence
    or
        response ← Pop the top response from the memory stack
  return(response)
```

Fig.1 ELIZA algorithm architecture

b) Corpus-based chatbots

добывать большие наборы данных разговоров между людьми, что может быть выполнено с помощью поиска информации (системы на основе IR просто копируют ответ человека из предыдущего разговора) или с помощью парадигмы машинного перевода, такой как нейросетевые системы с последовательностью последовательностей, чтобы научиться отображать высказывание пользователя и реакцию системы.

2- Dialogue Systems (DSs)

В настоящее время DS становятся настолько популярными, и большинство людей сталкивались с ними или даже использовали их. DS могут быть встроены в смартфоны, веб-браузеры, автомобили, роботов и другие компьютерные системы. Их можно использовать для разных целей и в различных приложениях, таких как социальное взаимодействие, роботизированные сервисы, мониторинг здоровья, образование и т. д.

DS имеют разные архитектуры, но имеют общие основные компоненты. Эти компоненты показаны на рис. 2:

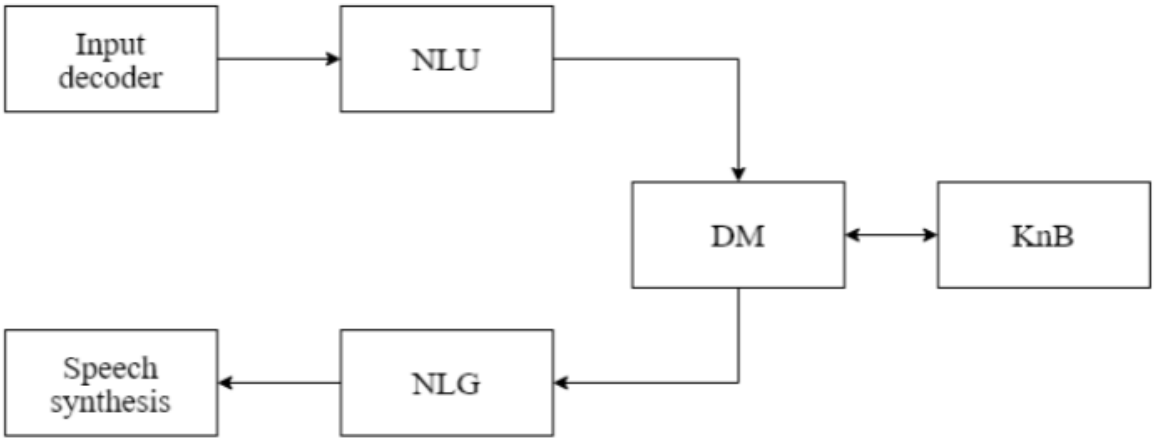


Fig.2 Dialogue System Components

2.1- Input decoder:

Этот компонент отвечает за распознавание ввода пользователя и перевод его в текст, поэтому его можно опустить, если ввод системы является текстом, но не другими формами, такими как голос, жесты и т. д. Этот компонент требует знания фонетики и фонологии, если вводится голосом. Фонетика - это отрасль лингвистики, которая изучает звуки речи, их производство, сочетание, описание и представление письменными символами, в то время как фонология изучает звуки речи или язык по их распределению или правилам произношения. Существует множество систем преобразования речи в текст, они называются системами автоматического распознавания речи (ASR). DS могут иметь другие типы ввода, такие как жесты, почерк и т. д.

2.2- NLU:

Эта компонентная задача состоит в том, чтобы понять, что пользователь хочет от высказывания, т. Е. Извлечь семантику из высказывания пользователя, определяя намерение пользователя (общая цель высказывания), факторы этого намерения (слоты) и область высказывания. Семантическое представление результата может быть использовано менеджером диалога позже.

2.3- Dialogue manager (DM)

Это фундаментальный компонент системы, поскольку он отвечает за управление всеми аспектами диалога. Он анализирует полученное состояние от компонента NLU и дополнительную информацию из базы знаний (KnB), чтобы выполнить несколько фундаментальных задач DS, таких как:

а) Ведение истории диалога.
б) Работа с искаженным или нераспознанным текстом компонентом NLU.
в) определение стратегии диалога.
г) управление потоком диалога.

2.4- Knowledge base (KnB):
Этот компонент хранит информацию, используемую DM, и может хранить общую и конкретную информацию о домене. Кроме того, он представляет собой интерфейс между DM и внешним программным обеспечением, которое DM может использовать, например, базы данных, экспертные системы и т. д. Таким образом, этот компонент также отвечает за преобразование запроса или высказывания из внутреннего формата, используемого DM в формат, используемый внешним программным обеспечением, вызываемым DM.

2.5- Natural language generator (NLG):
Этот компонент генерирует вывод, который будет показан пользователю в виде текста. Принимается решение о том, какая информация должна быть включена и как информация должна быть структурирована. Некоторые системы используют простые методы, такие как вставка и удаление извлеченной информации в заранее определенный шаблон слотов.

2.6- Speech synthesis:
Этот компонент преобразует ответ, сгенерированный NLG, из текста в речевую форму.

3- Classification of dialogue systems by dialogue controlling method
В зависимости от метода управления диалогом DS можно разделить на 3 категории.
а) Системы с конечным числом состояний.
б) Фреймовые системы.
в) План-ориентированная система.

4- Task-oriented dialogue systems
Ориентированные на задачи DS - это системы, которые общаются с пользователями, чтобы помочь пользователю в выполнении некоторых конкретных задач, таких как заказ еды, резервирование столика в ресторане, поиск рейсов, проживания и т. д., Ориентированные на задачи DS можно разделить на 2 категории :

a) Pipeline based systems:
Такая система обычно состоит из нескольких отдельных, последовательно выполняемых модулей; каждый блок берет вывод предыдущего и передает свой вывод следующему. Как правило, основными единицами для таких типов DS являются NLU, средство отслеживания состояния диалога, обучение политике диалога и NLG.
Когда NLU понимает, что пользователь хочет от высказывания, трекер состояния диалога сохраняет состояние диалога в зависимости от его истории, модуль обучения политики диалога изучает, какие действия следует предпринять в зависимости от текущего ввода системы и текущего состояния

диалога, и, наконец, NLU преобразует предпринятые действия в некоторую понятную человеку форму.

b) End-to-end systems:
Вместо использования конвейера из нескольких модулей, зависящих от вывода друг друга, этот тип DS использует один модуль, который может взаимодействовать с внешней базой данных, и просто принимает ввод сразу от пользователя и выводит окончательный результат, не имея этих разделяемые взаимодействующие единицы, как в конвейерных методах. Преимущество таких систем состоит в том, что нам не нужно настраивать интерфейсы между различными модулями или даже редактировать сам модуль каждый раз, когда мы меняем другие модули, которые взаимодействуют с ним, как в методах конвейера.

5- Natural language understanding (NLU)
Понимание разговорного языка (SLU) - ключевой компонент конвейера DS и подтема обработки естественного языка (NLP), которая фокусируется на том, чтобы заставить машину понимать то, что тело хочет сказать из текста. SLU в DS отвечает за извлечение семантики высказываний или запросов пользователей. SLU можно разделить на три основные задачи:
а) классификация доменов
б) обнаружение намерений
в) заполнение щелей
Эти три задачи нацелены на улавливание семантики высказываний пользователей: задача классификации предметной области состоит в том, чтобы идентифицировать предметную область высказывания (например, фильмы, полеты и т. д.), В то время как задача обнаружения намерения состоит в том, чтобы выяснить цель пользователя в отношении высказывания. высказывание (например, найти фильм, найти полет и т. д.), а задача заполнения слота - пометить каждое слово предложения как некоторый параметр намерения, например, для намерения «найти полет» параметры (слоты) могут быть источником , пункт назначения, дата и т. д.
В прошлом основные три задачи SLU обычно рассматривались отдельно. Но в настоящее время наблюдается тенденция к реализации совместных моделей для заполнения слотов и обнаружения намерений, а иногда и для всех трех задач вместе.
В нашем исследовании мы фокусируемся на двух задачах SLU: заполнении слотов и обнаружении намерений. Мы следуем тенденции, которая заключается в построении единой совместной модели для заполнения слотов и обнаружения намерений. Если модель обучается на высказываниях пользователя в паре с их семантическими фреймами (т. Е. Входом модели является последовательность слов (высказывание запроса пользователя), а выходными данными является семантический фрейм этого запроса, включая намерение пользователя и слоты (т. Е. , параметры запроса). По нашему мнению, интентные и семантические слоты имеют сильную взаимосвязь, поэтому построение совместной модели,

которая изучает и то, и другое одновременно, может помочь модели выяснить эту взаимосвязь.

Глава 3: **Генерация текста**

Создание связного, грамматически правильного и значимого текста - очень сложная задача, однако она имеет решающее значение для многих современных систем НЛП. До сих пор исследования в основном были сосредоточены на английском языке, для других языков как стандартизованные наборы данных, так и эксперименты с современными моделями встречаются редко. В этой работе мы i) предоставляем новый эталонный набор данных для моделирования на русском языке, ii) экспериментируем с популярными современными методами генерации текста, а именно с вариационными автокодировщиками и генеративными состязательными сетями, которые мы обучили на новом наборе данных. Мы оцениваем сгенерированный текст по таким показателям, как недоумение, грамматическая правильность и лексическое разнообразие. Генерация текста является ключевым компонентом многих систем НЛП, которые производят текст, таких как системы перевода, диалоговые системы или реферирование текста. Качество сгенерированного текста имеет решающее значение в этих системах, он должен быть последовательным и правильно сформированным, без грамматических ошибок и семантически значимым. Создание текста, похожего на человека, является сложной задачей, оно включает в себя моделирование синтаксических свойств высокого уровня и таких функций, как тональность и тема.

Генерация естественного языка (NLG) создает понятный человеку текст NL систематическим образом - на основе нетекстовых данных (например, базы знаний) или представлений значений (например, данного состояния диалоговой системы). Современные системы NLG часто используют (нейронные) языковые модели. Языковая модель (LM) представляет собой распределение вероятностей по последовательности слов и может использоваться для предсказания следующего слова с учетом входной последовательности.

В последние годы в NLG успешно применяются различные типы архитектур нейронных сетей, такие как вариационные автокодеры (VAE), генеративные состязательные сети (GAN) и рекуррентные нейронные сети (RNN). Здесь мы экспериментируем с этими архитектурами на русском языке.

Цели этого документа: (i) создать эталонный набор данных для языкового моделирования для русского языка, сопоставимый с популярным набором данных Penn Tree Bank (PTB) для английского языка, и (ii) адаптировать и обучить несколько государственных: современных языковых моделей и оценить их в задаче генерации русскоязычного текста. Мы создаем набор данных из 236 тысяч предложений путем выборки из набора данных Lenta News, предварительно обрабатываем текст и фильтруем предложения, не соответствующие определенным критериям качества. Затем мы обучаем шесть моделей (четыре модели VAE с разными методами планирования, seqGAN и LSTM RNNLM) на новом корпусе, оцениваем их по метрике недоумения и

вручную проверяем 100 предложений для каждой модели на предмет грамматической правильности. Мы достигаем лучших результатов с моделями VAE, нулевой вариант хорошо справляется с трудностями, но в целом циклическая модель VAE показывает самую высокую производительность, поскольку она генерирует наибольшую долю грамматически правильных предложений, которые имеют аналогичные характеристики (длина предложения и т. д.) в качестве обучающих данных.

TEXT SUMMARIZATION

Самые ранние работы по автоматическому реферированию текста относятся к 1950-м годам. в
за последние десять лет появилось много новых подходов в результате информационных
перегрузка в Интернете. Недавно было разработано несколько подходов на основе LSA.

1- Surface Level Approaches

Самые старые подходы используют индикаторы поверхностного уровня, чтобы решить, какие части текста важны. Первый алгоритм извлечения предложения был разработан в 1958 году.

Он использовал частоту терминов для измерения релевантности предложения. Идея заключалась в том, что при написании на заданную тему писатель будет повторять определенные слова по мере развития текста. Таким образом, релевантность термина считается пропорциональной его частоте в документе.

Термин «частота» позже используется для оценки и выбора предложений для резюме.

Другими хорошими индикаторами релевантности предложения являются положение предложения в документе, наличие слов заголовка или определенных ключевых слов (например, таких слов, как «важно» или «релевантно»).

2- Corpus-Based Approaches

Вполне вероятно, что документы в определенной области имеют общие термины в этой области, которые
не несут важную информацию. Их актуальность следует снизить. Исследования показали
что релевантность термина в документе обратно пропорциональна количеству документов в корпусе, содержащем термин.

3- Cohesion-Based Approaches

Извлекающие методы могут не уловить отношения между концепциями в тексте.

Анафорическим выражениям 2, которые относятся к событиям и сущностям в тексте, нужны их предшественники, чтобы их можно было понять. Резюме может стать трудным для понимания, если предложение, содержащее анафорическую ссылку, извлечено без предыдущего контекста. Связность текста включает отношения между выражениями, которые определяют связность текста. Связующие свойства текста были исследованы с помощью различных подходов к реферированию.

Лексические цепочки используют базу данных WordNet для определения взаимосвязанных отношений (т. Е. Повторения, синонимии, антонимии, гипернимии и холонимии) между терминами. Затем цепочки состоят из связанных терминов. Их баллы определяются на основе количества и типа отношений в цепочке. Для резюме отбираются предложения, в которых наиболее сильные цепочки сконцентрированы. Похожий метод, в котором предложения оцениваются в соответствии с упомянутыми в них объектами. Объекты идентифицируются системой разрешения совмещенных ссылок. Разрешение совместных ссылок - это процесс определения, относятся ли два выражения на естественном языке к одному и тому же объекту в мире.
Предложения, в которых встречаются часто упоминаемые объекты, переходят в аннотацию.

4- Rhetoric-Based Approaches

Теория риторической структуры (RST) - это теория организации текста. Он состоит из ряда риторических отношений, связывающих вместе текстовые единицы. Отношения соединяют воедино ядро - центральное для цели писателя и сателлит - менее центральный материал. Наконец, составлено древовидное представление. Затем необходимо извлечь текстовые единицы для резюме.
Приговоры наказываются в соответствии с их риторической ролью в дереве. Вес 1 присваивается спутниковым единицам, а вес 0 - единицам ядер. Окончательная оценка предложения дается суммой весов от корня дерева до предложения. В другом подходе каждый родительский узел идентифицирует своих ядерных потомков как выдающихся. Дети повышаются до родительского уровня. Процесс рекурсивен вниз по дереву. Оценка юнита зависит от уровня, полученного им после повышения.

5- Graph-Based Approaches

Алгоритмы на основе графиков, такие как HITS или Google's PageRank, успешно используются при анализе цитирования, социальных сетях и анализе структуры ссылок в Интернете. В алгоритмах ранжирования на основе графов важность вершины в графе рекурсивно вычисляется из всего графа. Модель на основе графов была применена к обработке естественного языка, в результате чего появился TextRank. Далее для суммирования был применен алгоритм ранжирования на основе графов.
Граф строится путем добавления вершины для каждого предложения в тексте, а ребра между вершинами устанавливаются с помощью взаимосвязей предложений. Эти связи определяются с помощью отношения подобия, где сходство измеряется как функция перекрытия контента. Перекрытие двух предложений можно определить просто
как количество общих знаков между лексическими представлениями двух предложений.
После запуска алгоритма ранжирования на графике предложения сортируются в порядке, обратном их оценке, и предложения с самым высоким рейтингом включаются в сводку.

6-Beyond Sentence Extraction

Существует большой разрыв между резюме, создаваемым текущими автоматическими составителями резюме, и рефератами, написанными людьми-профессионалами. Одна из причин заключается в том, что системы не всегда могут правильно определить важные темы статьи. Другой фактор заключается в том, что большинство составителей резюме полагаются на извлечение ключевых предложений или абзацев. Однако, если извлеченные предложения в исходной статье разъединены и соединены вместе в резюме, результат может быть непоследовательным, а иногда даже вводящим в заблуждение. В последнее время начали развиваться некоторые методы реферирования, не связанные с извлечением предложений. Вместо воспроизведения полных предложений из текста эти методы
либо сжимайте предложения, либо заново генерируйте новые предложения с нуля. Стратегия вырезания и вставки была предложена в материалах 1-го заседания Североамериканского отделения Ассоциации компьютерной лингвистики. Авторы выделили шесть операций редактирования при абстрагировании человека:
а) сокращение срока наказания,
б) сочетание предложений,
в) синтаксическое преобразование,
г) лексический перефраз,
д) обобщение и уточнение, и
е) повторный заказ.
ж) перечислить

Глава 4: **Эксперименты**
DS стали настолько популярными в широком спектре приложений, таких как бронирование авиабилетов, отелей или ресторанов, покупка билетов, персональные помощники AI и многие другие. Как показано на Рис 4.1, DS состоят из нескольких основных компонентов, из которых компонент SLU сосредоточен только на этой работе, и особенно в задачах заполнения слотов и обнаружения намерений. В общем, существует два подхода к построению моделей заполнения слотов и обнаружения намерений, первый заключается в построении отдельных моделей для каждой задачи, как показано на Рис 4.2.
Другой подход заключается в разработке совместной модели для решения этих двух задач, которая выбрана в данной работе и показана на Рис 4.2.

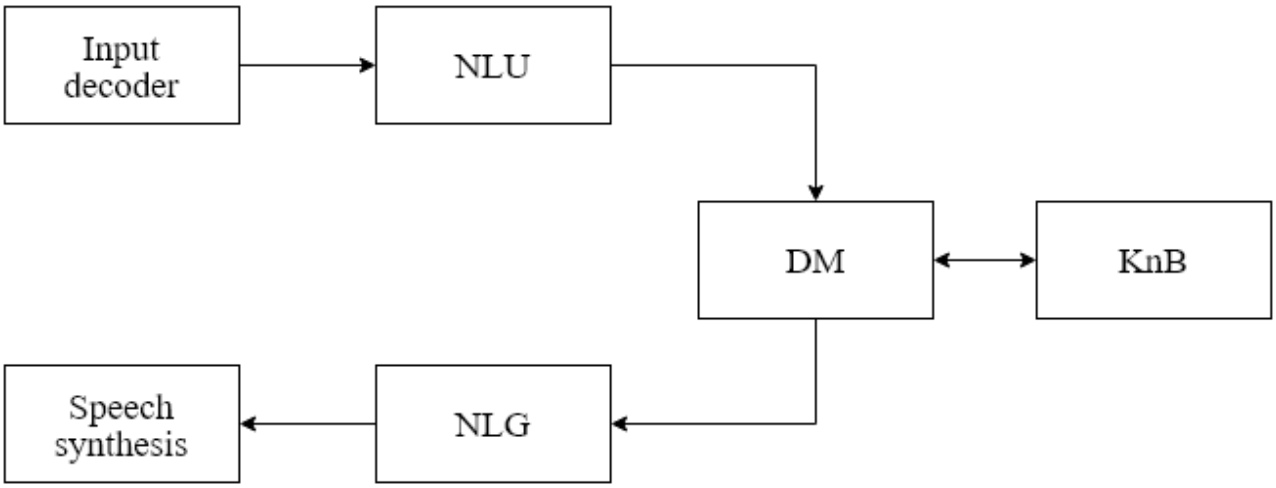
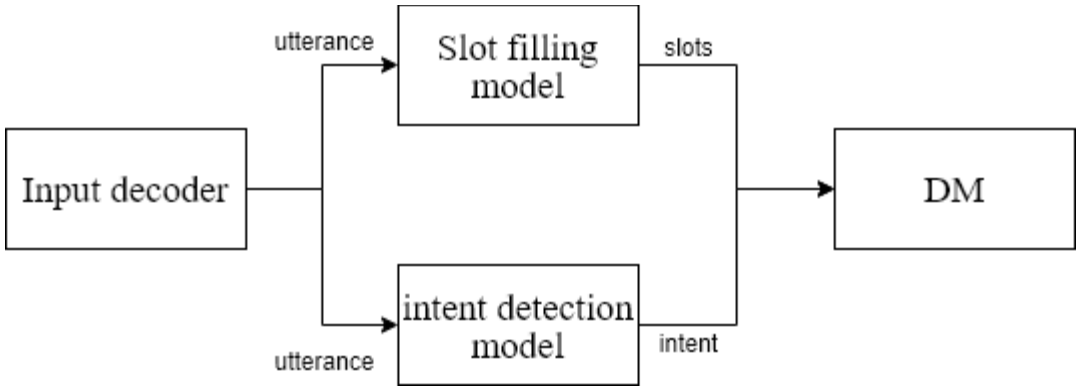


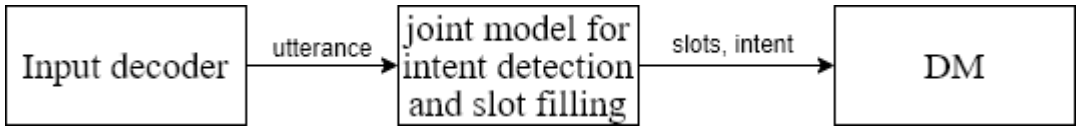
Рис 4.1 Dialogue System Components

Рис 4.2



separate model approach for slot filling and intent detection

Рис 4.3



joint model approach for slot filling and intent detection

Word Preprocessing:

Предварительная обработка слов включает в себя несколько операций, которые применяются, чтобы подготовить данные для подачи в модель, такие как токенизация, построение словарей, построение основных таблиц истинности, преобразование меток в горячие векторы и добавление отступов, чтобы высказывания имели такая же длина.

1- Tokenization:
высказывания в наборе данных преобразуются из строк, содержащих все высказывание, в векторы отдельных слов.
2- Building dictionaries:
Словари «слово в целое» и «целое число в слово» созданы, чтобы упростить и ускорить доступ и представление высказываний и слов из набора данных.
3- Converting the utterances to vectors of integers:
при использовании словарей в целочисленные словари высказывания с их семантическими фреймами преобразуются в векторы целого числа, что удобно в качестве входных данных для моделей, где метки, преобразованные в целочисленные векторы, будут формировать основные таблицы истинности, которые будут используется для оценки производительности моделей.
4- Padding the utterances and the labels to have the same frame:
это необходимый процесс, чтобы соответствовать ограничениям моделей, где эта фиксированная длина используется моделью для определения количества единиц и параметров обучения на каждом уровне.
5- Converting the labels to one-hot vector:
эта операция важна для сравнения меток с выходными данными последнего уровня модели, который имеет функцию активации softmax, которая выводит для каждого слова вектор, размер которого равен количеству различных классов.

Train and Test split:
В задачах машинного обучения данные обычно делятся на два набора обучающих и тестовых, где обучающий набор используется для обучения модели, а набор тестов используется для оценки того, как модель генерирует новые данные, которые она не видела в процессе обучения. . Обычно размер разделения дает больший размер для обучающего набора с почти 60-90% всех доступных данных и 10-40% для тестового набора. Размер может отличаться от одной задачи к другой, в зависимости от мнения разработчиков, размера данных и т. д. В этой работе сохраняется исходный размер разбиения данных. Но данные объединяются, а затем случайным образом разделяются.
Modeling:
В этом разделе слои модели и варианты дизайна подробно объясняются от дизайна ввода-вывода до слоев модели и выбора параметров настройки и других.
Input-output design for joint intent slot filling:
Заполнение слота обычно рассматривается как отображение каждого слова входной последовательности (размера T) в тег в стиле IOB. Итак, если у нас есть входная последовательность как **W = (w1, .. wT),** то выходные теги должны быть **S = s1, .. sT,** в предлагаемой модели мы следуем этому подходу, но для совместного моделирования обнаружения намерения и слота начинки были добавлены некоторые модификации этого подхода.
Only append intent to input and output

Мы добавляем слово в конец входной последовательности (W) и ожидаем, что на выходе оно будет означать намерение, связанное с этим высказыванием, это похоже на подход, использованный в последних результатах исследования, но разница в том, что в последнем results они добавляют статическое слово <EOS> в конец каждого высказывания, что, по нашему мнению, неразумно добавлять одно и то же слово во входные данные и каждый раз ожидать разных выходных данных. В нашем подходе мы не добавляем статическое слово, а динамически добавляем среднее значение вложений всех слов в высказывание, которое, по нашему мнению, разумно отражает различия между высказываниями.

Variational Auto-Encoders

Мы добавляем слово в конец входной последовательности (W) и ожидаем, что на выходе оно будет означать намерение, связанное с этим высказыванием, это похоже на подход, использованный в последних результатах исследования, но разница в том, что в последнем результаты. <EOS> в каждом конце высказывания, что, по нашему мнению, одно и то же слово во входные данные и каждый раз ожидать разных выходных данных. В нашем подходе мы не добавляем статическое слово, а динамически добавляем среднее значение вложений всех слов в высказывание, по нашему мнению, разумно относится к высказываниям.

Вариационный автоэнкодер (VAE) кодирует входную последовательность x в область в скрытом пространстве, а не в одну точку, эта область определяется с использованием многомерного гауссова априорного p (z), где последнее скрытое состояние кодера (z) проецируется на два отдельных вектора. Эти векторы представляют собой среднее значение и диагональную ковариационную матрицу априорной. Чтобы восстановить исходную последовательность, начальное состояние декодера выбирается из предыдущего, а затем используется для декодирования выходной последовательности. Таким образом, модель вынуждена быть способной декодировать правдоподобные предложения из каждой точки скрытого пространства, что имеет разумную вероятность ниже априорной. Стандартная рекуррентная языковая модель нейронной сети основана на серии предсказаний следующего шага, поэтому стандартный AE не обеспечивает интерпретируемое представление глобальных функций, таких как тема или синтаксические свойства высокого уровня.

VAE изменяет архитектуру AE, заменяя детерминированный кодировщик изученной моделью апостериорного распознавания q (z | x). Если бы VAE обучался со стандартной целью реконструкции AE, он бы научился кодировать x детерминированно, сделав q (z | x) исчезающе малым. Однако мы хотим, чтобы апостериорная оценка была близка к априорной (чаще всего стандартной гауссовской), поэтому у нас есть две цели, и цель состоит в том, чтобы оптимизировать следующую нижнюю границу:

$$L(\Theta; x) = -KL(q\Theta(z|x)||p(z)) + E_Z \sim_q \Theta(z|x)[logp\Theta(x|z)] \leq logp(x)$$

Глава 5: **Результаты**

Глава 6: **Заключение**

Объекты, (предмет) и методы исследования

Объектом исследования - достижение высокого уровня понимания текстов, анализа их данных и извлечения того, что от них требуется, а затем способности создавать автоматические тексты для ответа на тексты, которые мы проанализировали

Предметом исследования - управление огромным количеством комментариев, распространяемых в Интернете по определенной теме или продукту, и получение результатов на основе анализа этих данных и выработки ответов, если есть вопросы, без вмешательства человеческого фактора.

Методология и методы исследования. В этом исследовании использовались алгоритмы приложений глубокого искусственного интеллекта, а результаты были проанализированы в соответствии со статистическими стандартами и критериями, а шкала F1 была принята в качестве инструмента для оценки эффективности результатов в алгоритмах распознавания текста.
Статистика ошибок и поправочные коэффициенты использовались для оценки результатов генерации текста.

Результаты и их обсуждение

Результаты применения распределенной системы
Методы были протестированы с различными конфигурациями моделей на наборах эталонных данных ATIS и Snips, а затем сравнились с другими современными моделями с точки зрения заполнения слотов и определения цели. В этом разделе будут показаны результаты моделей с лучшими параметрами. Все рисунки в этой главе будут состоять из двух частей: левой для набора данных ATIS и правой для SNIPs.

Заполнение слотов

Рис. 1 показывает валидацию заполнения слотов F1-scoure как функции от количества итераций для всего диапазона итераций трех моделей. Это показывает, что три модели быстро обучаются в начале и после почти 20-й итерации. F1-scoure этих моделей начинает расти относительно медленнее, чтобы достичь своего максимума к 30-й итерации, что идеально подходит для кривой обучения (learning curve).

То же самое можно увидеть, если мы рассматриваем Рис. 1 , который показывает оценку потери обучения всех моделей в зависимости от количества итераций.

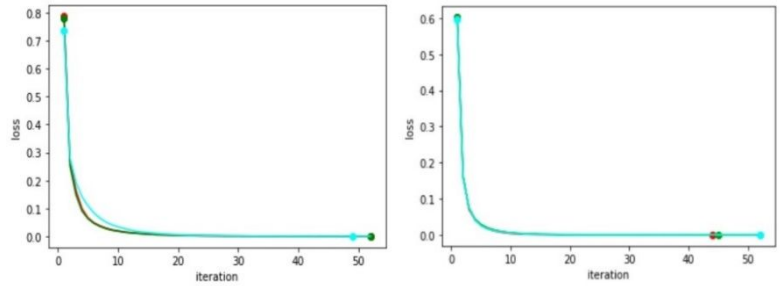


Рис. 1 Потеря обучения как функция от количества итераций для всех моделей

Если увеличить масштаб, чтобы увидеть, какая модель лучше других (т. е. нарисовать графики для итераций более 10) на Рис. 2 , то можно увидеть, что красная линия находится над другими линиями, а это означает, что красная модель имеет лучшую кривую оценки F1 для заполнения слотов.

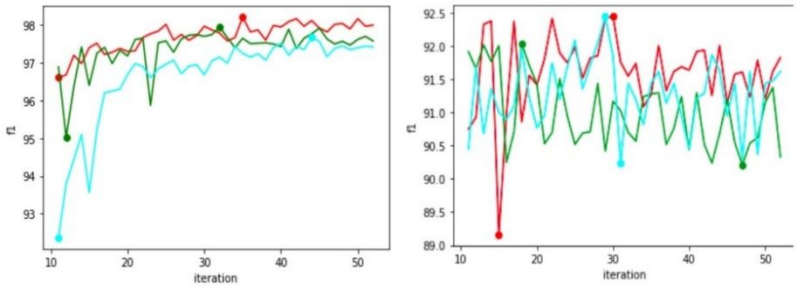


Рис. 2 F1-scoure валидации с увеличением масштаба для задачи заполнения слотов как функция от количества итераций для всех моделей

Если смотреть на Рис. (3, 4, 5) которые показывают для каждой модели свой F1-scoure обучения и валидации в зависимости от количества итераций, можно заметить, что все модели имеют хорошие кривые обучения (без высокого смещения, без высокой дисперсии) для набора данных ATIS, но для набора данных SNIPs рисунки показывают немного более высокую дисперсию, что означает наличие переобучения. Это переобучение может быть связано с тем фактом, что набор данных SNIPs содержит большее количество высказываний, но большее количество словарей и цели и слоты из разных доменов, поэтому ожидается, что получатся лучшие кривые обучения и более высокое f1-score, если будет добавлено больше данных для каждого домена.

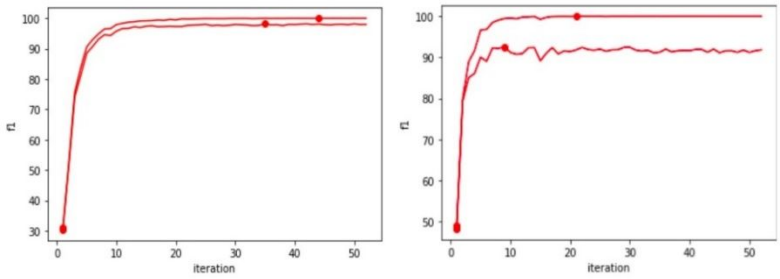


Рис. 3 F1-scoure обучения и валидации заполнения слотов как функция от количества итераций для красной модели

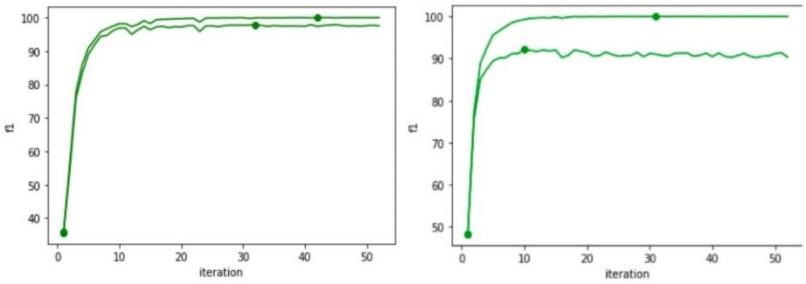


Рис. 4 F1-scoure обучения и валидации заполнения слотов как функция от количества итераций для зеленой модели

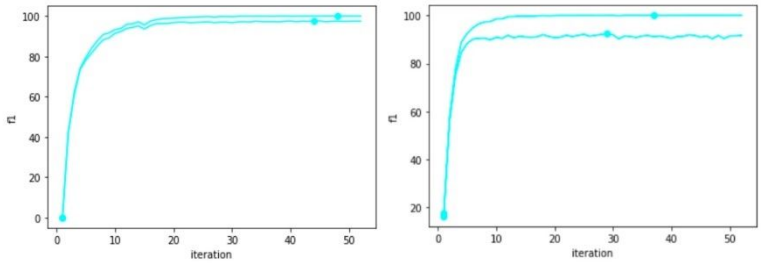


Рис.5 F1-scoure обучения и валидации заполнения слотов как функция от количества итераций для Голубой модели

Определение цели

Рис. 6 показывает точность валидации определения цели для всех моделей в зависимости от числа итераций. Это показывает, что все модели быстро учатся на первых 0-30 итерациях. Затем обучение замедляется и достигает максимума в итерациях 10-50. Также из этого рисунка видно, что красная модель лучше зеленой. Модели голубого и красного цвета имеют почти одинаковую максимальную точность, но голубая модель достигает этого максимума быстрее, чем красная, а кривая голубой модели начинается выше красной. Провал красной модели для набора данных ATIS около 18 итераций может быть связан со слоем случайного выключения, который выбирает случайные узлы для удаления своих выходов на каждой итерации для целей регуляризации. Иногда случаются неудачи, но вся ориентация кривой обучения остается прежней.

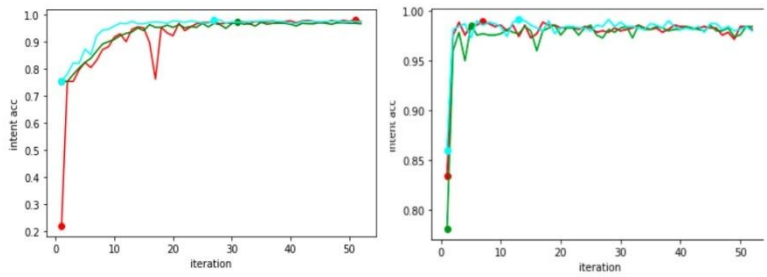


Рис. 6 Точность определения цели как функция от количества итераций для всех моделей

Если посмотреть на Рис. (7, 8, 9), которые показывают точность валидации и обучения определения цели для каждой модели отдельно в зависимости от количества итераций, то можно заметить что красная модель также лучше, чем зеленая и голубая модели с точки зрения переобучения, потому что красная линия показывает меньшую дисперсию, чем зеленая и голубая.

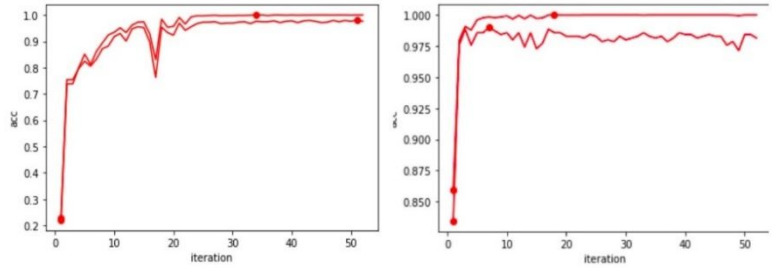


Рис.7 Точность обучения и валидации для задачи определения цели как функция от количества итераций для красной модели

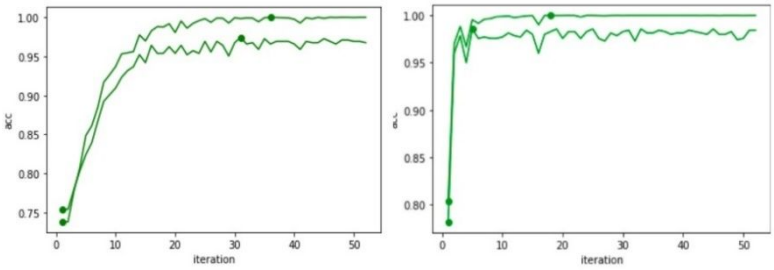


Рис. 8 Точность обучения и валидации для задачи определения цели как функция от количества итераций для зеленой модели

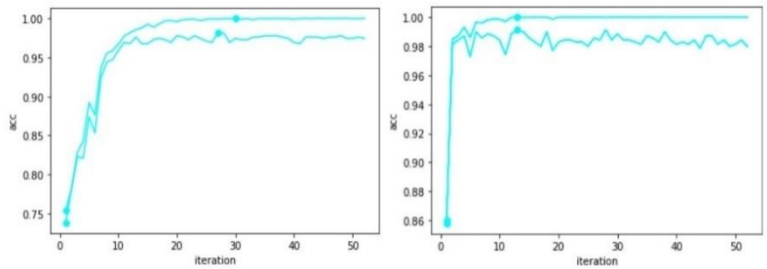


Рис. 9 Точность обучения и валидации для задачи определения цели как функция от количества итераций для голубой модели

Заключение

1- Intent detection and Intent detection :

В процессе оценки мы сосредоточили внимание на разнице между использованием разных архитектур нейронных сетей, мы также сравнили различные методы оптимизации для получения лучшей структуры нейронной сети, и в конце мы включили сравнение, основанное на типе повторяющейся единицы, используемой в модель. Мы завершили эксперименты 25 раз, взяли среднее значение образцов и вычислили стандартную ошибку. Мы представили наши результаты в таблицах.

Наши результаты показывают, что гибридные архитектуры работают лучше, чем другие модели чистой RNN или чистой CNN, когда мы использовали метод оптимизации dropout 0,25 и RMSProb, мы получили F1-балл 95,04 для гибридной модели по сравнению с 91,16 для модели свертки и 93,07 для рекуррентной модели. .

В этой статье рассматривается проблема заполнения слотов в Speken Language Understanding. В частности, мы сосредоточились на маркировке слотов, не обращая внимания на другую часть классификации намерений. Мы сформулировали нашу архитектуру обучения как иерархию пространственных функций CNN, за которыми следуют RNN для моделирования зависимостей во временной области. Экспериментальные результаты на наборе данных ATIS неизменно демонстрируют эффективность предложенного подхода. Стоит отметить, что можно реализовать комбинированные модели, которые решают две задачи одновременно, и доказано, что эти модели приводят к повышению производительности. Но все же для реализации полноценного чат-бота нам нужно будет генерировать текст, похожий на человеческий, в ответ на ввод пользователя. В будущей работе мы намерены изучить включение в нашу модель механизма внимания, который может предоставить дополнительную информацию для прогнозирования метки слота, и изучить нашу архитектуру с использованием других наборов данных для обобщения результатов.

2- **Text Generation**

В этой работе мы представляем новый набор данных для моделирования на русском языке (на основе набора данных Lenta News) и проводим сравнительное исследование двух современных методов генерации текста, а именно VAE и seqGAN. Наши результаты показывают влияние метода планирования на качество сгенерированного текста в VAE, где линейные и циклические расписания генерируют лучшие модели грамматически, однако нулевой метод показал лучшую сложность, но нерегулярное распределение скрытых кодов.

LSTM и SeqGAN смогли воспроизвести среднее значение и дисперсию длины предложений в исходном наборе данных, а также количество уникальных слов. Вкладом этой работы являются: i) предоставление (на GitHub) эталонного набора данных для моделирования на русском языке, содержащего в общей сложности 236 тыс. Предложений, ii) адаптация различных вариантов VAE и seqGAN к русскому тексту, iii) обширные эксперименты и оценки. с выбранными методами глубокого обучения, которые показывают, что

циклический подход VAE в целом работает лучше всего. Дальнейшая работа будет включать более глубокое исследование скрытых представлений, создаваемых VAE (и почему VAE генерируют менее разнообразные предложения), применение современных моделей, таких как LeakGAN и stu

Список работ, опубликованных по теме научно-квалификационной работы (диссертации)
Публикации в изданиях, рецензируемых ВАК

-

Публикации в других изданиях

1- A hybrid convolutional and recurrent network approach for conversational AI in spoken language understanding, B Zaity, H Wannous, Z Shaheen, I Chernoruckiy, PD Drobintsev, V Pak

2- Joint Slot Filling and Intent Detection in Spoken Language Understanding by Hybrid CNN-LSTM Model, MA Ali, B Zaity, P Drobintsev, H Wannous, I Chernoruckiy, A Filchenkov

3- Russian Natural Language Generation: Creation of a Language Modelling Dataset and Evaluation with Modern Neural Architectures, Z Shaheen, G Wohlgenannt, B Zaity, D Mouromtsev, V Pak

Аспирант **_____ФИО**

(подпись)