QA FOR TOURISM: ANSWERING RECOMMENDATION AND COMPARISON QUESTIONS

DANISH CONTRACTOR

AMARNATH AND SHASHI KHOSLA SCHOOL OF IT
INDIAN INSTITUTE OF TECHNOLOGY DELHI
NOVEMBER 2021

QA FOR TOURISM: ANSWERING RECOMMENDATION AND COMPARISON QUESTIONS

by

DANISH CONTRACTOR

AMARNATH AND SHASHI KHOSLA SCHOOL OF IT

Submitted

in fulfillment of the requirements of the degree of Doctor of Philosophy

to the

INDIAN INSTITUTE OF TECHNOLOGY DELHI
NOVEMBER 2021

DEDICATED TO

My parents - Yasmin and Osman

Certificate

This is to certify that the thesis titled **QA For Tourism: Answering Recommendation and Comparison Questions** being submitted by **Danish Contractor** for the award of **Doctor of Philosophy** in Department of Computer Science and Engineering is a record of bona fide work carried out by him under my guidance and supervision at the Amarnath and Shashi Khosla School of IT, Indian Institute of Technology Delhi. The work presented in this thesis has not been submitted elsewhere, either in part or full, for the award of any other degree or diploma unless otherwise stated explicitly. In particular, work done in Chapters 3, 4 and 5 were done jointly with undergraduate students. In each case, the part done by the collaborators appeared in their respective Bachelor's theses.

Mausam
Professor
Department of Computer Science and Engg.
Indian Institute of Technology Delhi
New Delhi- 110016

Parag Singla
Associate Professor
Department of Computer Science and Engg.
Indian Institute of Technology Delhi
New Delhi- 110016

Acknowledgments

I would not have been able to pursue my PhD, were it not for the constant nurturing support of my family - especially my mother Yasmin, who has been so patient and understanding throughout my doctoral journey. A special thanks also goes out to my sister, Seher, for always being there for me. It was only due to the supportive and enabling environment at home that allowed me to dedicate most of my waking hours to my work.

I would like to extend my deepest gratitude to my advisors - Mausam and Parag Singla. Thank you for having faith in me and giving me the freedom to work on problems that I found interesting. I am grateful for the encouragement, the continuous feedback, guidance and support throughout. You've both been a source of inspiration to me.

During the entire duration of my PhD, I have been a full-time employee at IBM Research, which continues to be a great place to work. I have learnt a lot from my colleagues and my PhD would not have been possible without the support of my friends and co-workers at IBM. I would especially like to thank my mentor, Venkat Subramaniam who gave me the first opportunity to learn and grow as a researcher and then also motivated me to pursue a PhD. I would like to thank my managers over the years - Sachin Joshi, Bikram Sengupta, Renuka Sindhgatta, Karthik Sankarnaryanan for trusting me and giving me the time, resources and flexibility to balance my work commitments during the course of my PhD. Thanks goes to my colleagues and teammates Dinesh Raghu, Vineet Kumar, Gaurav Pandey, Dhiraj Madan, Sumit Negi, Sumit Bhatia, Sachin Joshi for their support and helpful technical feedback.

I would like to thank and acknowledge my fantastic student collaborators Poojan Mehta, Barun Patra, Krunal Shah, Aditi Partap and Shashank Goel, all of whom have played a crucial role in shaping this work. Thank you Happy Mittal, Ankit Anand, Dinesh Khandelwal for always being a phone call away and helping me progress through the milestones of the PhD.

Thank you Azalenah, for all the 'thesis snacks' and for painstakingly helping me proof-read the entire (boring) thesis! Thank you Saumya Saxena, Aparajita Bharti, Naaz Mustafa and Wajida Contractor for helping me with translations of the Abstract. A shout-out to my amazing circle of friends - *'Thread'*, *'Doston'*, *'X G'*, *'Cantab'*, *'Kabootars of Trafalgar'*, *'Dim Bits'*, *'Bombay Kids'*, *'Rohit Kumar & Kids'* - thank you for all the joy, jokes and laughter!

Danish Contractor

Abstract

Travellers often post questions online to seek personalized travel recommendations by describing their preferences and constraints with respect to locations, points of interests, budget, etc. They also, at times, post queries asking for comparisons between cities, tourism sites, etc when making their travel plans. In this thesis we study the novel tasks of answering such *recommendation* and *comparison* questions from the tourism domain.

We focus our attention on a class of recommendation questions that seek entities. We refer to them as Multi-sentence entity-seeking recommendation questions (MSRQs) i.e., questions that expect one or more entities as an answer. In the tourism domain, such entities can occur in the form of Points-of-Interest (POIs); e.g, names of hotels, restaurants, tourist sites. We answer entity-seeking recommendation questions in two settings: (i) QA with intermediate annotations (ii) QA without intermediate annotations. In each setting we formulate a new problem and create new datasets which we hope will help further research in QA. In the first setting, we develop a pipelined model which breaks down the task of question-answering into a question-parsing task followed by knowledge-base querying. Learning a question-parser requires large amounts of training data and we overcome this challenge by employing a constraint driven learning framework that uses a small set of expert-annotated questions, along with a larger set of crowd-sourced partially-annotated questions.

In contrast to the first setting, in the second approach we use a collection of reviews to directly answer questions, without explicitly parsing questions. Answering such questions poses novel challenges of reasoning at scale, since review collections for each entity can be very large, noisy, contain subjective opinions, and each question can have thousands of entities to choose from to return as 'possible answers'. In response, we present a cluster-retrieve-rerank architecture that helps address some of these challenges. It first clusters review text for each entity to identify exemplar sentences describing an entity. It then uses a scalable neural information retrieval (IR) module to select a set of potential entities from the large candidate set. A reranker uses a deeper attention-based architecture to pick the best answers from the selected entities. Additionally, in order to accommodate reasoning over physical locations of entities, we extend this work by developing a joint spatio-textual model. We develop a modular spatial-reasoning network that uses geo-coordinates of location names mentioned in a question, and, of candidate answer entities, to reason over only spatial constraints. We combine the spatial-reasoner with the textual QA system to develop a joint spatio-textual QA model. We demonstrate that our joint spatio-textual model performs significantly better than models employing only spatial or textual reasoning.

Lastly, we also study the problem of answering comparison questions. We define a novel task of generating entity comparisons from textual corpora in which each document describes one entity at a time. We generate entity comparisons in a tabular form in which attribute-value phrases, opinion phrases,and other descriptions are clustered and organized topically, thus, allowing for direct comparisons. Our tabular summaries balance information about the entities being compared and in our user studies we find that users strongly preferred balanced clusters, and acquire as much information about the entities, by using the tables, as they do using articles.

सार

यात्री अक्सर अपनी पसंद, रुचियों के स्थल , बजट आदि के ज़रूरत के संबंध में व्यक्तिगत यात्रा सुझाव प्राप्त करने के लिए ऑनलाइन प्रश्न पोस्ट करते हैं। वे कभी–कभी अपने प्रश्न बनाते समय शहरों, पर्यटन स्थलों आदि के बीच तुलना करने के लिए भी पूछते हैं। इस थीसिस में हम पर्यटन से इस तरह के सुझाव और तुलनात्मक प्रश्नों के उत्तर देने के नए कार्यों का अध्ययन करते हैं। हम अपना ध्यान सुझाव प्रश्नों के एक वर्ग पर केंद्रित करते हैं जो अहम स्थलों की तलाश करते हैं। हम उन्हें बहु–वाक्य एंटिटी–खोज सुझाव प्रश्न के रूप में संदर्भित करते हैं, अर्थात, ऐसे प्रश्न जो उत्तर के रूप में एक या अधिक स्थलों की अपेक्षा करते हैं। पर्यटन क्षेत्र में, ऐसी स्थल 'रुचि के स्थल' (पीओआई) के रूप में हो सकती हैं; जैसे, होटल, रेस्तरां, पर्यटन स्थलों के नाम। हम सुझाव चाहने वाले प्रश्नों का उत्तर दो सेटिंग्स में देते हैं: (i) मध्यवर्ती एनोटेशन के साथ प्रश्नोत्तर (ii) मध्यवर्ती एनोटेशन के बिना प्रश्नोत्तर। प्रत्येक सेटिंग में हम एक नई दिशा में अन्वेषण करते हैं और नए डेटासेट बनाते हैं जो हमें उम्मीद है कि प्रश्नोत्तर में आगे के शोध में मदद करेगा। अंत में, हम तुलनात्मक प्रश्नों के उत्तर देने का भी प्रयत्न करते हैं। हम टेक्स्ट कॉर्पोरा से वस्तु तुलना उत्पन्न करने का एक नया कार्य परिभाषित करते हैं जिसमें प्रत्येक दस्तावेज़ एक समय में एक एंटिटी का वर्णन करता है। हम एक सारणीबद्ध रूप में एंटिटी तुलना उत्पन्न करते हैं जिसमें विशेषता–मूल्य वाक्यांश और राय वाक्यांश होते हैं। हमारे सारणीबद्ध सारांश तुलना जानकारी को संतुलित करते हैं और हमारे उपयोगकर्ता अध्ययनों में हम पाते हैं कि वह संतुलित समूहों को दृढ़ता से पसंद करते हैं।

تلخیص

مسافرین اکثر اپنی ترجیحات اور تحدیدات بیان کرتے ہوئے مخصوص مقامات ' پسندیدہ مقامات' بجٹ وغیرہ سے متعلق سفری تجاویز حاصل کرنے کیلئے آن لائن سوالات پوسٹ کرتے ہیں۔اور ساتھ ھی سفر کی منصوبہ بندی کے وقت وہ شہروں ' سیاحتی مقامات کے درمیان موازنہ سے متعلق سوالات بھی پوسٹ کرتے ہیں۔ان مقالہ جات میں ھم شعبہ سیاحت سے متعلق اس قسم کی تجاویز اورموازنہ کے جوابات دینے کے نئے کام کا جائزہ لینگے۔ھم اپنی توجہ تجاوزیز اور سوالات پر مرکوز کئے ہوئے ہیجو کسی چیز کی طالب ہیں۔ھم اسے ملٹی سینٹنس انٹیٹی سیکنگ تجاویز سوالات(MSRQs) قراردیتے ہیں مثلاً ایسے سوالات جو ایک یا ایک سے زائد جوابات چاھتے ہیں۔اس قسم کے انٹی ٹیزشعبہ سیاحت میں پوائنٹس آف انٹرسٹ کی(POIs) طرح ہوتے ہیں ' مثلاً ہوٹلس' رسٹورنٹ' سیاحتی مقامات ۔ھم انٹیٹی سیکنگ تجاوزیر اور سوالات کے دو ترتیب میں جواب دیتے ہیں انٹرمیڈیٹQA(i): تشریحات کیساتھQA(ii)انٹرمیڈیٹ تشریحات کے بغیر۔ھر ترتیب میں ھم ایک نیا مسئلہ وضع کرتے ہیں اور نئے ڈیٹاسیٹس بناتے ہیں جس سے امید ھے کہ ھمیآئندہ QA کی تحقیق میںمزید مدد ملے گی۔آخر میں ھم موازنہ کے سوالات کا جواب دینے سے متعلق مسئلہ کی بھی اسٹڈی کرتے ہیں۔ھم ٹیکسٹل کارپورا سے انٹیٹی موازنہ کو نکالنے کے کام کو وضع کرتے ہیں جس میں ھر ڈاکیومنٹ ایک وقت میں ایک انٹیٹی کو بیان کرتا ھے۔ھم انٹیٹی موازنہ کو جدولی شکل مینتیار کرتے ہیں جس میں اٹربیوٹ والیوفریسس' تجاوزیر کے فریسس اور دیگر وضاحتیں کلسٹرڈ اور لفظی اعتبار سے منظم ہوتی ھے۔جو براہ راست موزانہ کی اجازت دیتی ھے۔انٹیٹس کے بارے میں ھمارے جدولی خلاصے ' متوازن انفارمیشن جسکا موازنہ کیا جاتا ھے ۔ ھمارے استفادہ کنندوں کی تحقیق میں ھم نے یہ بھی پایا کہ استفادہ کنندے متوازن کلسٹرس کو زیادہ ترجیح دیتے ہیں۔

Contents

Variable	Definition
i, j, k, l	Locally declared indexing variables
x, y, z	Locally declared variables to define functions
$\mathbb{M}$	Training Set
$\mathbb{C}$	Set of cities
$\mathbb{E}$	Set of entities
$\mathbb{U}$	Set of partially labeled posts
$\mathbb{P}$	Set of POI types supported; $\mathbb{P}$ = {hotel, attraction, restaurant}
p	POI type p $\in \mathbb{P}$
q	a user question
q	Embedding representation of q
e	An entity e $\in \mathbb{E}$
e	Embedding representation of e
$\mathbf{E_e}$	Matrix consisting of sentence embeddings for an entity e
$\hat{e}_q$	Question-aware embedding representation of e
ϕ	CRF Feature
ω	CRF Feature Weight
ρ_k	Weight associated with k^{th} constraint in Constraint Conditional Modeling (CCM)
$\mathcal{C}_k$	Violation score associated with the kth constraint in CCM
γ	Weight to control importance given to partially labeled posts in CCM
$\mathbf{A}$	Attention weights (matrix) for an encoded sequence
$\mathbf{H}$	Matrix consisting of hidden states of an encoded sequence
$\mathbf{W_E}$	Weight matrix for computing question-entity attention
$\mathbf{A_E}$	Attention weights (matrix) for generating entity embeddings
$\mathbf{w}$	Distance weight vector
w_i^d	Distance weight of the ith location-mention
d_k	Distance of an entity from the kth location mention in a question
$\mathbf{d}'$	Distance vector
$\mathbf{A_l}$,b_l	Weight matrix and bias term respectively for any feed-forward block at layer l
$\mathbf{B}$	Vector of position indices in question with B label after $B-I$ encoding
$\mathcal{S}_\mathcal{T}$	Textual Reasoning Score
ψ_T	Scaling weights for $\mathcal{S}_\mathcal{T}$
$\mathcal{S}_\mathcal{L}$	Spatial Reasoning Score
ψ_L	Scaling weights for $\mathcal{S}_\mathcal{L}$
$\mathcal{S}$	Entity Relevance Score
σ	Sigmoid function
$\bar{\rho}$	Spearman's rank coefficient
α, β	Weights for joint scoring in Spatio-Textual Reasoning
Θ	Parameters of a model
η	Regularization Weight for EB G-pLSA

Table 1: List of variables

List of Figures

List of Tables

Part I

Prologue

Chapter 1

Introduction

According to a 2019 report[1] by Bain & Company, travellers make between $33 - 500$ web-searches before making bookings; some users consult in excess of 50 travel websites, spending a third of their time online conducting travel related activities. A recent survey,[2] by the popular hotel booking website Booking.com, found that nearly 57% of the $12,500$ respondents wanted a single application that could support their planning, booking and travel needs, while nearly 31% of global travellers report wanting voice activated assistants, to answer travel queries.

The global leisure-tourism spending in 2019 was estimated to be approximately USD 4.7 trillion[3] and today, there are hundreds of online services aimed at easing the burden of travel-planning for travellers – for example, travel aggregator websites search multiple service providers to suggest optimal flight schedules, hotel reservations, car rentals, etc based on a user's location, destination and travel dates (Figure 1.1). In addition to commercial systems, researchers have also developed methods that could help the tourism industry – for example, methods that provide Points-of-Interest (POI) recommendations using 'spatial' and user 'preference' features [Cong et al., 2009, Zhang et al., 2016, Tsatsanifos and Vlachou, 2015, Li et al., 2016], a users' social media profile [Yiu et al., 2007] or web-search click-through logs [Zhao et al., 2019a]; other methods include those that generate tourist itineraries based on route maps, user interests and travel time [Jiaoman et al., 2018, Padia et al., 2019].

We find that, though there are a wide range of solutions and methods designed towards improving the experience of travel-planning, none of them allow a user to express their travel requirements in free-form natural language text. As a result, users often do not find exact answers to their needs and spend hours online, scouting websites for their

[1] https://www.bain.com/insights/todays-traveler-infinite-paths-to-purchase/

[2] https://globalnews.booking.com/bookingcom-reveals-8-travel-predictions-for-2019

[3] https://www.statista.com/statistics/1093335/leisure-travel-spending-worldwide/

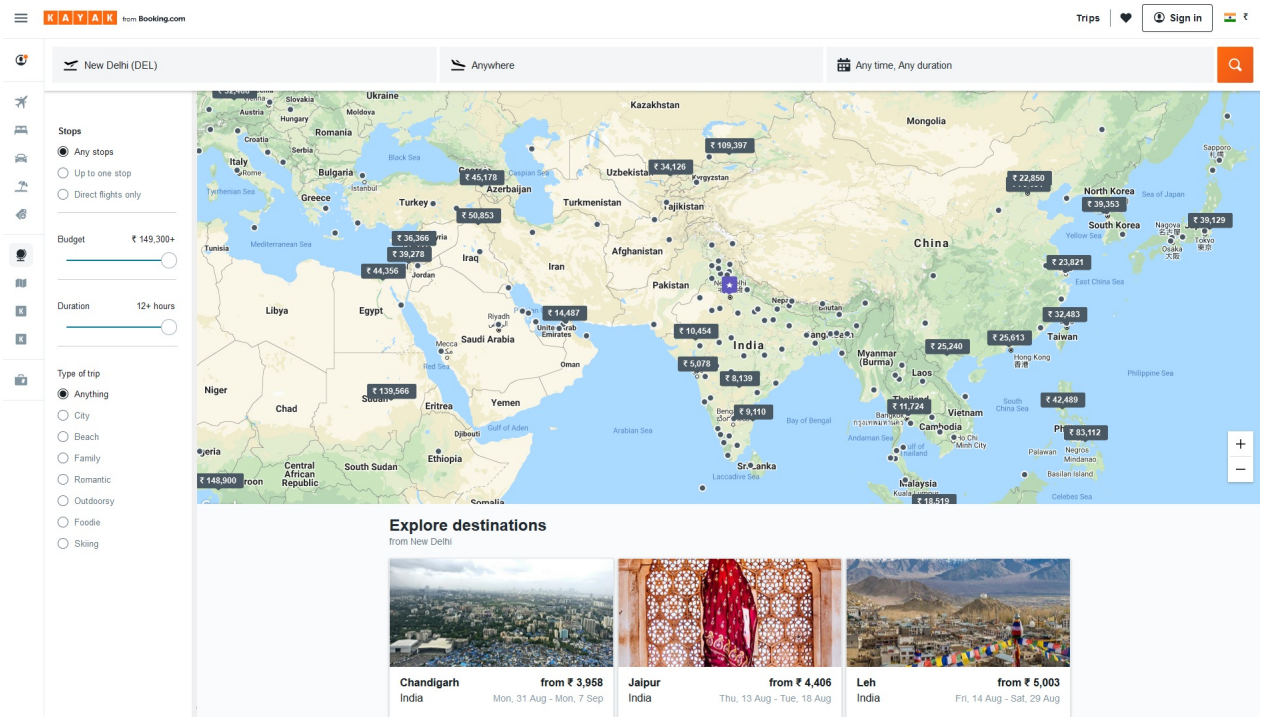


Figure 1.1: Travel aggregator website Kayak.com allows searching for multi-destination flight options, hotels, deals, etc.

specific travel-related queries. In some cases they may even post their questions on travel forums, in the hope of getting personalized travel information from other users. In 2016, TripAdvisor.com reported[4] over 900,000 new topics being created on its travel forums annually.

The fields of language processing and QA have made rapid progress in the last few years; yet, why do travelers need to rely on user responses in web-forums to seek travel information? Shouldn't a QA system be able to serve a user's travel needs? In this thesis, we take the first step towards solving this problem, by developing novel methods aimed at returning answers to the questions users post on travel forums. Figure 1.2 shows an example of a question posted on the popular travel forum (Trip Advisor), along with the responses from other forum users. As can be seen, the traveller is interested in finding restaurant recommendations that would be suitable for children. The traveller also describes their cuisine preferences, budgetary constraints,etc and other forum users respond by offering suggestions of restaurants that might fulfil those requirements.

[4]https://www.tripadvisor.com/PressCenter-c4-Fact_Sheet.html

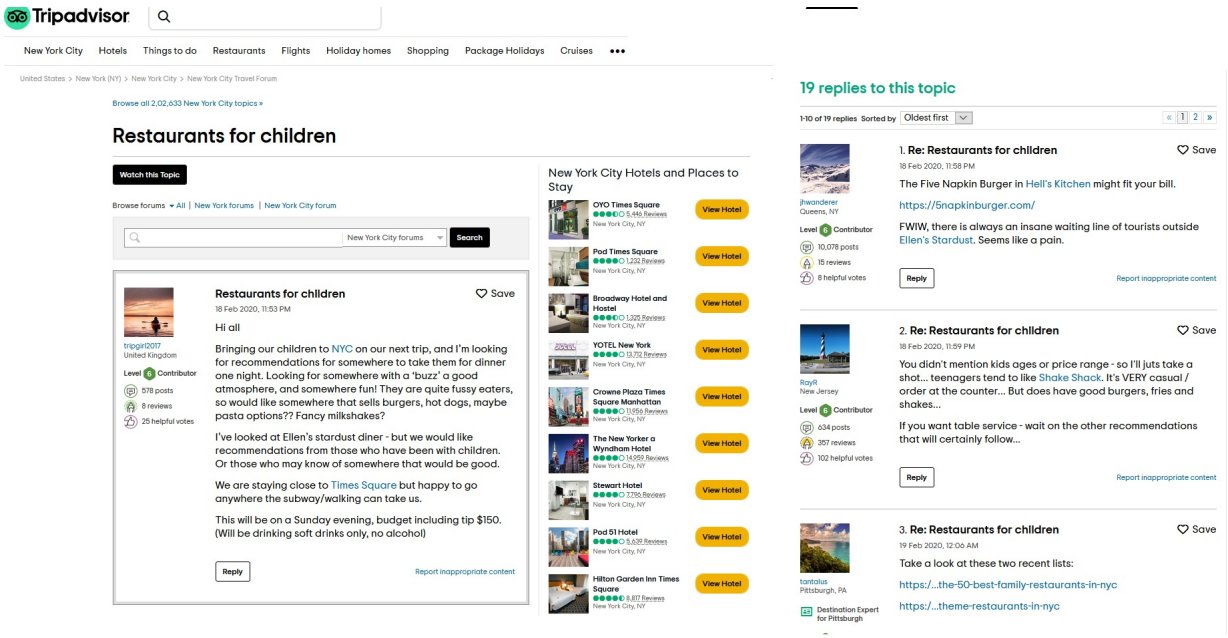


Figure 1.2: A question posted on a popular travel forum website - TripAdvisor.com along with responses from forum users.

1.1 Types of Questions in a Travel forum

Travel forums contain a variety of user questions associated with planning and seeking recommendations. We present a few types of questions, along with real-world examples of each from TripAdvisor.com:

- **Recommendation Questions:** Similar to the example in Figure 1.2, in such questions, users ask for personalized travel recommendations based on their preferences (and constraints) which may include budgets, locations, timings, etc. Example: *"We're arriving into Havana from the UK in the late afternoon and are staying at the Hotel Florida. By the time we get there and check in it will be early evening. Can anyone recommend a good restaurant nearby so that we dont have to venture too far on our first night after a long journey? The Hotel is at Calle Obispoesq. a Cuba. Ciudad de La Habana. Any suggestions appreciated"*

- **Comparison Questions:** In these questions, users post queries asking for comparisons on cities, tourism sites, etc. Example: *"I'm traveling to Prague and Belgrade for two weeks. From Belgrade we are thinking to travel to either Rome or Athens for 3 days. Anyone suggest one over the other?"*

- **Validation / Suggestion of itineraries/ Route enquiries:** Users post queries where they outline a specific travel plan and ask for feedback from the forum com-

munity. Users may sometimes even request for complete itineraries after specifying their needs. Example: *"My husband and I are making our first trip to the Middle East in October 2016. I have come up with an itinerary but wanted to put it out there to see if any one had any suggestions or if they saw something that they might think would be challenging. I would greatly appreciate any feedback: Fly from Los Angeles to Tel Aviv on Oct 6. Fly from Tel Aviv (830 am) to Amman (915 am) on Oct 8. Stay in Amman from Oct 8 - 11. Fly from Amman (1005 am) to Cairo (1035 am) on Oct 11. Stay in Cairo from Oct 11 - 13. Fly from Cairo (645 am) to Amman (905 am) on Oct 13. Fly from Amman (7 am) to Tel Aviv (745 am) on Oct 14. Stay in Israel from Oct 14 - 20. Fly from Tel Aviv (1155 am) to Los Angeles (911 pm) on Oct 20. The itinerary seems pretty good to me but I wanted to make sure I wasn't missing anything. Thanks for your help!"*

- **Look up/ Fact-check questions** These questions pertain to a factual query – such as asking about timings or if there is a certain kind of service available, etc. Example: *"We are going to Saranda for two weeks late May - early June (flying from Sweden to Corfu and taking the ferry to Saranda). We are a family of four, and we would like to rent a car in Saranda to see more of Albania. Is this possible? We would need also to rent a child seat for our youngest child (11 months old)."*

1.2 Scope of Research

As can be seen, users post a variety of questions on travel forums and each class of questions requires a different type of QA system for answering. To take tractable first steps in answering questions on tourism forums, we limit our work in this thesis to two classes of questions:

- **Recommendation Questions**: We work with recommendation questions that seek Points-of-Interest (POIs), such as hotels, attractions and restaurants, as answers. Such questions may consist of multiple sentences and express vague or underspecified requirements, that result in subjective answers. In addition, users may also express preferences and constraints about budgets, locations, timings, etc which requires deeper reasoning and the use of external knowledge sources for answering. We refer to these types of questions as multi-sentence entity-seeking recommendation questions (MSRQ) or POI-recommendation questions.

- **Comparison Questions** We work with questions that require comparison of entities and to demonstrate our work, we use 'cities' as our entity type. We return

comparisons in the form of a tabular summary consisting of phrases describing the entities under consideration.

1.3 Contributions

To the best of our knowledge, we are the first to present work on answering such recommendation and comparison questions. We define a series of novel research tasks and introduce new datasets along with our models for answering both types of questions. We hope these will help improve the state-of-the-art in QA and also help build commercial QA systems for tourism. While our methods have been developed for tourism questions, we also demonstrate applicability of work in other domains, where feasible.

1.3.1 Answering Recommendation Questions

To address the challenges associated with answering recommendation questions, we begin by studying the problem of Question-Understanding. This helps develop a pipelined QA model that first parses a question, and then retrieves an answer entity (POI) from a downstream knowledge store[5] (Chapter 3). The model we develop has the advantage of requiring very little training data and has the ability to use existing query-based knowledge stores. However, relying on an existing black-box knowledge source has drawbacks – for instance, we are limited by the query end-points exposed by the knowledge store and we have no control over the reasoning process employed for answering. We therefore, also answer questions directly using a collection of POI-entity reviews and a set of labeled QA pairs, without generating an intermediate semantic representation for questions (Chapter 4). In contrast to the previous approach, while this method requires large amounts of training data (in the form of QA pairs), it allows us to develop answering methods with deeper reasoning. Finally, as mentioned previously, questions can also encode constraints that require reasoning on budgets, timings, locations, etc. Thus, we also develop a joint spatio-textual reasoning model capable of answering questions that express constraints over one of these features – locations (Chapter 5). We summarize the QA tasks defined for answering recommendation questions, along with their contributions below.

1.3.1.1 Task 1: QA with Intermediate Annotations

- **Problem Definition:** We formulate the problem of understanding (parsing) multi-sentence entity-seeking recommendation questions (MSRQ) as a semantic labeling

[5]We use Google Places.

task, over an *open* representation. It makes minimal assumptions about the schema of the answering knowledge base. Each token in the question is associated with a semantic label which helps identify salient information in the MSRQ – for example, the *type*, *attribute* or *location* of the target answer entity. We then use the labeled tokens to construct a *query* that is executed on the Google Places API, to return entity answers.

- **Challenge:** *Low-Resource Training* – Building typical machine learning models to automatically label tokens in an MSRQ, requires large amounts of annotated data. However, due to the complexity of the questions in our task, sourcing completely labeled questions is not only expensive and time consuming, it is also error prone. We therefore, need to develop methods that can operate in low data settings and can also effectively utilize partially labeled sequences, as training data.

- **Contribution:** At the core of our model, we use a BiLSTM (bi-directional LSTM) CRF [Huang et al., 2015] to label MSRQs. To train the CRF, we extend the Constraint Driven Learning framework (CoDL) [Chang et al., 2007] to work with partially labeled instances. To overcome the challenges of operating with less training data, we supplement the model by using BERT embeddings [Devlin et al., 2019], hand-designed features, as well as a Constraint Conditional Model (CCM) to handle hard and soft constraints spanning multiple sentences. These constraints encode task-specific knowledge and help reduce the space of valid sequence label outputs for the CRF – for example, in our task each question should have at least one token indicating the '*type*' of the answer entity.

  We demonstrate the strength of our work by applying it to the novel task of answering real-world POI-recommendation questions. We train our system using just 150 fully-labeled posts and a set of 400 partially labeled posts. We find that the use of our labels helps answer 36% more questions with 35 % more (relative) Hits@3 scores, as compared to baselines such as those based on keyword based searches [Vtyurina and Clarke, 2016]. We also demonstrate how our framework can rapidly enable the parsing of MSRQs in an entirely new domain (books recommendation) with small amounts of training data and little change in the semantic representation. To the best of our knowledge we were amongst the first to apply pre-trained language models to BiLSTM CRFs and are the first to develop BiLSTM CCMs.

1.3.1.2 Task 2: QA without Intermediate Annotations

- **Problem Definition:** We introduce the novel task of answering POI-recommendation questions using knowledge present in user reviews of candidate POIs (entities). We harvest a QA dataset that contains 47,124 paragraph-sized real user questions from travelers seeking recommendations for hotels, attractions and restaurants. Each question can have thousands of candidate answers to choose from and each candidate is associated with a collection of unstructured reviews.

- **Challenge:** *Reasoning at Scale* – The dataset we created is especially challenging because commonly used neural architectures for reasoning and QA are prohibitively expensive for this task. To deal with challenges of scale, typical QA methods reduce the search space of candidates, by filtering documents using methods such as BM25 [Robertson and Zaragoza, 2009] ranking. However, such approaches do not work well on our task because review documents express opinions on similar topics; this results in documents having similar TF-IDF scores. Further, documents in our task are longer than those seen in typical QA tasks and commonly used tricks, such as arbitrarily truncating documents [Joshi et al., 2017] based on section headings, etc are not meaningful, as our documents lack structure (each document is a collection of reviews).

- **Contribution:** We harvest and release a dataset for this novel task. We present our scalable solution based on a *cluster-select-rerank* approach. It first clusters review text for each entity to identify exemplar sentences describing an entity. It then uses a scalable neural information retrieval (IR) module to select a set of potential entities from the large candidate set. A reranker uses a deeper attention-based architecture to pick the best answers from the selected entities. We find that our strategy performs better than a pure IR or a pure attention-based reasoning approach yielding nearly 25% relative improvement in Hits@3 scores over both approaches.

1.3.1.3 Task 3: Improving QA with Spatio-Textual Reasoning

- **Problem Definition:** We introduce the novel task of answering POI-recommendation questions that requires joint reasoning over knowledge present in user reviews, as well as, the geo-spatial coordinates of POIs.

- **Challenge:** *Spatio-Textual Reasoning* – Joint reasoning over spatial and textual data for the POI-recommendation task is challenging because spatial constraints

in questions may be under-specified, ambiguous and subjective – for example they may contain constraints such as "near Times Square", "within driving distance" or "within walking distance" from a particular location, etc. The distance between a location mentioned in text and a candidate entity needs to be computed and then reasoned over in the context of the constraints. In addition, not all locations mentioned in the question are required to be reasoned over – for example, users may mention their last vacation or where they are from, etc. Lastly, questions also have other ambiguous constraints that need to be reasoned over using review documents for each entity. Since we develop our work using the dataset harvested in the previous task, the challenges of scale also apply.

- **Contribution:** We first develop a modular spatial-reasoning neural network that uses geo-coordinates of location names mentioned in a question, and of candidate answer POIs, to reason over only spatial constraints. We then combine our spatial-reasoner with a textual reasoner in a joint model and present experiments on a real world POI-recommendation task. To the best of our knowledge, this is the first model that jointly reasons over spatial and textual data. We report substantial improvements over baseline models that do not use joint spatio-textual reasoning.

1.3.2 Answering Comparison Questions

In our work on answering questions that seek to compare and contrast entities, we make the following contributions:

- **Problem Definition:** We define a novel task of automatically generating *entity comparisons* from text. Our output is a table that semantically clusters descriptive phrases about entities.

- **Challenge:** *Information Balance* – We would like users of our system to be able to acquire as much information as possible from our comparison table. We hypothesize that presenting comparisons in a way that *balances information* between the two entities being compared, would be more beneficial than simple clustering or other methods that are only *aware* of the entities while clustering. In addition, we would like the aspects for comparison to be dependent on the pair of entities being compared – for instance, comparing two ancient roman cities may require more finer clusters as opposed to when an ancient roman city is compared with a city famous for winter sports.

- **Contribution:** We present comparisons between entities in a tabular format, in which each row denotes an aspect of comparison and the entries are textual phrases describing each entity for that aspect. Our system uses an information extraction pipeline followed by a novel clustering algorithm that tries to balance the amount of information from entities in each aspect to generate meaningful comparisons. We conduct user studies to demonstrate the effectiveness of our system and we find that entity-balanced clusters are overwhelmingly preferred by users. We also find that users acquire as much information about the entities, by using our comparison tables, as they do using articles.

1.4 Thesis Outline

In the next chapter we discuss some background and related work that may be useful to review before reading the thesis. The rest of the thesis is organized as follows: Chapters 3, 4 and 5 describe our work on the three aspects of answering recommendation questions. Chapter 6 presents our work on answering comparison questions. We conclude the thesis and discuss possible directions for future work, in Chapter 7.

Chapter 2

Background & Related Work

Background: We begin this chapter by presenting an introductory review of some recent methods for modeling textual sequences (Section 2.1). We then present background on a discriminative machine learning model called Conditional Random Field (CRF), which we use for labelling textual sequences (Section 2.1.3). We then also include some preliminary reading on topic models that may be helpful to review, in Section 2.2.
Related Work: Due to the challenging nature of language processing, the problem of Question-Answering (QA) has been studied using a variety of specialized tasks, each of which help investigate a different aspect of QA. Depending on the problem being studied, QA tasks may make assumptions about the nature of answers (eg: spans in text [Rajpurkar et al., 2018], documents as answers [Nguyen et al., 2016], multiple choice selection [Lai et al., 2017]), or about the knowledge source being used – for example, the use of a backend DB system [Basik et al., 2018] or a paragraph of text as knowledge [Rajpurkar et al., 2018]. We therefore also include a brief survey of recent QA tasks in Section 2.3 along with a brief introduction to attention in neural models for QA in Section 2.3.3.

2.1 Sequence Modeling and Tagging

Deep learning based methods for QA need to create representations of text – such representations may be created by representing words using sparse vectors such as those based on TF-IDF weights, or by using dense vectors learnt from a neural network. Word2Vec [Mikolov et al., 2013] is a commonly used method to initialize dense representations for words. The neural network learns word representations by using a large collection of text in which, for a given word, the network learns to predict the surrounding bag-of-words within a fixed context window (skip-gram architecture) or it could also use the surrounding context bag of words to predict the current word (CBOW architecture).

Other methods for creating word vector representations include using factorization of word co-occurrence matrices [Pennington et al., 2014].

A major limitation of such word-vector architectures is that they do not take into account, the sequential relationship of words in text. Recurrent Neural Network (RNN) architectures [Rumelhart et al., 1986], including variants such as Long-Term Short-Term Memory (LSTM) [Hochreiter and Schmidhuber, 1997] and Gated Recurrent Units (GRU) [Cho et al., 2014], are designed to overcome this limitation. Such architectures are commonly used to encode text (sequences) in deep learning models, though recently, methods based on Transformer architectures [Vaswani et al., 2017] have been shown to perform better than RNN based architectures. In this section we present a brief introduction to both, RNNs as well as, Transformer based methods for encoding text.

2.1.1 Recurrent Neural Networks

Recurrent neural networks (RNNs) [Rumelhart et al., 1986] are designed to model sequential data – in the case of text, an RNN can be used to a model a sentence by viewing it as a temporal sequence of words. At each time-step t the RNN computes the output o_t given by a function $f(w_t, o_{t-1}; \theta)$ where w_t is a representation of the input word at t, o_{t-1} is the output at the previous time step and θ are parameters. However, such an RNN can be limited in terms of its representational power because at each time-step the RNN only has access to the state that was output previously. Thus in practice, RNNs use additional states called *hidden* states (h_t) that also accumulate information seen till time step t.

Each node in an RNN with hidden states computes the following:

$$a_t = b + \mathbf{W}h_{t-1} + \mathbf{U}w_t \tag{2.1}$$

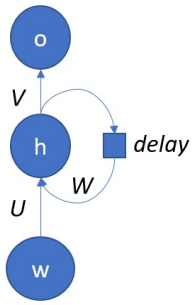


Figure 2.1: Rolled computational graph depicting an RNN. Figure adapted from [Goodfellow et al., 2016].

$$h_t = tanh(a_t) \tag{2.2}$$

$$o_t = c + \mathbf{V}h_t \tag{2.3}$$

where parameters $\mathbf{W}$, $\mathbf{U}$ and $\mathbf{V}$ are weight matrices, b and c are bias terms. The initial hidden state is usually initialized to zero or may be randomly initialized. The rolled computational graph of an RNN is depicted in Figure 2.1, where the delay circuit adds a delay of one time step. RNNs can also be extended to have multiple layers or bi-directionally encode sentences.

One of the weaknesses of RNNs is the problem of "vanishing gradients" while training. This arises due to exponentially smaller weights given to long term interactions (as compared to short-term ones) because of repeated multiplications from the chain-rule of differentiation.

2.1.1.1 Long-term Short-Term Memory (LSTM)

Long-term Short-Term Memory (LSTM) cells [Hochreiter and Schmidhuber, 1997] are one of the solutions to overcome the problem of vanishing gradients. Each node in an RNN network is modeled using LSTM cells which internally contain multiple *gates*, including those that allow the input to flow through unchanged. This helps ameliorate the problem of vanishing gradients.

An LSTM cell contains three gates: (i) Forget Gate (G_F) (ii) Input Gate (G_I) (iii) Output Gate (G_O) which are represented as vectors. All gate vectors G_g ($g \in \{I, O, F\}$) can be represented as:

$$G_g^{(t)} = \sigma(\mathbf{W_g}w_t + \mathbf{U_g}h_{t-1} + b_g) \tag{2.4}$$

where σ is the sigmoid function, $\mathbf{W_g}$, $\mathbf{U_g}$ are gate-type specific weight matrices, b_g is a gate specific bias term, w_t is the current input vector and h_{t-1} is the previous output vector.

Each LSTM cell maintains its own state s_t which is updated using the values of the forget gate vector $G_F^{(t)}$ and the input gate vector $G_I^{(t)}$ at time step t. The current cell state vector s_t is given by:

$$s_t = G_F^{(t)} \odot s_{t-1} + G_I^{(t)} \odot (\mathbf{W}w_t + \mathbf{U}h_{t-1} + b) \tag{2.5}$$

where $\odot$ denotes element-wise multiplication, s_{t-1} the previous cell state vector, h_{t-1} is the previous output vector, w_t is the current input vector, $\mathbf{W}$ and $\mathbf{U}$ are weight matrices and b is a bias term.

Lastly, the output vector h_t of an LSTM cell is given by:

$$h_t = G_O^{(t)} \odot tanh(s_t) \tag{2.6}$$

where $G_O^{(t)}$ is the output gate vector. LSTMs have been shown to model long range dependencies better than vanilla-RNNs [Goodfellow et al., 2016]. We use LSTMs in Chapter 3 to encode questions.

2.1.1.2 Gated Recurrent Units (GRU)

A simplification of the LSTM cell results in a Gated Recurrent Unit (GRU) [Cho et al., 2014] cell, where the primary difference from an LSTM cell is that, a single gate governs both "forgetting", as well as, "updation" of the cell state. A GRU cell gate is expressed in the same form as Equation 2.4. However, GRUs only use two gates: (i) Update Gate (G_U), (ii) Reset Gate (G_R), and does not use cell states.

The equation of the output state of a GRU cell is given by:

$$h_t = (1 - G_U^{(t)}) \odot h_{t-1} + G_U^{(t)} \odot tanh(\mathbf{W}w_t + \mathbf{U}(G_R^{(t-1)} \odot h_{t-1}) + b) \tag{2.7}$$

$\mathbf{W}, \mathbf{U}$ are weight matrices, b is a bias term, $G_U^{(t)}$ is the update gate vector, $G_R^{(t)}$ is the reset gate vector at time step t. By using the reset and update gates, the network can learn to ignore parts of the state vectors. GRUs are faster to train as compared to LSTMs due to fewer parameters and their performance is often comparable to that of LSTMs. RNNs including both GRUs and LSTMs, also have bi-directional variants in which sequences are encoded in both front-to-back and back-to-front directions. In such cases, the hidden/output states of encodings from both directions are concatenated at appropriate sequence positions (output state from the forward encoding at position i is combined with the backward encoding of position $\mathrm{T} - i$, where T is the length of the sequence). We extensively use bi-directional LSTMs (BiLSTM) in chapter 3, and bi-directional GRUs in chapters 4 and 5 while encoding questions.

RNNs are trained by defining loss functions which score the output values of the network, against ground-truth values. The parameters of the network are updated using gradient based methods to minimize the loss – this involves unfolding the temporal graph of the RNN (a process referred to as '*back propagation through time*' – BPTT). The exact objective used depends on the task at hand – for instance, in sequence classification tasks, the output state of the RNN may be used to generate a vector indicating the probabilities of different classes; the objective in this case would minimize the cross-entropy between the predicted class distribution and the ground-truth distribution.

2.1.2 Transformer Networks

As discussed previously, RNNs, including those based on LSTMs and GRUs, use hidden states from the previous time step while computing the output at the current state. However, a recently introduced class of networks called Transformers [Vaswani et al., 2017], models sequences using *attention* blocks. These attention blocks compute attention-weights for each position in a sequence based on every other position in the sequence and can be parallelized for computation. Networks employing transformers have established new state-of-the-art results on a variety of Question-Answering tasks [Devlin et al., 2019].

A transformer encoder network consists of stacked "blocks" that consist of a self-attention layer and a feed-forward layer. Let $\mathbf{X}$ be the input embedding matrix, i.e. a matrix consisting of vectors representing each token of a sentence in embedding space. For the purpose of this section, we assume that this input matrix also internally encodes positions of tokens. The self-attention layer first creates three matrices $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ called the *query, key and value* matrices respectively. These are created using simple linear transformations applied over the input $\mathbf{X}$ and use weight matrices $\mathbf{W}^Q, \mathbf{W}^K, \mathbf{W}^V$, each corresponding to the query, key and value respectively.

The self-attended representation $\mathbf{Z}$ for input $\mathbf{X}$ is given by:

$$\mathbf{Z} = \frac{(\mathbf{Q}\mathbf{K}^T)\mathbf{V}}{\sqrt{dim_k}} \tag{2.8}$$

where the dim_k is the dimension of the key vectors used. The self-attended representation is further transformed using a position-wise feed-forward network as expressed below:

$$FF(z) = max(0, \mathbf{W_1}z + b_1) + \mathbf{W_2} + b_2 \tag{2.9}$$

where $FF(z)$ takes a self-attended representation of a token generated in $\mathbf{Z}$ using Equation 2.8, $\mathbf{W_1}, \mathbf{W_2}$ are weight matrices and b_1, b_2 are bias terms. This gives the final encoded output from one block. As mentioned previously, a transformer stacks multiple such blocks to encode an input sequence.

In practice, transformer networks also employ multiple self-attention layers called "heads". The output of each such layer is combined, and then transformed, by a linear operation, to get the overall self-attended representation. Further, encoder blocks also employ residual connections and layer normalization which we have omitted for brevity.

Question Answering tasks frequently use pre-trained transformer based models, such as BERT [Devlin et al., 2019]. BERT has been pre-trained using two tasks – (i) the

task of *masked* language modeling (MLM), where approximately 15% of the tokens in the input sequence have been masked and the network attempts to predict the actual token, (ii) Next Sentence Prediction (NSP), where the network classifies as a pair of two sentences based on whether they were likely to be consecutive sentences in a larger document. The model uses special tokens to indicate sentence boundaries and employs task-specific top-layers while training. The parameters of the transformer blocks are available as pre-trained models. Pre-trained BERT is typically used in the bottom layers of task-specific QA networks and its use has helped improve QA systems. We use BERT in Chapters 3 and 5. Another pre-trained transformer model used in this thesis is the Universal Sentence Encoder [Cer et al., 2018]. It is trained using multi-task learning where a single transformer encoder is used for tasks based on language modeling [Kiros et al., 2015], conversation generation [Henderson et al., 2017], sentence classification and natural language inference [Bowman et al., 2015]. This trained encoder is used to generate sentence representations in Chapter 4.

2.1.3 Conditional Random Fields for Sequence Tagging

Given a pair of vectors, x, y corresponding to the input and output-label sequence respectively, a Conditional Random Field (CRF) [Lafferty et al., 2001] models the probability distribution of $P(y|x)$.

To model CRFs we define *feature functions* $\phi_k(y_t, y_{t-1}, x)$, $k = 1 \ldots \mathrm{K}$, where y_t is the label of the t^{th} position of sequence y. For example, in case of text (sentences) a feature function ϕ_k could encode features such as capitalization of a word at position t, with corresponding transitions in the output label between y_{t-1} and y_t. Multiple features functions may thus be defined and these functions could even be generated, using outputs of other networks, such as an LSTM [Huang et al., 2015]. The graphical model representation of a linear-chain CRF is shown in Figure 2.2.

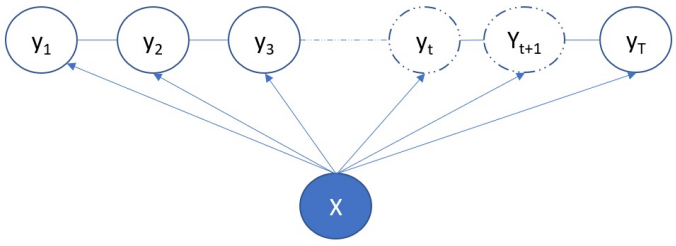


Figure 2.2: Conditional Random Field depicted as a graphical model

The expression for $p(y|x)$ is given by:

$$p(y|x) = \prod_{t=1}^{T} \frac{exp(\sum_k = 1^K \omega_k \phi_k(y_t, y_{t-1}, x_t))}{Z(x)} \tag{2.10}$$

where ω_k is the weight associated with feature function ϕ_k, and $Z(x) = \sum_y exp(\sum_k^K \omega_k \phi_k(y_t, y_{t-1}, x_t))$ and is a normalization term used to convert the feature functions to a probability distribution. It is also referred to as the *partition* function. The conditional log likelihood (LL) of a CRF is thus given by:

$$\sum_{i=1}^{|\mathbb{M}|} log(P(y^{(i)}|x^{(i)})) = \sum_{i=1}^{|\mathbb{M}|} \sum_{t=1}^{\mathrm{T}} \sum_{k=1}^{K} \omega_k \phi_k(y_t^{(i)}, y_{t-1}^{(i)}, x_t^{(i)})) - \sum_{i=1}^{|\mathbb{M}|} log(Z(x^{(i)})) \tag{2.11}$$

where $x^{(i)}, y^{(i)}$ are the i^{th} input-output sequence pairs in the training set of size $|\mathbb{M}|$.

We determine the labels of an unseen sequence by computing $y = argmax_y p(y|x)$ using the Viterbi Algorithm [Forney, 1973]. Note that since we compute the $argmax$ we can ignore the partition function while computing $p(y|x)$ (Equation 2.10).

Conditional Random Fields with LSTMs: Instead of using manually defined feature functions, CRFs can also utilize embeddings from LSTMs to create input feature functions. In the case of text sequences, given an output sequence label set $\mathbb{Y}$,and the LSTM output vectors h_t corresponding to each word x_t, LSTM-CRFs define the emission scores by mapping h_t to $\mathbb{R}^{|\mathbb{Y}|}$.CRFs are used along with bi-directional LSTMs and BERT in Chapter 3.

Parameter Estimation: The parameters of a CRF are estimated by gradient based methods such as SGD [Bottou, 2010]. The partial derivatives of the log-likelihood expression in Equation 2.11 are given by:

$$\frac{\delta}{\delta_{\omega_k}}(LL) = \sum_{i=1}^{|\mathbb{M}|} \sum_{t=1}^{T} \phi_k(y_t^{(i)}, y_{t-1}^{(i)}, x_t^{(i)}) - \sum_{i=1}^{|\mathbb{M}|} \sum_{t=1}^{T} \sum_{y,y'} \phi_k(y, y', x_t^{(i)}) p(y, y')|x^{(i)}) \tag{2.12}$$

The partition function as well as the second term in Equation 2.12 can be computed by running the 'Forward Algorithm' [Stratonovich, 1965, Koller and Friedman, 2009], which is a dynamic programming based algorithm that stores and reuses repetitive computations.

2.2 Topic Modeling

A topic model is a statistical model for discovering latent topics in a large collection of text. Intuitively, the models rely on the assumption that words belonging to a particular topic are likely to co-occur more frequently, and therefore, a topic can be visualized as a probabilistic distribution over words, implying that certain words are more likely to be associated with a topic than others. We begin with a review of non-probabilistic methods for topic models followed by Latent Dirichlet Allocation [Blei et al., 2003].

2.2.1 LSI: Latent Semantic Indexing

One of the initial attempts at modeling topics in documents was using a method called Latent Semantic Analysis [Deerwester et al., 1990] (also known as Latent semantic Indexing). It uses a term-document matrix (**X**) where each cell in the matrix denotes the frequency count of a term in the document. It then uses Singular Value Decomposition (SVD) to reduce dimensionality of the matrix as shown in Figure 2.3. The Term document matrix **X** is a product of the matrices $\mathbf{TSD}^T$ where **T** is a *words* x *dims* sized matrix, **S** is a diagonal matrix and **D**= *dims* x *documents* sized matrix.

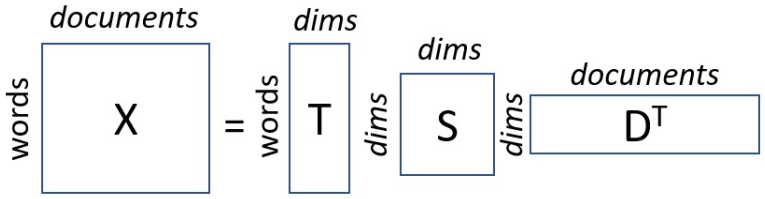


Figure 2.3: Latent Semantic Indexing

The dimensions effectively model the "topics" in the collection, with the size of the diagonal matrix being used to control the number of topics discovered. Thus, LSI captures semantic relationships between words. However, it does not give a probabilistic interpretation to the topics.

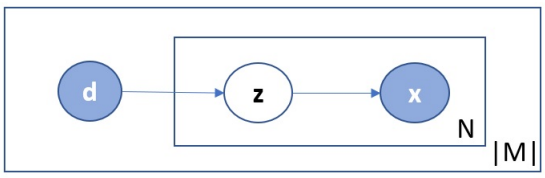


Figure 2.4: Probabilistic Latent Semantic Indexing – the figure uses *Plate Notation* to depict the graphical model. Here, 'circles' correspond to random variables and the 'rectangles' (plates) correspond to repetitions of random variables. Shaded circles denote observed variables while un-shaded circles denote unobserved (latent) variables.

2.2.2 pLSI: Probabilistic Latent Semantic Indexing

In this model each word in a document is a sample from a mixture model [Hofmann, 1999] as shown in Figure 2.4. Each word is generated from a single topic, and thus different words in a document may be generated from different topics. A document is reduced to a probability distribution over topics. Let the set of documents $\mathbb{D}$ be $\{d_1 \dots d_{|\mathbb{M}|}\}$, let the set of words $\mathbb{X}$ be $\{x_1 \dots x_N\}$ and let the set of topics ($\mathbb{Z}$) be $\{z_1 \dots z_K\}$. The generative process of the pLSI model (Figure 2.4) is as follows :

- Choose a document d with probability $p(d)$
- Choose a topic z with probability $p(z|d)$
- Choose a word x with probability $p(x|z)$

The joint distribution of documents ($\mathbb{D}$) and words ($\mathbb{X}$) in a corpus can be given by:

$$P(\mathbb{D}, \mathbb{W}) = \prod_{d=1}^{|\mathbb{M}|} \prod_{x=1}^{N} p(d) \sum_{z=1}^{K} p(z|d)\ p(x|z) \tag{2.13}$$

where $|\mathbb{M}|$ is the number of documents in the corpus, N is the number of words in a document and K is the number of topics. The parameters of this model can be estimated using Expectation-Maximization (EM).

The **E-Step** to compute the posterior probability is given by:

$$p(z|x, d) = \frac{p(x|z)\ p(z|d)}{\sum_z p(x|z)\ p(z|d)} \tag{2.14}$$

The **M-Step**, which updates parameters of the multinomial distributions for $p(x|z)$ and $p(z|d)$, using the value of $p(z|x, d)$ from the **E-Step** is given by:

$$p(w|z)^{new} = \frac{\sum_{d=1}^{|\mathbb{M}|} n(d, x)\ p(z|x, d)}{\sum_{d=1}^{M} \sum_{x=1}^{N} n(d, x)\ p(z|x, d)} \tag{2.15}$$

$$p(z|d)^{new} = \frac{\sum_{x=1}^{N} n(d, x)\ p(z|x, d)}{\sum_{z=1}^{K} \sum_{x=1}^{N} n(d, x)\ p(z|x, d)} \tag{2.16}$$

where, $n(d, x)$ denotes the number of times word x occurs in a document d and $p(d) = \frac{1}{|\mathbb{M}|}$.

2.2.3 Latent Dirichlet Allocation

The LDA model [Blei et al., 2003] is an improvement of the pLSI model – the model uses Dirichlet priors for the generation of documents. This makes the number of parameters independent of the number of documents and it also allows it to model unseen documents.

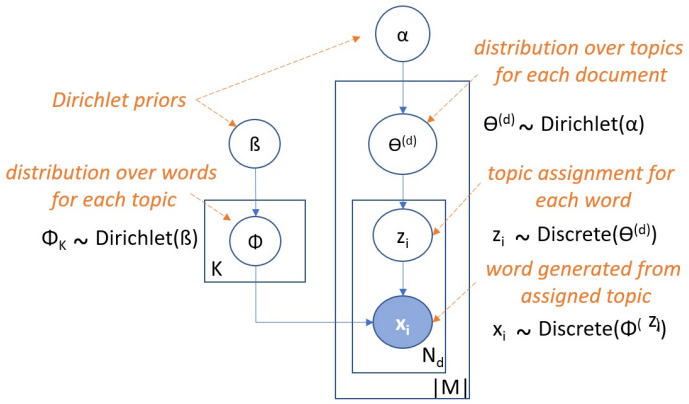


Figure 2.5: Latent Dirichlet Allocation

Given Dirichlet priors α and β, the generative process of the LDA model (Figure 2.5) is as follows:

- Choose $\Theta^{(d)} \sim Dirichlet(\alpha)$.
- Choose $\Phi_k \sim Dirichlet(\beta)$.
- For each word
  - Choose a topic $z_i \sim Multinomial(\Theta^{(d)})$.
  - Choose a word $x_i \sim Multinomial(\Phi_{z_i})$.

The joint distribution of the hidden and observed variables in LDA can be written as:

$$\prod_{k=1}^{K}[p(\Phi_k|\beta)\prod_{d=1}^{|\mathbb{M}|}[p(\Theta^{(d)}|\alpha)\prod_{i=1}^{N_d}p(z_i|\Theta^{(d)})p(x_i|z_i,\Phi_k)]] \tag{2.17}$$

where z_i is the topic assigned to the ith word x_i in a document d, N_d is the number of words in document d, $|\mathbb{M}|$ is the number of documents in the corpus, K is the number of topics.

2.2.3.1 Inference

The inference problem in a probabilistic model can be stated as follows: For the set of observed variables $\mathbb{X}$ and the set of unobserved/latent variables $\mathbb{Y}$, a set of parameters θ,

we would like to estimate $P(\mathbb{Y}|\mathbb{X};\theta)$. Computing this probability may be intractable and therefore one can use methods such as Monte Carlo Markov Chain (MCMC) based Gibbs sampling to sample values from this distribution and train a simpler model $Q(\mathbb{Y})$ on the sampled values. One of the drawbacks of MCMC based methods is that they can be slow and require adequate time for mixing before the samples are accurate. Thus, methods such as Variational EM which are faster can be used, but they require a careful selection of objective function that decides whether Q is a good approximation of P. Detailed derivations of the gibbs sampling equations for LDA can be found in [Wang, 2008] and a study of inference methods can be found in [Koller and Friedman, 2009, Eisner, 2011]. We use LDA in Chapter 6 to discover *descriptive phrases* for entities.

2.2.4 Gaussian Mixture Models and Clustering

While topic models specifically study the distribution of words and topics within documents, *clustering* models group related data instances based on a measure of 'relatedness' or 'similarity'. In the case of a text collection, clustering may be performed at any level of granularity – for example, to group document instances or to group sentences or words within a document.

Given a collection of data points $x_1, \ldots x_i, \ldots x_{|\mathbb{M}|}$, a Gaussian Mixture Model (GMM) models data using the joint distribution $p(x_i, z_i)$ where z_i is a latent variable (corresponding to a cluster label). The joint distribution $p(x_i, z_i)$ can be expressed as $p(z_i)p(x_i|z_i)$ and GMMs model $p(z_i)$ as a Multinomial distribution i.e, $z_i \sim \text{Multinomial}(\Phi)$, parameter Φ_j is $p(z_i = j)$, $j \in [1, K]$ where K is the number of mixtures (clusters) and $\sum_{j=1}^{K} \Phi_j = 1$. $p(x_i|z_i)$ is modeled as a Gaussian distribution with parameters μ, Σ. The log likelihood of the model is given by:

$$LL(\mu, \Sigma, \Phi) = \sum_{i=1}^{|\mathbb{M}|} [\log\, p(x_i|z_i; \mu, \Sigma) + \log\, p(z_i; \Phi)] \tag{2.18}$$

We use Expectation-Maximization (EM) to learn the parameters of the model where in the **E-Step** the posterior probability of z_i's can be computed using the existing model parameters and x_i, as given below:

$$p(z_i = j|x_i; \mu, \Sigma, \Phi) = \frac{p(z_i = j; \Phi)p(x_i|z_i; \mu, \Sigma)}{\sum_{j=1}^{k} p(z_i = j; \Phi)p(x_i|z_i; \mu, \Sigma)} \tag{2.19}$$

In the **M-Step** we update the parameters using the probability of z_i from the **E-Step**. The **M-Step** equations are written as:

$$\Phi_j^{\text{new}} = \frac{1}{M}\sum_{i=1}^{M} p(z_i = j|x_i; \mu, \Sigma, \Phi) \tag{2.20}$$

$$\mu_j^{\text{new}} = \frac{\sum_{i=1}^{M} p(z_i = j|x_i; \mu, \Sigma, \Phi)x_i}{\sum_{i=1}^{M} p(z_i = j|x_i; \mu, \Sigma, \Phi)} \tag{2.21}$$

$$\Sigma_j^{\text{new}} = \frac{\sum_{i=1}^{M} p(z_i = j|x_i; \mu, \Sigma, \Phi)(x_i - \mu_j)(x_i - \mu_j)^T}{\sum_{i=1}^{M} p(z_i = j|x_i; \mu, \Sigma, \Phi)} \tag{2.22}$$

Relationship with k-means clustering: The k-means clustering algorithm makes the assumption that z_i are known from its **E-Step** $z_i = \text{argmin}_j ||x_i - \mu_j||^2$. Thus, it does not use a probability for assignments and can be viewed as making "hard assignments" in a GMM. Thus, its **M-Step** equation is given by:

$$\mu_j^{\text{new}} = \frac{\sum_{i=1}^{M} \mathbb{1}\{z_i = j\}x_i}{\sum_{i=1}^{M} \mathbb{1}\{z_i = j\}} \tag{2.23}$$

The covariance matrix parameters are not updated in k-means clustering as it assumes a diagonal covariance matrix with constant variance on the diagonal for all Gaussians. We use k-means clustering extensively in this thesis and we use Gaussian Mixture Models in Chapter 6.

2.3 Question Answering Tasks

The recent developments in deep learning methods has led to rapid progress in the field of Question-Answering. This includes development of systems for conversation agents [Choi et al., 2018, Reddy et al., 2018], machine reading and comprehension [Rajpurkar et al., 2018, Abujabal et al., 2019], fusing information from structured and unstructured knowledge sources [Sun et al., 2018], solving math word problems and algebraic questions [Amini et al., 2019, Hopkins et al., 2019], etc. We present a brief overview of Question-Answering tasks, organized based on the nature of knowledge used: (1) QA using structured and semi-structured knowledge sources (2) QA using unstructured text based knowledge.

2.3.1 QA using Structured & Semi-structured Knowledge

Simple factoid questions such as "*Who was the first man on the moon*?" can often be answered by entity relations in structured and semi-structured knowledge bases such as Freebase [Bollacker et al., 2008], DBPedia [Lehmann et al., 2015] and OpenIE KBs [Das

et al., 2017]. Methods developed for answering questions using knowledge bases vary – for example, systems may map questions to templatized queries [Fader et al., 2014], parse a question into arguments that can be mapped to structured knowledge base fields [Berant and Liang, 2014, Fader et al., 2013, Basik et al., 2018] or answer questions end-to-end by encoding questions, as well as constituents of the knowledge graph, into vector representations [Bordes et al., 2014b, 2015, Reddy et al., 2014, Yih et al., 2016, Xu et al., 2020, Vakulenko et al., 2019]. Methods have also been developed to address challenges arising from the need for multi-hop inference across entity relationships (Eg: *"Who was the brother of the first US president?"*, handling relationship constraints, counting and summation (Eg: "*How many planets does the Solar System have?*") [Lin et al., 2018, Lan and Jiang, 2020].

Structured knowledge could also exist in database systems and natural language queries posed to such systems (Eg: *"Return authors who have more papers than Bob in VLDB after 2000"*) need to automatically translate questions, using underlying DB-schema, into an SQL query that can be executed over a database [Pazos R. et al., 2013, Basik et al., 2018]. Recently techniques that allow NLI queries to express SQL joins in free natural text making minimal assumptions about the database schema, have also been developed [Saha et al., 2016].

Finally, conversational systems, which can be viewed as multi-turn QA systems, require dialog context resolution before answers can be returned. Such systems may be modeled using rule-driven workspaces [Moore et al., 2017] or using end-to-end models that are trained on historical conversation logs [Sutskever et al., 2014, Vaswani et al., 2017]. In some conversational QA tasks, models need to learn to make API queries to a knowledge base which returns result sets. A model may then need to refer to these result sets while generating responses in a conversation [El Asri et al., 2017, Gangi Reddy et al., 2019, Eric et al., 2020, Raghu et al., 2021] (Eg: *"Is there a flight available for the 24th?"* where additional details about *source* and *destination* may have been provided in a previous dialog turn, and the system needs to query a ticketing system before responding with an answer).

2.3.2 QA using Unstructured Knowledge

One of the earliest formulations of question-answering from unstructured knowledge is the task of returning documents for a query. Methods developed for such tasks may use sparse vector representations to encode topical relationships to help retrieve documents [Robertson and Zaragoza, 2009] or use dense vectors from neural models to encode questions and documents [Mitra et al., 2017, Mitra and Craswell, 2019, Karpukhin et al.,

2020]. Neural methods for such tasks aim to maximize a function based on the product $f(q)\ g(d)$ where f, g are functions that generate representations of the question q, and a document, d respectively. Such representations can be generated using recurrent encoders based on LSTMs/GRUs or even pre-trained transformer based models such as BERT (eg: in the Dense Passage Retriever Model [Karpukhin et al., 2020]). Training such neural networks typically involves the use of cross-entropy or max-margin based loss functions, that learn to score the correct answer-document higher than given a sampled set of incorrect answer-documents.

Recently, QA tasks that require deeper reasoning on text for answering questions have been proposed. Variants of these tasks include machine reading comprehension tasks, where answers either need to be extracted from a given passage [Rajpurkar et al., 2018], generated using information in a passage [Reddy et al., 2018] or be chosen from a set of multiple choice questions [Lai et al., 2017]. In each setting, models are trained to utilize annotations which indicate the relevant span of text that could be used to return an answer to the question. Such tasks have been developed in multiple domains with the aim of addressing challenges specific to those domains (for example, in Medical articles [Zhang et al., 2018], High School Science [Clark et al., 2016, Sachan et al., 2011], IT Technical Support [Castelli et al., 2019]). Recent models for passage based QA tasks typically predict the span of answer text from passages by jointly encoding both, question and passage using transformer networks, and then classifying token positions of the passage to indicate the beginning and end of answer-spans.

While most such QA tasks assume that the passage (or document) required for answering a question is known, some tasks also require that the relevant passage be retrieved. Models developed for such tasks study different flavours of the problem – for example, when gold passage annotations are unavailable [Nguyen et al., 2016, Dunn et al., 2017, Joshi et al., 2017], not having access to answer-span annotations [Nguyen et al., 2016], or when answers may need to be returned after multi-hop or joint-reasoning over one or more passages [Yang et al., 2018]. Specialized QA tasks for common-sense reasoning [Abujabal et al., 2019, Huang et al., 2019], entailment from text [Saeidi et al., 2018], scripts and game-show formats [Iyyer et al., 2014] have also been developed.

A widely studied area of unstructured QA relates to Community Question-Answering – tasks that involve questions posted on web-forums along with a conversation-thread consisting of web-user responses. QA problems on such data include finding similar questions so that existing user-answers can be re-used [Hoogeveen et al., 2018a], finding unanswered questions so that users can be asked to provide answers [Deepak et al., 2017], answer questions using product descriptions and discussions/reviews [Gupta et al., 2019], etc.

Another specialized class of QA relates to the problem of answering mathematical word problems where answers need to be synthesized in symbolic form before they are computed [Amini et al., 2019, Hopkins et al., 2019]. Other tasks with numerical computation involve comparisons and simple arithmetic, based on information provided in unstructured text [Dua et al., 2019, Ran et al., 2019]. Recent models developed for numerical reasoning tasks such as NAQANet [Dua et al., 2019] and NumNet [Ran et al., 2019] reason over the explicit mentions of numerical quantities within a question or passage, by building Graph Neural Networks [Zhou et al., 2018] that represent those quantities as nodes. Nodes may be represented with with contextual vector embeddings of those mentions and edges could encode 'type' information, as well as $>$,$=$,$<$ relations.

Lastly, conversational QA tasks relying on unstructured knowledge have also been defined. These tasks mimic their non-conversational counter-parts involving machine reading for span-based answers [Choi et al., 2018], paraphrases [Reddy et al., 2018] or could even require generating follow-up questions before an answer can be returned [Saeidi et al., 2018].

To conclude, the problem of Question-Answering has been studied using multiple specialized tasks depending on the nature of knowledge, style of answering, the domain as well as, its application. While these tasks have played a crucial role in improving the state-of-art in Question-Answering, many real-world problems remain challenging and unsolved – this includes the task of answering recommendation and comparison questions, studied in this thesis.

2.3.3 Attention in Neural Question Answering

Typical architectures for neural QA create vector representations for questions (q), documents (d) and use an answering function $f(q, d)$, specific to the QA task. For instance, in case of span-based answering, f may return the start and end positions of an answer-span from a document. In other tasks, such as those based on document retrieval, $f(q, d)$ may return a relevance score. Questions and documents may be encoded using LSTMs, and the final output state of the LSTMs could be used to create the corresponding vector representations. However, in practice, when using architectures such as LSTMs, the final output-state is not very effective for representing the full sequence. Therefore, additional network layers that *attend* over the output states at each time-step are used to generate an overall representation.

In general, an attention function $f_{att}(h_1 \dots h_T)$ applied over the hidden states h_t of an LSTM (or any similar RNN) returns attention weights $a_1 \dots a_T$ for each output state.

These are used to generate an attended vector representation $\hat{h}$ of the sequence:

$$\hat{h} = \sum_{i=1}^{T} a_i h_i \tag{2.24}$$

There are many methods for attending on sequences – for instance, intra-attention [Cheng et al., 2016] uses the hidden state matrix $\mathbf{H}$ (where the t^{th} hidden state is represented by h_t) to generate an attention matrix $\mathbf{A}$ as follows:

$$\mathbf{A} = \text{softmax}(v_a \text{tanh}(\mathbf{W}_a \mathbf{H}^T)) \quad \text{and} \quad \hat{h} = \mathbf{AH} \tag{2.25}$$

where $\mathbf{A}$ is the attention matrix, $\mathbf{W}_a$ and v_a are attention parameters.

Instead of creating independent self-attended representations of questions and documents, it is often helpful to create representations that are *aware* of the other [Bahdanau et al., 2015, Luong et al., 2015]. For instance, one could create a question-aware document representation d_q using multiplicative attention [Luong et al., 2015]:

$$\mathbf{A}_d = \text{softmax}(q \mathbf{W}_E \mathbf{H}_d^T) \quad \text{and} \quad \hat{d}_q = \mathbf{A}_d \mathbf{H}_d \tag{2.26}$$

where q is a representation of the question, $\mathbf{H}_d$ is hidden state matrix for the document, $\mathbf{A}_d$ is the question-aware attention matrix and $\mathbf{W}_E$ is a parameter matrix. Alternatively one could also apply additive attention [Bahdanau et al., 2015] as given by:

$$\mathbf{A_d} = \text{softmax}(v_a \text{tanh}(\mathbf{W}_q q + \mathbf{W}_d \mathbf{H_d}^T)) \quad \text{and} \quad \hat{h} = \mathbf{A_d H_d} \tag{2.27}$$

where q is a representation of the question, v_a is an attention parameter vector, $\mathbf{H}_d$ is hidden state matrix for the document, $\mathbf{A}_d$ is the question-aware attention matrix and $\mathbf{W}_q$, $\mathbf{W}_d$ are parameter matrices. There are many variants of attention including key-value attention [Daniluk et al., 2017], which uses related ideas of key and value vectors as in Equation 2.8 [Vaswani et al., 2017], skim-attention [Yu et al., 2017] which uses reinforcement learning to skip and attend over certain portions of text. In practice, QA systems often generate multiple representations using different 'attention-heads' [Vaswani et al., 2017] or different methods of attention which are combined together during the answering task [Seo et al., 2016].

We do not include a detailed survey of attention methods or neural architectures for QA as this would be beyond the scope of this thesis. We use attention-based representations of text in Chapters 4 and 5.

Part II

Recommendation Questions

Chapter 3

QA with Intermediate Annotations

In this chapter, we introduce the novel task of *understanding* and *answering* recommendation questions. Specifically, we focus our attention on multi-sentence *entity-seeking* recommendation questions (MSRQs), i.e., questions that expect one or more entity recommendations as answer. In the case of the tourism domain, such entity-answers may be Points-of-Interest (POI), within a city. Figure 3.1 shows an example MSRQ from a tourism forum,[1] where the user is interested in finding a *hotel* that satisfies some constraints and preferences; an *answer* to this question is thus the name of a hotel (entity) which needs to satisfy some properties such as being a 'budget' option. A preliminary analysis of such entity-seeking recommendation questions from online forums reveals that almost all of them consist of multiple sentences – they often elaborate on a user's specific circumstance before asking the actual question.

In order to understand and answer MSRQs, we first convert a question into a machine

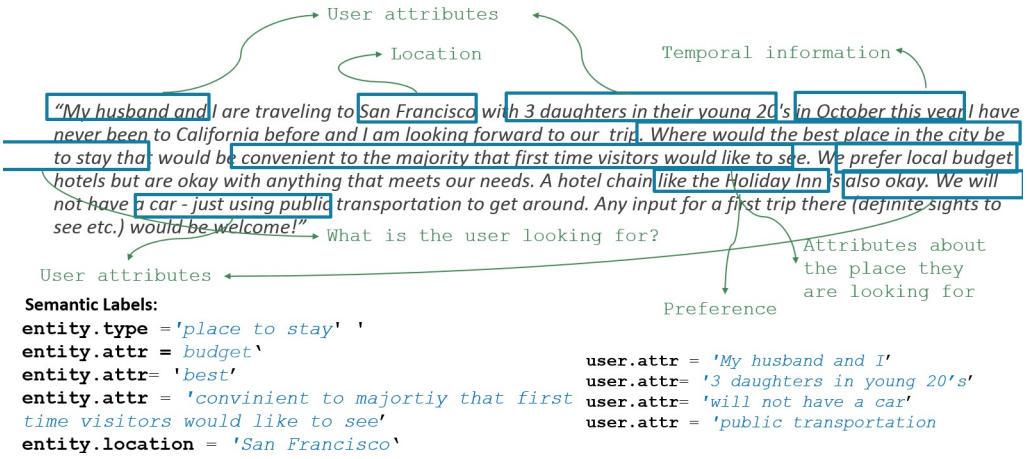


Figure 3.1: An entity-seeking MSRQ and annotated with our semantic labels

[1] http://tripadvisor.com

representation, consisting of labels identifying the *informative* portions in a question. We then use these labels, to query a downstream knowledge base, to return entity answers.

To keep our work applicable to a wide variety of domains, our machine representation does not use a domain-specific vocabulary. Instead, we design an *open* semantic representation, inspired in part by Open QA [Fader et al., 2014], in which we explicitly annotate the answer (entity) type; other answer attributes, while identified, are not further categorized. Eg., in Figure 3.1, 'place to stay' is labeled as *entity.type*, while 'budget' is labeled as an *entity.attr*. We also allow attributes of the *user* to be represented. Domain specific annotations such as *location* for tourism questions are permitted.

We pose the task of understanding MSRQs as a semantic labeling task where tokens from the question are annotated with a semantic label from our open representation. However, in contrast to related literature on semantic role labeling [Yang and Mitchell, 2017], slot filling tasks [Bapna et al., 2017] and query formulation [Wang and Nyberg, 2016, Vtyurina and Clarke, 2016, Nogueira and Cho, 2017], semantic labeling of MSRQs raise several novel challenges.

MSRQs express a wide variety of intents and requirements which span across multiple sentences, requiring the model to capture within-sentence as well as inter-sentence interactions effectively. In addition, questions can be unnecessarily belabored requiring the system to reason about what is important and what is not. Lastly, we find that generating training data for parsing MSRQs is hard due to the complex nature of the task. Thus, this requires the models to operate in low training data settings.

In order to address these challenges and label MSRQs, we use a bi-directional LSTM CRF (BiLSTM CRF) [Huang et al., 2015] as our base model and extend it in three ways. First, we improve performance by inputting contextual embeddings from BERT [Devlin et al., 2019] into the model. We refer to this configuration as BERT BiLSTM CRF. Second, we encode knowledge by incorporating hand-designed features as well as semantic constraints over the entire multi-sentence question during end-to-end training. This can be thought of as incorporating Constrained Conditional Model (CCM)-style constraints and inference [Chang et al., 2007] in a neural model. Finally, we find that crowdsourcing complete annotations is hard, since the task is complex. In this work, we are able to improve training by partially labeled questions, which are easier to source. To the best of our knowledge, we were amongst the first to apply pre-trained language models such as BERT to BiLSTM CRFs[2] and we are the first to develop BiLSTM CCMs.

[2]contemporaneous work includes the development of such models for NER tagging [Dai et al., 2019, Souza et al., 2019]

3.1 Contributions

In summary, our work makes the following contributions:

1. We present the novel task of understanding multi-sentence entity-seeking recommendation questions (MSRQs). We define *open* semantic labels, which minimize schema or ontology specific semantic vocabulary and can easily generalize across domains. These semantic labels identify *informative* portions of a question that can be used by a downstream answering component.

2. The core of our model uses a BERT BiLSTM CRF model. We extend this by providing hand-designed features and using CCM inference, which allows us to specify within-sentence as well as inter-sentence (hard and soft) constraints. This helps encode prior knowledge about the labeling task. To the best of our knowledge we are the first to develop neural sequence tagging models that incorporate CCM constraints.

3. We present detailed experiments on our models using tourism POI-recommendation questions. We also demonstrate how crowd-sourced partially labeled questions, can be effectively used in our constraint based tagging framework, to help improve labeling accuracy. We find that our best model achieves 15pt improvement in F1 scores over a baseline BiLSTM CRF.

4. We present the novel task of answering tourism POI-recommendation questions using a web based semi-structured knowledge source. Our semantic labels help formulate a more effective query to knowledge sources and our system answers 36% more questions with 35 % higher (relative) Hits@3 scores, as compared to baselines.

5. We also demonstrate the applicability of our semantic labels for MSRQs in a new domain about book recommendations, with minimal training data.[3]

3.2 Overview

Given a multi-sentence entity-seeking recommendation question, our goal is to first parse and generate a semantic representation of the question using labels that identify *infor-*

[3]The work in the chapter was done jointly with Poojan Mehta and Barun Patra. Poojan implemented the initial scripts used for data collection and annotation. Barun contributed significantly to the development and implementation of the CCM based models. The part of the work done by both appeared in their respective B.Tech theses.

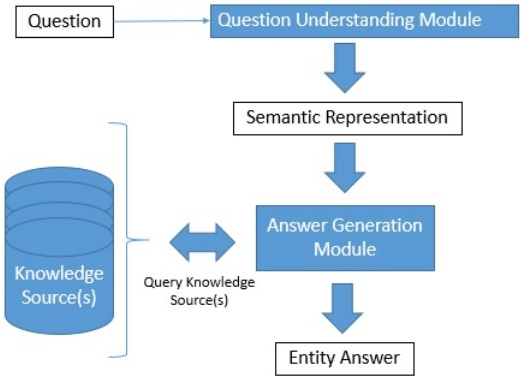


Figure 3.2: Schematic Representation of the QA system

mative portions of a question. The semantic representation of the question can then be used to return an entity answer for the question, using a knowledge source.

Thus, our QA system consists of two modules (see Figure 3.2): question understanding (MSRQ parsing), and a querying module to return entity answers. The modularized two-step architecture allows us to tackle different aspects of the problem independently. The semantic representation generated by the question understanding module is generic and not tied to a specific corpora or ontology. In this chapter we experiment with the Google Places Web collection[4] as our knowledge source. It consists of semi-structured data including geographic information, entity categories, entity reviews etc. The collection is queried using a web API that accepts an unstructured text string as query.

3.3 Related Work

To the best of our knowledge, we are the first to explicitly address the task of *understanding* multi-sentence entity-seeking questions and demonstrate its use in an answering task. There are different aspects of our work that relate to existing literature and we discuss them in this section. We begin by contrasting our work on multi-sentence question understanding and answering with recent work on question-answering (Section 3.3.1). We then include a review of related work on semantic representations of questions (Section 3.3.2) followed by a brief survey of recent literature on semantic labeling (Section 3.3.3). We conclude with a summary in Section 3.3.4.

[4]https://developers.google.com/places/web-service/intro

Question Type	Knowledge Type	Answer Type	Related Work
Single Sentence	Structured (eg. DBPedia, Freebase)	Entity	[Lukovnikov et al., 2017, Bordes et al., 2014b, 2015]
	Structured (Open IE style KBs)	Entity	[Fader et al., 2014, Berant and Liang, 2014]
	Structured + Unstructured (Open IE style KBs with supporting text passages on entities)	Entity	[Das et al., 2017]
	Structured (Databases)	Tables/ Table rows	[Saha et al., 2016, Pazos R. et al., 2013]
	Unstructured	Text Spans	[Rajpurkar et al., 2018, Trischler et al., 2017, Trivedi et al., 2017, Chen et al., 2017a, Joshi et al., 2017, Yang et al., 2018, Dua et al., 2019]
	Unstructured	Text Passages	[Vtyurina and Clarke, 2016, Wang and Nyberg, 2016, 2015]
	Multiple choice answers	Answers from specified choices	[Guo et al., 2017, Khot et al., 2017, Lai et al., 2017, Zhang et al., 2018, Welbl et al., 2018]
Multi-sentence	Unstructured	Text (Answer) passages	[Singh and Simperl, 2016, Romeo et al., 2016, Srba and Bielikova, 2016, Bogdanova and Foster, 2016]
	Unstructured (QA pairs+Wikipedia)	Entity	[Iyyer et al., 2014]
	Semi-structured meta-data + Unstructured (Entity Reviews)	**Entity**	**Our work**

Table 3.1: Related work: Question Answering

3.3.1 Question Answering Systems

There are two common approaches for QA systems – joint and pipelined, both with different advantages. The joint systems usually train an end-to-end neural architecture, with a softmax over candidate answers (or spans over a given passage) as the final layer [Iyyer et al., 2014, Rajpurkar et al., 2018]. Such systems can be rapidly retrained for different domains, as they use minimal hand-constructed or domain-specific features. But, they require huge amounts of labeled QA pairs for training.

In contrast, a pipelined approach [Fader et al., 2014, Berant and Liang, 2014, Fader et al., 2013, Kwiatkowski et al., 2013, Vtyurina and Clarke, 2016, Wang and Nyberg, 2016] divides the task into two components – question processing (understanding) and querying the knowledge source. Our work in this chapter follows the second approach.

We choose to summarize popular approaches in QA systems on the basis of: (a) type of questions they answer, (b) nature of KB/Corpus used for answering, (c) nature of answers returned by the answering system (See Table 3.1).

In our work on answering multi-sentence entity-seeking recommendation questions, we return entity-answers. The problem of returning direct, (non-document/passage) answers to questions from background knowledge sources has been studied, but primarily for single sentence factoid-like questions [Fader et al., 2014, Berant and Liang, 2014, Yin et al., 2015, Sun et al., 2015, Saha et al., 2016, Khot et al., 2017, Lukovnikov et al., 2017, Zheng et al., 2018, Zhao et al., 2019b]. Reading comprehension tasks [Rajpurkar et al., 2018, Trischler et al., 2017, Joshi et al., 2017, Trivedi et al., 2017, Yang et al., 2018, Dua et al., 2019] require answers to be generated from unstructured text also only return answers for relatively simple (single-sentence) questions.

Other works have considered multi-sentence questions, but in different settings, such as the specialized setting of answering multiple-choice SAT and science questions [Seo et al., 2015, Clark et al., 2016, Khot et al., 2017, Guo et al., 2017, Lai et al., 2017, Zhang et al., 2018], mathematical word problems [Liang et al., 2016], and textbook questions [Sachan et al., 2016]. Such systems do not return entity answers to questions. Community QA systems [Bogdanova and Foster, 2016, Shen et al., 2015, Qiu and Huang, 2015, Tan et al., 2015, Pithyaachariyakul and Kulkarni, 2018] match questions with *user*-provided answers, instead of entities from background knowledge-source. IR-based systems [Vtyurina and Clarke, 2016, Wang and Nyberg, 2016, Pithyaachariyakul and Kulkarni, 2018] query the Web for open-domain questions, but return long (1000 character) passages as answers; they have not been developed for, or tested on entity-seeking questions. These techniques that can handle multi-sentence questions [Vtyurina and Clarke, 2016, Wang and Nyberg, 2016, Pithyaachariyakul and Kulkarni, 2018] typically perform retrieval using keywords extracted from questions; these do not "understand" the questions and cannot answer many tourism questions, as our experiments show (Section 3.7). The more traditional solutions (e.g., semantic parsing) that parse the questions deeply can process only *single*-sentence questions [Kwiatkowski et al., 2013, Fader et al., 2014, Berant and Liang, 2014, Fader et al., 2013, Zheng et al., 2018].

Finally, systems such as QANTA [Iyyer et al., 2014] also answer complex multi-sentence questions but their methods, can only select answers from a small list of entities and also require large amounts of training data with redundancy of QA pairs. In contrast, the Google Places API we experiment with (as our knowledge source) has millions of entities. It is important to note that for answering an MSRQ, the answer space can include thousands of candidate entities per question, with large unstructured review documents about each entity that help determine the best answer entity. Thus, these documents are significantly longer than passages (or similar length articles) that have traditionally been used in neural QA tasks.

We discuss literature on parsing (understanding) questions in the next section.

3.3.2 Question Parsing

QA systems use a variety of different intermediate semantic representations. Most of them, including the rich body of work in NLIDB (Natural Language Interfaces for Databases) and semantic parsing, parse *single* sentence questions into a query based on the underlying ontology or DB schema, and are often learned directly by defining grammars, rules and templates [Zettlemoyer, 2009, Liang, 2011, Kwiatkowski et al., 2013, Berant et al., 2013, Yih et al., 2015, Sun et al., 2015, Saha et al., 2016, Reddy et al., 2016, Khot et al., 2017, Cheng et al., 2017, Lukovnikov et al., 2017, Abujabal et al., 2017, Zheng et al., 2018]. Work such as [Fader et al., 2014, Berant and Liang, 2014] build *open* semantic representations for single sentence questions, that are not tied to a specific knowledge source or ontology. We follow a similar approach and develop an open semantic representation for multi-sentence entity-seeking recommendation questions. Our representation uses labels that help a downstream answering component return entity answers.

Some works build neural models that represent a question as a continuous-valued vector [Bordes et al., 2014a,b, Xu et al., 2016, Chen et al., 2016, Zhang et al., 2016] but such methods, require significant amounts of training data. Other systems rely on IR and do not construct explicit semantic representations at all [Sun et al., 2015, Vtyurina and Clarke, 2016]; they rely on selecting keywords from the question for querying and as shown in our experiments do not perform well for answering multi-sentence entity-seeking questions. Work such as that by Nogueira and Cho (2017) uses reinforcement learning to select query terms in a document retrieval task and requires a large collection of document-relevance judgments.

We now summarize recent methods employed to generate semantic representations of questions.

3.3.3 Neural Semantic Parsing

There is a large body of literature dealing with semantic parsing of single sentences, especially for frames in PropBank and FrameNet [Palmer et al., 2005, Baker et al., 1998]. Most recently, methods that use neural architectures for SRL (Semantic Role Labeling) have been developed [Zhou and Xu, 2015, Xia et al., 2019b]. For instance, work by Zhou and Xu (2015) uses a BiLSTM CRF for labeling sentences with PropBank predicate argument structures, while work by He et al. (2018) relies on a BiLSTM with BIO-encoding constraints during LSTM decoding. Other related work by Yang and Mitchell (2017) proposes a BiLSTM CRF model that is further used in a graphical model that encodes SRL structural constraints as factors. Work such as [Bapna et al., 2017] uses a BiLSTM

tagger for predicting task-oriented information slots from sentences. Our work uses similar approaches for labeling (parsing) MSRQs, but we note that such systems cannot be directly used in our task due to their model-specific optimization for their label space. However, we adapt the label space of the Deep SRL system [He et al., 2017] for our task and use its predicate tagger as a baseline for evaluation (Section 3.6).

3.3.4 Summary

In summary, while related work shares aspects with our task there are three main distinguishing features that are not jointly addressed in existing work: (i) **Question Type**: A major focus of existing work has been on single sentence questions, sometimes with the added complexity arising out of entity relations and co-reference. Such questions are often posed as "which/where/when/who/what" questions. However, our work uses multi-sentence questions which can additionally contain vague expression of intents as well as information that is irrelevant for the answering task. (ii) **Knowledge**: Most information seeking questions either answer factoid-style questions from knowledge graphs and structured knowledge bases or answer them from paragraphs of text which contain explicit answers. In contrast, our work uses a black-box knowledge source that accepts a free-text query. Our querying representation makes no assumptions about the nature of the underlying knowledge store and is based on a very general semantic representation.(iii) **Answer-type**: Existing QA systems either return answer spans (reading comprehension tasks), or documents (from the web or large text collections) to fulfill a knowledge-grounded information query that relies on explicit mention (or with some degree of semantic gap) of the answer. In contrast, our QA pipeline returns entity answers from a (blackbox) web API that accepts a text string as query. The API, internally uses structured and unstructured data, including entity reviews containing subjective opinions, to return an answer.

In the next section we describe our question representation (Section 3.4) followed by details about our labeling system (Section 3.5). We present experiments in Section 3.6 and details of our answering component in Section 3.7. We then demonstrate how our labeling scheme can be reused for a different domain – for book recommendation questions (Section 3.8). We conclude the chapter in Section 3.9.

3.4 Semantic Labels for MSRQs

As mentioned earlier, our question understanding component parses an MSRQ into an *open* semantic representation. Our choice of representation is motivated by two goals.

First, for wider applicability of our work, we wish to make minimal assumptions about the domain of the QA task and therefore, minimize domain-specific semantic vocabulary.[5] Second, we wish to identify only the *informative* elements of a question, so that a robust down-stream QA or IR system can meaningfully answer it. As a first step towards a generic representation for an MSRQ, we make the assumptions that a multi-sentence question is asking only one final question, and that the expected answer is one or more entities. This precludes Boolean, comparison, 'why'/'how', and multiple part questions

We have two labels associated with the entity being sought: *entity.type* and *entity.attr*, to capture the type and the attributes of the entity, respectively. We also include a label *user.attr* to capture the properties of the user asking the question. The semantic labels of *entity.type* and *entity.attr* are generic and will be applicable to any domain. Other generic labels to identify related entities (eg: in questions where users ask for entities similar to a list of entities) could also be defined. We also allow the possibility of incorporating additional labels which are domain specific. For instance, for the tourism domain, location could be important, so we can include an additional label *entity.location* describing the location of the answer entity.

Figure 3.1 illustrates the choice of our labels with an example from the tourism domain. Here, the user is interested in finding a 'place to stay' (*entity.type*) that satisfies some properties such as 'budget' (*entity.attr*). The question includes some information about the user herself e.g., 'will not have a car' which may become relevant for answering the question. The phrase 'San Francisco' describes the location of the entity and is labeled with a domain specific label (*entity.location*).

We deliberately keep the choice of our labels simple which allows us to adapt to multiple different domains with minimal change in the representation. However, additional domain specific labels could also be helpful for downstream QA tasks, for instance, *entity.attr* for restaurants could be made more specific by adding labels for cuisine types, seating capacity, budgets, etc, if the knowledge base captures these attributes and the querying interface can consume such labels. This would be similar in spirit to works that create schema-aware query representations from free-text queries [Ochieng, 2020, Őzcan et al., 2020, Saha et al., 2016].

3.5 MSRQ Semantic Labeling

We formulate the task of outputting the semantic representation for a user question as a sequence labeling problem. Given a question q with tokens (words) $x_1 \dots x_i \dots x_T$, and

[5] Our representation can easily be generalized to include domain-specific semantic labels.

the label space $\mathbb{Y} = \{entity.attr, entity.location, entity.type\}$, we tag each word x_i with a label from $\mathbb{Y}$. Thus, there is a one-to-one correspondence between our token-level label set and the semantic labels described in Section 3.4. We utilize a BERT BiLSTM CRF for sequence labeling and as described previously, we extend the model in order to address the challenges posed by MSRQs: (a) First, we incorporate hand-engineered features especially designed for our labeling task. (b) Second, we make use of a Constrained Conditional Model (CCM) [Chang et al., 2007] to incorporate within-sentence as well as inter-sentence constraints. These constraints act as a prior and help ameliorate the problems posed by our low-data setting. (c) Third, we use Amazon Mechanical Turk (AMT) to obtain additional partially labeled data which we use in our constraint-driven framework.

3.5.1 Features

We incorporate a number of (domain-independent) features into our BERT BiLSTM CRF model where each unique feature is represented as a multi-hot vector and concatenated with the BERT embedding representation of each token. In experiments with BiLSTM CRF models without BERT, we replace the BERT embeddings with pre-trained Word2Vec [Mikolov et al., 2013] embeddings that are concatenated with the multi-hot feature embeddings.

3.5.1.1 Lexical & Token Features

We include features to indicate:

- whether a token begins with an upper-case character
- whether a token is a numeric quantity
- the part-of-speech tag of the token using the tagging scheme of the Penn Tree Bank[6]
- whether a token is a noun
- whether a token is a 'location' as identified by an off-the-shelf Named Entity Recognizer [Finkel et al., 2005].

3.5.1.2 Type and Attribute Features

Type Features: We found that questions frequently include phrases such as "*what would be the best place to …*" or "*can anyone recommend where to …*". To create features

[6]https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

POS Regex. Patterns	Examples Matched
$[WDT\|WP\|WP\$\|WRB]+\ TO\ [VB\|VBD\|VBG\|VBN\|VBP\|VPZ]+$	*"where to eat", "what to see"*
$[NN\|NNS\|NNP\|NNPS]+\ TO\ [VB\|VBD\|VBG\|VBN\|VBP\|VPZ]+$	*"place to stay", "sight to see"*

Table 3.2: Regular Expressions of POS-based patterns used to create indicator features for *entity.type* tokens. We ignore WP tags when the tag is associated with '*who*'.

indicative of *entity.type* tokens in such questions, we created simple patterns (Table 3.2) based on part-of-speech tags to identify nouns of interest, as well as, using dependency parses of sentences, to determine the nearest head-noun targets of verb phrases. The noun targets so identified are used as features for *entity.type* labels.

Attribute Features: We generate the dependency parse of sentence and mark adjective/noun tokens in the sentence with an 'attribute indicator' if any of the token ancestors in the parse tree, are identified as a potential *entity.type* (based on the type-feature described previously).

Descriptive Phrases: We use adjective-noun phrases called "descriptive phrases" [Contractor et al., 2016] and indicate them using a feature. The nouns of such descriptive phrases are often indicative of *entity.type* labels while adjectives indicate *entity.attr* labels – for example, in the phrase "*...a great restaurant ...*" the the word "*restaurant*" is a good candidate for *entity.type*. We describe the creation and extraction of descriptive phrases in more detail in Chapter 6 where they are extensively used to build tables for comparing entities.

3.5.1.3 Word Embedding features and Count-based features

Word Embedding Features: We trained Word2Vec [Mikolov et al., 2013] on a large collection of $80,000$ tourism questions and clustered words in embedding space using k-means clustering. Each word in the vocabulary was assigned a cluster-id and we use these ids as features.

Count-based features: Using counts of tokens in a question have been found to be helpful in prior work [Vtyurina and Clarke, 2016] and we therefore use question-level token-frequencies as a feature.

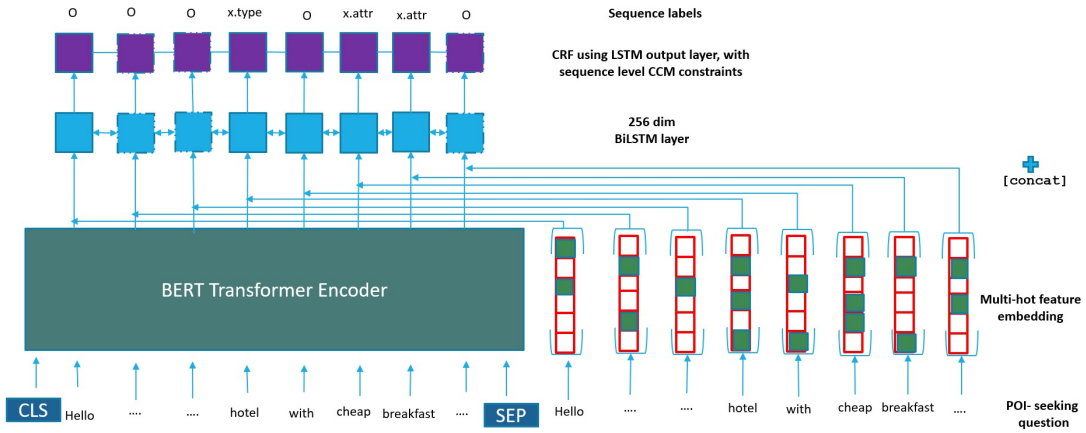


Figure 3.3: BERT BiLSTM CCM with features for sequence labeling.

3.5.2 Constraints

Since we label multiple-sentence questions, we need to capture patterns spanning across sentences. One alternative would be to model these patterns as features defined over non-adjacent tokens (labels). But this can make the modeling quite complex. Instead, we model them as global constraints over the set of possible labels.

We design the following constraints: (i) type constraint (hard): every question must have at least one *entity.type* token, (ii) attribute constraint (soft), which penalizes absence of an *entity.attr* label in the sequence, and (iii) a soft constraint that prefers all *entity.type* tokens occur in the same sentence. The last constraint helps reduce erroneous *entity.type* labels but allows the labeler, to choose *entity.type*-labeled tokens from multiple sentences only if it is very confident. Thus, while the first two constraints are directed towards improving recall, the last constraint helps improve precision of *entity.type* labels

In order to use our constraints, we employ Constrained Conditional Models (CCMs) for our task [Chang et al., 2007] which use an alternate learning objective expressed as the difference between the original log-likelihood and a constraint violation penalty:

$$\sum_i \omega^T \phi(\mathbf{x^{(i)}}, \mathbf{y^{(i)}}) - \sum_i \sum_k \rho_k \mathcal{C}_k(\mathbf{x^{(i)}}, \mathbf{y^{(i)}}) \tag{3.1}$$

Here, i indexes over all examples and k over all constraints. $\mathbf{x^{(i)}}$ is the i^{th} sequence and $\mathbf{y^{(i)}}$ is its labeling. ϕ and ω are feature and weight vectors respectively. $\mathcal{C}_k$ and ρ_k denote the violation score and weight associated with k^{th} constraint. The ω parameters are learned analogous to a vanilla CRF and the ρ parameters are computed based on counting the number of times constraints are violated in the validation set. Inference in CCMs is formulated as an Integer Linear Program (ILP) [Chang et al., 2007]. The original CCM

formulation was in the context of regular CRFs [Lafferty et al., 2001] and and we extend its use in a combined model of BERT BiLSTM CRF with CCM constraints (referred to as BERT BiLSTM CCM) that is trained end-to-end (Figure 3.3).

Specifically, let $\mathbb{Y}$ be the set of label indices.[7] Let T be the sequence length and $x_1 \cdots x_T$ be the tokens. We decompose the first term of Equation 3.1 into $\phi_{emit}(x_t)[l]$, denoting the emission scores[8] associated with input token x_t and label $l \in \mathbb{Y}$, and $\phi_{trans}[l_b, l_e]$ ($\phi_{trans} \in \mathbb{R}^{|\mathbb{Y}|\times|\mathbb{Y}|}$), denoting the transition weight matrix associated with a transition from label $l_b \rightarrow l_e$. Then

$$
\begin{aligned}
\max_{\mathbb{1}} \quad & \mathcal{C}_1(\mathbb{1}) = \sum_{l\in\mathbb{Y}} \phi_{emit}(x_0)[l]\mathbb{1}_{0,l} + \sum_{t=1}^{T-1}\sum_{l_b\in\mathbb{Y}}\sum_{l_e\in\mathbb{Y}} (\phi_{emit}(x_t)[l_e] + \phi_{trans}[l_b, l_e])\mathbb{1}_{t,l_b,l_e} \\
\text{s.t} \quad & \forall_{l\in\mathbb{Y}}\mathbb{1}_{0,l} \in \{0,1\} \\
& \forall_{t\in 1\cdots T-1}\forall_{l_b\in\mathbb{Y}}\forall_{l_e\in\mathbb{Y}}\mathbb{1}_{t,l_b,l_e} \in \{0,1\} \\
& \sum_{l\in\mathbb{Y}} \mathbb{1}_{0,l} = 1 \\
& \forall_{t\in 1\cdots T-2}\forall_{l\in\mathbb{Y}} \sum_{1_b\in\mathbb{Y}} \mathbb{1}_{t,l_b,l} = \sum_{1_e\in\mathbb{Y}} \mathbb{1}_{t,l,l_e}
\end{aligned}
\tag{3.2}
$$

defines the Viterbi decoding for a linear chain CRF. The variable $\mathbb{1}_{0,l} = 1$ if the first token of the sequence is tagged l in the optimal Viterbi sequence, and zero otherwise. Furthermore $\mathbb{1}_{t,l_b,l_e} = 1$ if the t^{th} token is tagged with label l_e and the $(t-1)^{th}$ token is tagged l_b in the optimal Viterbi sequence, and is marked zero otherwise. The last constraint ensures that the number of incoming transitions with label l equal the number of outgoing transitions from label 'l'.

Type Label Constraints (Hard): In order to model the type-based hard constraint (there has to be at least one *entity.type* label in the sequence), we add the following constraint to the optimization problem:

$$
\mathbb{1}_{0,\text{entity.type}} + \sum_{i=1}^{T-1}\sum_{l_b\in\mathbb{Y}} \mathbb{1}_{t,l_b,\text{entity.type}} \geq 1 \tag{3.3}
$$

Here, $\mathbb{1}_{0,\text{entity.type}} = 1$ if the first token is tagged as a type, while $\sum_{l_b\in\mathbb{Y}} \mathbb{1}_{t,l_b,\text{entity.type}} = 1$ if the t^{th} token is tagged as an entity.

Attribute Label Constraints (Soft): In order to model the attribute based constraint (the non existence of an *entity.attr* label in the sequence is penalized), we introduce a

[7] We overload the notation for labels and their associated indices So $l_e \in \mathbb{Y}$ denotes an index of a label, while *entity.type* $\in \mathbb{Y}$ denotes the index associated with entity.type

[8] the output of the feed-forward layer in the BiLSTM-CRF

dummy variable $\hat{d}$ for our ILP formulation. Then, given the constraint violation penalty η, we change the model optimization problem as:

$$\begin{aligned} \max_{\mathbb{1},\hat{d}} \quad & \mathcal{C}_2(\mathbb{1},\hat{d}) = \mathcal{C}_1(\mathbb{1}) - \eta \cdot \hat{d} \\ \text{s.t} \quad & \hat{d} \in \{0,1\} \\ & \mathbb{1}_{0,\text{entity.attr}} + \sum_{t=1}^{T-1} \sum_{l_b \in \mathbb{Y}} \mathbb{1}_{t,l_b,\text{entity.attr}} + \hat{d} \geq 1 \end{aligned} \tag{3.4}$$

Here, if the constraint is violated, then $\hat{d} = 1$ and the objective suffers a penalty of η. Conversely, since it is a minimization over $\hat{d}$ as well, if the constraint is satisfied, then $\hat{d} = 0$ and the objective is not penalized. The ILP constraints from Equation 3.3 and Viterbi decoding also continue to apply.

Inter-sentence Type Constraint: We model the constraint that all *entity.type* labels should appear in a single sentence. We implement this as a soft constraint by imposing an $L1$ penalty on the number of sentences containing an *entity.type* (thereby insuring that fewer sentences contain type labels). Let ν_p denote the index of the start of the p^{th} sentence, such that $\{w_r; \nu_p \leq r < \nu_{p+1}\}$ are the tokens in the p^{th} sentence (note that $\nu_0 = 0$). Let the list of sentences be $\mathbf{S}$ and we define indicator variables $z_1, \ldots z_p \ldots z_{|\mathbf{S}|}$, with $z_p = 1$ if the p^{th} sentence contains a type. Let η_2 be the associated penalty. We modify the optimization problem then as follows:

$$\begin{aligned} \max_{\mathbb{1},\hat{d},\mathbb{Z}} \quad & \mathcal{C}_2(\mathbb{1},\hat{d}) - \eta_2 \cdot (\sum_p z_p) \\ \text{s.t} \quad & \forall_p z_p \in \{0,1\} \\ & \forall_p \forall_{r,\nu_p \leq r < \nu_{p+1}} z_p - \sum_{l_b} \mathbb{1}_{r,l_b,\text{entity.type}} \geq 0 \end{aligned} \tag{3.5}$$

Here, the variable r indexes over the tokens for the p^{th} sentence. $\sum_{l_b} \mathbb{1}_{r,l_b,\text{entity.type}} = 1$ if the r^{th} token is a type, and is 0 otherwise (thus, the last constraint is defined for each word in a sentence and it is satisfied only if the p^{th} sentence has a word with an *entity.type* label). Hence if any of the tokens in the p^{th} sentence is labeled a type, $z_p = 1$ – Note that combined with Equation 3.3, we also have $\sum_p z_p \geq 1$.

Full ILP formulation: The full ILP problem is summarized below:

$$
\begin{aligned}
\max_{\mathbb{1},\hat{d},\mathbb{Z}} \quad & \sum_{l\in\mathbb{Y}} \phi_{emit}(x_0)[l]\mathbb{1}_{0,l} + \sum_{t=1}^{T-1}\sum_{l_b\in\mathbb{Y}}\sum_{l_e\in\mathbb{Y}}(\phi_{emit}(x_t)[l_e] + \phi_{trans}[l_b,l_e])\mathbb{1}_{t,l_b,l_e} - \eta\cdot\hat{d} - \eta_2\cdot(\sum_p z_p) \\
\text{s.t} \quad & \forall_{l\in\mathbb{Y}}\mathbb{1}_{0,l} \in \{0,1\} \\
& \forall_{t\in 1\cdots T-1}\forall_{l_b\in\mathbb{Y}}\forall_{l_e\in\mathbb{Y}}\mathbb{1}_{t,l_b,l_e} \in \{0,1\} \\
& \sum_{l\in\mathbb{Y}}\mathbb{1}_{0,l} = 1 \\
& \forall_{t\in 1\cdots T-2}\forall_{l\in\mathbb{Y}}\sum_{1_b\in\mathbb{Y}}\mathbb{1}_{t,l_b,l} = \sum_{1_e\in\mathbb{Y}}\mathbb{1}_{t,l,l_e} \\
& \mathbb{1}_{0,\text{entity.type}} + \sum_{i=t}^{T-1}\sum_{l_b\in\mathbb{Y}}\mathbb{1}_{t,l_b,\text{entity.type}} \geq 1 \\
& \hat{d} \in \{0,1\} \\
& \mathbb{1}_{0,\text{entity.attr}} + \sum_{t=1}^{T-1}\sum_{l_b\in\mathbb{Y}}\mathbb{1}_{t,l_b,\text{entity.attr}} + \hat{d} \geq 1 \\
& \forall_p z_p \in \{0,1\} \\
& \forall_p\forall_{r,\nu_p\leq r<\nu_{p+1}}\ z_p - \sum_{l_b}\mathbb{1}_{r,l_b,\text{entity.type}} \geq 0 \\
& \sum_p z_p \geq 1
\end{aligned}
\tag{3.6}
$$

3.5.3 Partially labeled data

Data Collection: In order to obtain a larger amount of labeled data for our task, we make use of crowd-sourcing (Amazon Mechanical Turk). Since our labeling task can be complex, we divide our crowd task into multiple steps. We first ask the crowd to (i) filter out forum questions that are not entity-seeking questions. For the questions that remain, the crowd provides (ii) *user.∗* labels, and (iii) *entity.∗* labels. Taking inspiration from He et al. (2015), for each step, instead of directly asking for token labels, we ask a series of indirect questions as described in the next section that can help source high-precision annotations.

3.5.4 Crowd-sourcing Task

We defined three AMT tasks in the form of questionnaires:

- Questionnaire 1 : To identify posts of relevance for our task. This is to filter posts

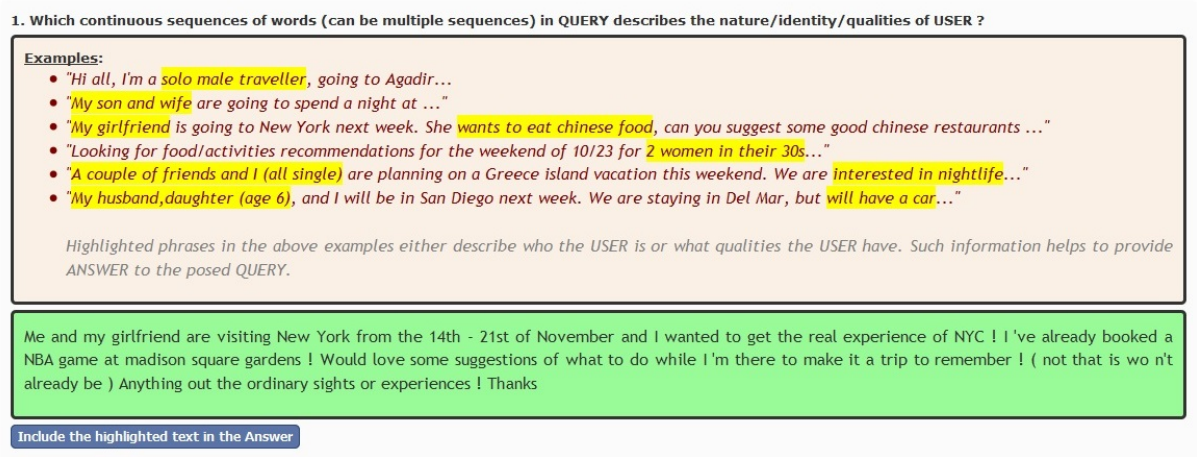


Figure 3.4: Snippet of the second questionnaire given to AMT workers

that may be unrelated to our task since forum posts can often contain reviews, advertisements, etc.

- Questionnaire 2 : To identify the *user* entities and its labels.
- Questionnaire 3 : To identify the answer entities and its labels.

In the first questionnaire (AMT Task 1) we ask the users to identify any non-entity seeking questions as well the number of entity types requested in a given query. We remove any posts that ask for multiple entity types.[9] The second questionnaire (AMT Task 2) asks the following question to the AMT workers.

- *"Which continuous sequences of words (can be multiple sequences) in the QUESTION describes the nature/identity/qualities of USER ?*

We paid $0.20 to each worker for this task. The QUESTION refers to the actual question posed by a user on a forum page and the answer to these questions gives us the *user.attr* labels. Figure 3.4 shows a sample snippet of the questionnaire.

The last questionnaire asks the following questions to the AMT workers.

- *"Given that the USER is asking only a single type of recommendation/suggestion, which sequence of words (only one sequence from a single sentence, prefer a continuous sequence) in QUESTION tells you what the USER is asking for ?"*
- *"What is the shortest sequence of words in "A1 (Answer to Question 1)" describes a category ? e.g. place to stay, restaurant, show, place to eat, place to have dinner, spot, hotel."*

[9]This is only so that additional work on resolving attributes and entities is not required Resolving entities and their corresponding attributes is a useful direction for future work.

	type	*attr*	*loc*
Avg. token level agreement	47.98	37.78	68.56

Table 3.3: Agreement for *entity* labels on AMT

- *"What words/phrases (need not be continuous, can be multiple) in the QUESTION give a sense of location about the ANSWER or "A2 (Answer to Question 2)"*
- *"What words/phrases (need not be continuous, can be multiple) in the QUESTION give more description about the ANSWER or the "A2 (Answer to Question 2)"*

These questions give us the *entity.type*, *entity.location* and *entity.attribute* labels. We paid $0.30 to each worker for this task.

We obtain two sets of labels (different workers) on each question. However, due to the complex nature of the task we find that workers are not complete in their labeling and we therefore only use token labels where both set of workers agreed on labels. Thus, we are able to source annotations with high precision, while recall can be low. Table 3.3 shows token-level agreement statistics for labels collected over a set of 400 tourism POI-recommendation questions. Some of the disagreement arises from labeling errors due to complex nature of the task. In other cases, the disagreement results from their choosing one of the several possible correct answers. E.g., in the phrase "*good restaurant for dinner*" one worker labels *entity.type* ='restaurant', *entity.attr* ='good' and *entity.attr* ='dinner', while another worker simply chooses the entire phrase as *entity.type*.

3.5.5 Training with partially labeled posts

We devise a novel method to use this partially labeled data, along with our small training set of expert labeled data, to learn the parameters of our CCM model. We adapt the Constraints driven learning (CODL) framework [Chang et al., 2007] which uses a semi-supervised iterative weight update algorithm, where the weights at each step are computed using a combination of the models learned on the labeled and the unlabeled set [Chang et al., 2007].

Given a dataset consisting of a few fully labeled as well as unlabeled examples, the CoDL learning algorithm first learns a model using only the labeled subset. This model is then used to find labels (in a hard manner) for the unlabeled examples while taking care of constraints (Section 3.5.2). A new model is then learned on this newly annotated set and is combined with the model learned on the labeled set in a linear manner. The parameter update can be described as:

$$(\omega^{(t+1)}, \rho^{(t+1)}) = \gamma(\omega^{(0)}, \rho^{(0)}) + (1-\gamma)\text{Learn}(\mathbb{U}^{(t)}) \tag{3.7}$$

Here, t denotes the iteration number, $\mathbb{U}^{(t)}$ denotes the unlabeled examples and Learn is a function that learns the parameters of the model. In our setting, Learn trains the neural network via back-propagation. Instead of using unlabeled examples in $\mathbb{U}^{(t)}$ we utilize the partially labeled set whose values have been filled in using parameters at iteration t and, inference over the set involves predicting only the missing labels. This is done using the ILP based formulation described previously, with an added constraint that the predicted labels for the partially annotated sequences have to be consistent with the human labels. γ controls the relative importance of the labeled and partial examples.

3.6 Evaluation

The goal of our experimental evaluation was to analyze the effectiveness of our model for the task of understanding MSRQs. We next describe our dataset, evaluation methodology and our results in detail.

3.6.1 Dataset

For our current evaluation, we used the following three semantic labels: *entity.type*, *entity.attr*, *entity.location*. We also used a default label *other* to mark any tokens not matching any of the semantic labels.

We use 150 expert-annotated tourism forum questions (9200 annotated tokens) as our labeled dataset and perform leave-one out cross-validation. This set was labeled by the author of the thesis and another colleague, by resolving differences in-person, to produce the combined labeled set. For experiments with partially labeled learning, we add 400 partially-annotated questions from crowd-sourced workers to our training set. As described in Section 3.5.4, each question is annotated by two workers and we retain token labels marked the same by two workers, while treating the other labels as unknown. We still compute a leave one out cross-validation on our original 150 expert-annotated questions (complete crowd data is included in each training fold).

3.6.2 Methodology

Sequence-tagged tokens identify *phrases* for each semantic label; therefore, instead of reporting metrics at the token level, we compute a more meaningful joint metric over tagged phrases. We define a matching-based metric that first matches each extracted

segment with the closest one in the gold set, and then computes segment-level precision using constituent tokens. Analogously, recall is computed by matching each segment in gold set with the best one in extracted set. As an example, for Figure 3.1, if the system extracts "convenient to the majority" and "local budget" for *entity.attr* (with gold *entity.attr* being "budget", "best" and "convenient to the majority that first time visitors would like to see"), then our matching-metric will compute precision as 0.75 (1.0 for "convenient to the majority"(covered completely by "convenient to the majority that first time visitors would like to see") and 0.5 for "local budget"(partially covered by "budget")) and recall as 0.45 (1.0 for "budget" (completely covered by predicted entity "local budget"), 0.0 for "best" (not covered by any predicted entities) and 0.333 for "convenient to the majority ... like to see"(covered by predicted "convenient to the majority")).

We use the Mallet toolkit[10] for our baseline CRF implementation and the GLPK ILP-based solver[11] for CCM inference. In the case of BiLSTM based CRF, we use the implementation provided by Gardner et al. (2017). The BiLSTM network at each time step feeds into a linear chain CRF layer. The input states in the LSTM are modeled using a 200-dimension word vector representation of the token. These word vector representations were with pre-trained using the Word2Vec model [Mikolov et al., 2013] on a large collection of $80,000$ tourism questions. In case of BERT BiLSTM CRF we use the contextualized BERT embeddings from the BERT-small pretrained model as an input to the LSTM layer and BERT implementation from HuggingFace Transformers [Wolf et al., 2019]. For CoDL learning we set γ to 0.9 as per the original authors' recommendations.

3.6.3 Results

Table 3.4 reports the performance of our semantic labeler under different incremental configurations. We find that the BiLSTM CRF and the BERT BiLSTM CRF based models (middle and lower halves of the table respectively) outperform a CRF system (upper half of the table) in each comparable setting - for instance, using a baseline vanilla CRF based system using all features gives us an aggregate F1 of 50.8 while the performance of a BiLSTM CRF and BERT BiLSTM CRF using features are 56.2 and 64.4 respectively. As a baseline we use the neural predicate tagger from the Deep SRL system [He et al., 2017] to utilize our label space and we find that it performs similar to our CRF setup. The use of hand-designed features, CCM constraints in the BERT BiLSTM CRF (referred to as BERT BiLSTM CCM) along with learning from partially annotated

[10]http://mallet.cs.umass.edu/

[11]https://www.gnu.org/software/glpk/

Model	F1 (entity.type)	F1 (entity.attr)	F1 (entity.loc)	F1 (aggr)
Deep SRL [He et al., 2017]	48.4	47.8	53.2	49.8
CRF (all features)	51.4	45.3	55.7	50.8
CCM	59.6	50.0	56.1	55.2
CCM (with all crowd data)	55.1	42.2	46.7	48.0
PS CCM	58.5	50.6	60.3	56.5
BiLSTM CRF	53.3	47.6	52.1	51.0
BiLSTM CRF+Feat	58.4	48.1	62.0	56.2
BiLSTM CCM+Feat	59.4	49.8	62.3	57.2
PS BiLSTM CCM+Feat	62.9	50.4	61.5	58.3
BERT Labeling	59.6	50.6	59.5	56.6
BERT BiLSTM CRF	63.4	56.5	**73.4**	64.4
BERT BiLSTM CRF+Feat	63.9	**57.9**	69.2	63.7
BERT BiLSTM CCM+Feat	66.5	56.7	72.9	65.3
PS BERT BiLSTM CCM+Feat	**70.8**	56.0	72.4	**66.4**

Table 3.4: Sequence tagger $F1$ scores using CRF with all features (feat), CCM with all features & constraints, and partially-supervised CCM over partially labeled crowd data. The second set of results mirror these settings using a bi-directional LSTM CRF. Results are statistically significant (paired t-test, p value<0.02 for aggregate $F1$ for each CRF and corresponding CCM model pair). Models with "PS" as a prefix use partial supervision.

Model	F1 (entity.type)	F1 (entity.attr)	F1 (entity.loc)	F1 (aggr)
CRF + lexical + token feat.	45.1	42.2	52.2	46.5
+ type feat. + desc. phrase + attr feat.	47.3	44.2	55.1	48.9
+ Word2Vec cluster feat. + word count feat.	51.4	45.3	55.7	50.8

Table 3.5: Feature ablation study using a vanilla CRF model.

crowd data has over a 15pt gain over the baseline BiLSTM CRF model. Further, we note that the usage of hand-curated features, within-sentence and cross-sentence constraints as well as partial supervision, each help successively improve the results in all configurations. Next, we study the effect of each of these enhancements in detail.

Effect of features: In an ablation study performed to learn the incremental importance of each feature (See Table 3.5), we find that descriptive phrases, and our hand-constructed multi-sentence type and attribute indicators improve the performance of each label by 2-3 points. Word2Vec features help type detection because *entity.type* labels often occur in similar contexts, leading to informative vectors for typical type words. Frequency of non stopword words in the multi-sentence post are an indicator of the word's relative importance, and the feature also helps improves overall performance.

Effect of constraints: A closer inspection of Table 3.4 reveals that the vanilla CRF configuration sees more benefit in using our CCM constraints as compared to the BiLSTM CRF based model (4pt vs 1pt). To understand why, we study the detailed precision-recall

Algorithm	Prec	Recall	F1
CRF (all features)	66.9	41.7	51.4
CCM (all features)	62.1	57.2	59.6
BiLSTM CRF with Features	54.1	63.6	58.4
BiLSTM CCM with Features	55.1	64.5	59.4
BERT BiLSTM CRF with Features	66.4	61.5	63.9
BERT BiLSTM CCM with Features	65.0	68.0	66.5

Table 3.6: (i) Precision and Recall of *entity.type* with and without CCM inference.

characteristics of individual labels; the results for *entity.type* are reported in Table 3.6. We find that the BiLSTM CRF based model has significantly higher recall than their equivalent vanilla CRF counter-part while the opposite trend is observed for precision. As a result, since two of the three constraints we used in CCM are oriented towards improving recall,[12] we find that they improve overall F1 more by finding tags that were otherwise of lower probability (i.e. improving recall). Interestingly, in case of the BERT BiLSTM CRF based model we find that precision-recall characteristics are similar (higher precision than recall) to those seen in the vanilla CRF based setup, and thus again, the benefit of using constraints is larger.

Effect of partial-supervision: In order to further understand the effect of partial-supervision, we trained a CCM based model that makes use of *all* the crowd-sourced labels for training, by adding conflicting labels for a question as two independent training data points. As can be seen, using the entire noisy crowd-labeled sequences (row labeled "CCM (with all crowd data)" in upper half of Table 3.4) hurts the performance significantly resulting in an aggregate $F1$ of just 48.0 while using partially labeled data with CCM results in an F1 of 56.5. The corresponding $F1$ scores of partially-supervised BiLSTM CCM and BERT BiLSTM CCM systems (trained using partially labeled data) are 58.3 and 66.4 respectively.

Overall: Our results demonstrate that the use of each of hand-engineering features, within-sentence and inter-sentence constraints and use of partially labeled data help improve the accuracy of labeling MSRQs.

3.7 Answering System

We now demonstrate the usefulness of our semantic labels and tagging framework by enabling a QA system which returns entity answers for MSRQs. Given a labeled question q_{lab}, an API based retrieval system $\mathbb{KB}$, we return an answer entity e using $\mathbb{KB}$, by

[12]Recall that we require at least one *entity.type* (hard constraint) and prefer at least one *entity.attr* (soft constraint)

constructing a text-based query $\bar{q}$ using q_{lab}. To the best of our knowledge we are the first to attempt such a QA task. We use the best performing tagging system (PS BERT BiLSTM CCM with features) to generate the semantic labels of the questions. These semantic labels and their targets are used to formulate a query $\bar{q}$ to the Google Places collection ($\mathbb{KB}$), which serves as our knowledge source.[13] The Google Places collection contains details about eateries, attractions, hotels and other points of interests from all over the world, along with reviews and ratings from users. It exposes an end point that can be used to execute free text queries and it returns entities as results.

We convert the semantic-labels tagged phrases into a Google Places query $\bar{q}$ via the transformation: "concat(*entity.attr*) concat(*entity.type*) in concat(*entity.location*)". Here, *concat* lists all tokens with type/attribute/location labels with space separators. Since some of the attributes may be negated in the original question, we filter out these attributes and do not include it as part of the query for Google Places.

Detection of Negations: We use a list of *triggers* that indicate negation. We start with a manually curated set of seed words, and expand it using synonym and antonym counter fitted word vectors [Mrksic et al., 2016]. The resulting set of *trigger* words flag the presence of a negation in a sentence. We also define the scope of a negation trigger as a token (or a set of continuous tokens with the same label) labeled by our sequence tagger that occur within a specified window of the trigger word. Table 3.7 reports the accuracy of our negation rules as evaluated by the author of the thesis. The 'Gold' columns denote the performance when using gold semantic label mentions. The 'System' columns are the performance when using labels generated by our sequence tagger.

Baseline: Since there are no baselines for this task, we adapt and re-implement a complex QA system (called WebQA) originally meant for finding appropriate Google results (documents) to questions posed in user forums [Vtyurina and Clarke, 2016]. WebQA first short-lists a set of top 10 words in the question using a tf-idf based scheme computed over the set of all questions. A supervised method is then used to further shortlist 3-4 words, to form the final query. In our setting we lack the data to train a supervised method for selecting these words from the tf-idf ranked list. Therefore, for best performance, instead of using supervised learning for further shortlisting keywords (as in the original paper), in our implementation, we choose the 3-4 best words manually from the top 10 words. This query executed against the Google Places collection API returns answer entities instead of documents.

We randomly select 300 new unseen questions (different from the questions used in the previous section), from a tourism forum website and manually remove 110 of those

[13] https://developers.google.com/places/web-service/

	Gold			System		
	P	**R**	**F1**	**P**	**R**	**F1**
Negations	86	66	74.6	85	62	71.7

Table 3.7: Performance of negation detection using gold sequence labels, and system generated labels

System	**Hits@3 (attempted) (%)**	**MRR**	**Hits@3 (all questions) (%)**
WebQA	31.6	0.28	19.5
WebQA (manual)	41.8	0.35	39.4
MSRQ-QA	**56.7**	**0.46**	**53.6**

Table 3.8: QA task results using the Google Places web API as knowledge source.

that were not entity-seeking. The remaining 190 questions form our test set. Note that, since we do not perform further filtering on these questions (unlike in the training data), this set may also contain questions seeking more than one type of entity. Our annotators manually check each entity-answer returned by the systems for correctness. Inter-annotator agreement for relevance of answers measured on 1300+ entities from 100 questions was 0.79. Evaluating whether an entity answer returned is correct is subjective and time consuming. For each entity answer returned, annotators need to manually query a web-search engine to evaluate whether an entity returned by the system adequately matches the requirements of the user posting the question. Given the subjective and time consuming nature of this task, we believe 0.79 is an adequate level of agreement on entity answers.

Results: Table 3.8 reports the Hits@3 score, which gives credit if any one of the top three answers returned is a correct answer – we only compute this score on questions where an answer is returned by the API. We also report the Hits@3 score over all questions i.e, the percentage of questions with correct hits (within the top three results) over the full set of questions. To distinguish between the two, we refer to them as '*Hits@3(attempted)*' and '*Hits@3 (all questions)*', respectively. Lastly, we also report Mean Reciprocal Rank (MRR) on the subset of attempted questions (any answer returned). In case the user question requires more than one entity type,[14] we mark an answer correct as long as one of them is attempted and answered correctly. Note that these answers are ranked by Google Places based on relevance to the query.

As can be seen, the use of our semantic labels (MSRQ-QA) results in nearly 15 point higher Hits@3 (attempted) score and a 14 point higher Hits@3 (all questions) score as compared to WebQA (manual), because of a more directed and effective query to Google

[14]A question can ask for multiple things, eg., 'museums' as well suggestions for "hotels".

No.	Question	Entity Type	System Answer
1	My family and my brother's family will be in Salzburg over Christmas 2015. We have arranged to do the Sleigh Ride on Christmas day but are keen to do a local style Christmas Day dinner somewhere. Any suggestions?	Special Dinner place	St. Peter Stiftskulinarium, Sankt-Peter-Bezirk 14, 5020 Salzburg
2	Heading to Salzburg by car on Friday September 18th with my wife and her parents (70's) and trying to make the most of the one day. Thinking about a SOM tour, but not sure what the best tour is, not a big fan of huge groups or buses, but would sacrifice for my Mother in Law (LOL). Also thinking about Old Town or the Salzburg Fortress. Any suggestions? Where to park to have easy access as well as a great place for dinner.Thanks so much!	Tour	Bob's Special Tours, Rudolfskai 38, 5020 Salzburg, Austria
3	What can you do in Helsinki on a Sunday morning? What would you recommend a tourist to do or see on a Sunday morning? I'll be ariving at 7 in the morning, and it seems like everything's closed on a Sunday morning- either its not open on Sundays or else it'll open but later on in the day.	Things to do / see	Senate Square, 00170 Helsinki, Finland Ateneum,Kaivokatu 2, 00100 Helsinki, Finland ..
4	I am planning to visit Agra for 2 days in mid Dec with my friends.My plan is to try some street food and do some local shopping on day 1 and thus wish to stay in a good budget 3 star hotel (as I wont be spending much time in the hotel) at walking distance from such street foodlocal shopping market.Then on the 2nd day, I want to just relax and enjoy the hotel.(I have booked a premium category hotel, Radisson Blu for this day hoping for a relaxed stay)Please suggest some good hotel or market around which I should book an hotel for my first day.	Hotel with location constraints	Hotel Taj Plaza, Agra, Taj Mahal East Gate, Near Hotel Oberoi Amar Vilas, VIP Road, Shilpgram, Agra, Uttar Pradesh 282001, India
5	Hi there. I am going to Tallinn in a month from just one night on a Saturday. I am 28 and am going with 5 of my friends. Were should we stay so we are near the best clubs in the city? Any recomendations are appreciated!!! Thanks.	Place to stay close to clubs	Club Prive, Tallinn, Estonia
6	A few friends and I are coming up to Newport for a couple of nights and are looking for restaurant suggestions. We are thinking something casual for the first night. Is Flo's any good? And then something nicer on Saturday night....preferably a restaurant with good seafood. Also, any suggestions for good breakfast?	Restaurant based on cuisine	The Red Parrot Restaurant, 48 Thames St, Newport, RI 02840, United States
7	Dear All forum members, I am Yash Khatri from Delhi.I am travelling to Srinagar on 13th July,2016 to 17th July,2016.I am going there for a show, and I'll be free on 15th and 16th July, 2016. I was thinking to hire a bike at Srinagar and travel toGulmargPahalgam.Queries :1) Where can I rent a bike at Srinagar and how much will it cost me?2) What is better for a quick visit; Gulmarg or Pahalgam?Please help!Thanks	Motorcycle rental	Kashmir Bikers - Bike Rentals, Sheikh complex , shiraz chowk ,khanyar, Near j&k bank khanyar, Srinagar, Jammu and Kashmir 190003
8.	In a couple of weeks, we will have an almost 2 hr layover in Zagreb before flying on to Dubrovnik. Any recommendations for lunch ?	A location for lunch that can be visited in a 2 hour layover	Hotel Dubrovnik,Gajeva ul. 1, 10000, Zagreb, Croatia
9.	Hi,I am looking for a good hotel in Shillong (preferably near Police bazar) which would offer free wifi, spa and are okay with unmarried couples. My budget is 3k maximum. please suggest the best place to stay.	Hotel with location and budget constraints	Hotel Pegasus Crown, Ward's Lake Road, Police Bazar, Shillong, Meghalaya 793001, India ;

Table 3.9: Some sample questions from our test set and the answers returned by our system. Answers in green are identified as correct while those in red are incorrect.

Error Type	**Error (%)**	**Examples**
Incorrect answer returned due to incorrect *entity.type*	23	Bad *entity.type* extractions results in incorrect answers.
Incorrect answer returned by knowledge source	23	*entity.attribute* criteria was not fulfilled - eg. Shop allows renting bicycles but not for tours.
Incorrect answer returned due to incomplete labeling	17	*entity.attribute* not getting extracted
Incorrect Answer/Answer not returned due to knowledge source limitations	37	Query requesting places "around" a city, or between two cities, *entity.type* extracted as "day trips", "cruises", etc. Requests for *entity.type* where queries were about bus services, activities and train stations.

Table 3.10: Classification of errors made by our MSRQ-labels based answering system (using Google Places web API as knowledge source)

Places collection. Overall, our semantic labels based QA system (MSRQ-QA) answers approximately 54% of the questions with a Hits@3 score of 57% for this challenging task of answering MSRQs.

Qualitative Study and Error Analysis of MSRQ-QA: Table 3.9 presents some examples of questions[15] answered by the QA system. As can be seen our system supports a variety of question intents/entities and due to our choice of an open semantic representation, we are not limited to specific entity types, entity instances, attributes or locations. For example, in $Q1$ the user is looking for "local dinner suggestions" on Christmas eve, and the answer entity returned by our system is to dine at the "St. Peter Stiftskulinarium" in Salzburg, while in $Q2$ the user is looking for recommendations for "SOM tours" (Sound of Music Tours). A quick internet search shows that our system's answer, 'Bob's Special Tours', is famous for their SOM tours in that area. This question also requests for restaurant suggestions in the old town, but since we focus on returning answers for just one *entity.type*, this part of the question is not attempted by our system. Questions with more than one *entity.type* requests are fairly common and this sometimes results in confusion for our system especially if *entity.attribute* tags relate to different *entity.type* values. Since we do not attempt to disambiguate or link different *entity.attribute* tags to their corresponding *entity.type* values, this is often a source of error. Our constraint that

[15]Actual questions posted on forums at TripAdvisor.com

forces all *entity.type* labels to come from one sentences mitigates this to some extent, but this is can still be a source of errors. $Q4$ is incorrect because the entity returned does not fulfil the location constraints of being close to the "bazar" while $Q5$ returns an incorrect entity type.

$Q9$ is a complicated question with strict location, budget and attribute constraints and the top ranked returned entity "Hotel Pegasus Crown" fulfills the most requirements of the user.[16]

Error Analysis: We conducted a detailed error study on 105 of the test set questions and we find that approximately 60% of questions were not answered by our QA system pipeline due to limitations of the knowledge source while approximately, 40% of the 'recall'[17] loss in the system can be traced to errors in the semantic labels. See Table 3.10 for a detailed error analysis.

3.8 Understanding MSRQs in another domain

In contrast to methods that require tens of thousands of training data points, our question understanding framework works with a few hundred questions. We demonstrate the general applicability of our features and constraints by employing them on the task of understanding multi-sentence questions seeking *book* recommendations.

Using questions collected from an online book reading forum[18] we annotated[19] 95 questions with their semantic labels. We retrained both CRF and CCM based supervised systems as before on this dataset. Because location is not relevant for books, we use the two general labels: *entity.type* and *entity.attr*.

We train the labeler with no feature adaptation or changes from the one developed for tourism, retaining the same constraints as before. We tune the hyper-parameters with a grid-search. Table 3.11 shows the performance of our sequence labeler over leave-one out cross-validation. We find that that our generic features for *type* and *attr* defined earlier work acceptably well for this domain as well and we obtain $F1$ scores comparable to those seen for tourism. These experiments demonstrate that simple semantic labels can indeed be useful to represent multi-sentence questions and that such a representation is easily applicable to different domains.

[16]The hotel does not offer a spa and even with manual search we could not find a better answer

[17]Hits@3 (all questions)

[18]https://forums.onlinebookclub.org/

[19]Inter-annotator agreement measured on 30% of the data was 0.75

Algorithm	F1 (type)	F1 (attr)	F1 (aggr)
CRF	41.5	42.1	41.8
CCM	52.1	43.8	47.9
BiLSTM CRF	52.6	39.9	46.3
BiLSTM CRF+Feat	54.6	45.1	49.9
BiLSTM CCM+Feat	55.9	44.6	50.3
BERT BiLSTM CRF	68.4	53.7	61.1
BERT BiLSTM CRF+Feat	**70.8**	52.0	61.4
BERT BiLSTM CCM+Feat	69.4	**55.8**	**62.6**

Table 3.11: Labeling performance for Book recommendation questions (paired t-test, p value<0.01 for aggregate $F1$ in vanilla CRF and CCM model pairs & BiLSTM CRF and CCM model pairs).

3.9 Summary

In this chapter, we presented the novel task of understanding multi-sentence entity-seeking recommendation questions (MSRQs). MSRQs expose novel challenges for semantic parsing as they contain multiple sentences requiring cross-sentence interactions and also need to be built in low data settings due to challenges associated with sourcing training data. We defined a set of open semantic labels that we used to formulate a multi-sentence question parsing task.

Our solution consists of sequence labeling based on a BiLSTM CRF model. We used hand-engineered features, inter-sentence CCM constraints, and partially-supervised training, enabling the use of crowdsourced incomplete annotation. We find these methods results in a 7pt gain over baseline BiLSTM CRFs. The use of contextualized pretrained embeddings such as BERT result in an additional 6-8pt improvement. We further demonstrated the strength of our work by applying the semantic labels towards the novel task of answering tourism POI-recommendation questions using a web-API based unstructured knowledge source. Further, we demonstrated how our approach allows rapid bootstrapping of MSRQ semantic parsers for new domains.

The work presented in this chapter is the first attempt towards QA in the challenging setting of answering tourism POI-recommendation questions directly on the basis of information in unstructured and semi-structured knowledge sources. Our best model answered 54% of the questions with a Hits@3 score of 57 pts. Resources from the work done in this chapter are available at https://ibm.biz/MSRQ-QA.

Instead of relying on a method that needs intermediate annotations for answering, one could also train a neural system that uses only the correct answer entities as a source of supervision. This would require generating a large dataset of labeled QA pairs which could perhaps be sourced semi-automatically using data available in tourism QA forums.

However, answer posts in forums can often refer to multiple entities and automatically inferring the exact answer entity for the question can be challenging. Further, we will need to devise efficient techniques to deal with hundreds of thousands of potential class labels (entities). We study this direction of work in our next chapter and present methods that help overcome some of these challenges.

Chapter 4

QA without Intermediate Annotations

In the previous chapter, we developed a system for answering MSRQs that used the Google Places API, as its knowledge source. While the system's approach of parsing and API-based retrieval had the advantage of working with less training data, it did not allow us to use expressive queries. For instance, a complex constraint expressed by a phrase such as "*suitable for a large gathering of teenagers and serves cheap, non-spicy, Indian food*", could not be modeled explicitly as we relied on a text-based query, generated using our semantic labels. As a result, we had no control over the reasoning process used for answering, and such constraints may not have been adequately considered by the black-box retrieval system.

Therefore, in this chapter, we develop an approach for answering MSRQs, using a collection of reviews describing entities,[1] as our knowledge source. We find that this task poses novel real-world challenges of reasoning at scale as discussed below.

Challenges of Reasoning: Answering MSRQs (POI-recommendation questions) requires models to reason over entity reviews that could contain sarcasm, contradictory opinions etc, as well as, mentions of other entities (e.g., for comparison). Thus, the nature of reasoning[2] for answering such questions is different from typical machine-reading comprehension [Rajpurkar et al., 2018, Khashabi et al., 2018, Yang et al., 2018], entailment-based reasoning or common-sense reasoning tasks [Clark et al., 2019, Chen et al., 2019,

[1] We use the word 'entity' and 'POI' interchangeably

[2] Note that our use of the word 'reasoning' in this thesis is in similar to its use in recent QA literature, where 'reasoning' is used to refer to the 'inference' of answers using textual and/or symbolic knowledge sources [Abujabal et al., 2019, Yang et al., 2018]. This is in contrast to traditional notions of 'reasoning' in Computer Science and Mathematics which often involve theorem proving and axiomatic reasoning over symbolic representations.

Huang et al., 2019, Abujabal et al., 2019]. In addition, as discussed in the previous chapter, questions may also include requirements based on physical location, budget, timings as well as, other subjective considerations about ambience, quality of service, etc.

Challenges of Scalability: Questions have a large candidate answer space in our task since there may be thousands of POIs in a city, each represented by hundreds of reviews (e.g., New York has tens of thousands of restaurants to choose from). To address challenges of reasoning at scale in QA tasks, existing models, often employ retriever-ranker architectures which first reduce the search space by filtering documents using methods such as BM25 [Robertson and Zaragoza, 2009] ranking, and then apply deeper reasoning models on the reduced set. Additionally, methods may also use document structure to extract salient portions of the document or truncate documents to the first $800 - 1000$ tokens [Kundu and Ng, 2018, Fisch et al., 2019] to improve scalability. However, as our experiments in Section 4.6 show, neither strategies are effective in our task. Pruning the search-space based on TF-IDF scores does not work well, because documents in our task consist of opinions that are expressed in reviews; thus, they share a large common vocabulary, resulting in similar TF-IDF scores. In contrast, documents used in machine-reading comprehension [Khashabi et al., 2018, Yang et al., 2018], entailment-based reasoning or common-sense reasoning tasks [Chen et al., 2019, Huang et al., 2019, Abujabal et al., 2019], etc have distinguishing terms (for example, due to topical entities), which help generate better TF-IDF scores. To further illustrate with an example, in our task, the average TF-IDF based inter-document cosine similarity for review documents of restaurants in New York is 0.35, while, for training data paragraphs in SQuAD [Rajpurkar et al., 2018], it is just 0.05.

In addition, arbitrarily truncating review documents can cause a loss of crucial information. Thus, typical QA algorithms which apply cross-attention between question and candidate answer texts, do not scale in our task where entities may have long (bag-of-reviews) documents (see Table 4.4 for comparison of document sizes across different QA datasets).

In response to the novel challenges of reasoning at scale posed by our task, we present a scalable three-stage *cluster-select-rerank* model. It extends traditional retriever-ranker architectures by incorporating a clustering stage which first *clusters* text for each entity to identify exemplar sentences describing an entity. Then, similar to recent work on open domain QA [Karpukhin et al., 2020], instead of employing retriever architectures based on sparse vector representations, it uses a neural information retrieval (IR) module with dense representations of questions and entities to *select* a set of potential entities from the large candidate set. Finally, it uses a *reranker* with a deeper attention-based architecture to pick the best answers from the selected entities.

4.1 Contributions

(1) We introduce the novel task of answering multi-sentence entity-seeking (POI-seeking) recommendation questions using a collection of reviews describing entities. (2) We harvest a novel dataset[3] of tourism questions consisting of 47,124 QA pairs extracted from online travel forums. Each QA pair consists of a question and an answer entity ID, which corresponds to one of the over 200,000 POI-entity review documents collected from the Web. (3) We present detailed experiments with task specific baselines such as those using BM25, POI-ratings and the cluster-select-rerank (CsrQA) approach. We include detailed ablation studies highlighting the importance of each stage in the CsrQA pipeline. (4) We find that the CsrQA approach does better than a pure IR or a pure attention-based reasoning approach yielding nearly 25% relative improvement in Hits@3 scores over both approaches.[4]

4.2 Data Collection

Most recent QA datasets have been constructed using crowdsourced workers who either create QA pairs given documents [Rajpurkar et al., 2018, Reddy et al., 2018] or identify answers for real world questions [Nguyen et al., 2016, Kwiatkowski et al., 2019]. Creating QA datasets manually using the crowd can be very expensive. We therefore chose to automatically harvest a dataset using tourism forums and a collection of reviews. We first crawled forum posts along with their corresponding conversation thread, as well as, meta-data including the date and time of posting. We then also crawled reviews for restaurants and attractions, for each city from TripAdvisor. Hotel reviews were scraped from Booking.com. Entity meta-data such as the address, ratings, amenities was also collected where available.

Filtering Questions: We observe that, apart from questions, forum users also post summaries of trips, feedback about services taken during a vacation, along with open-ended non entity-seeking questions such as, queries about the weather or economic climate of a location. We remove such questions using precision oriented rules which discard questions that do not contain any one of the phrases in the set {"*recommend*", "*suggest*", "*where*", "*place to*" "*best*" and "*option*"}. While the use of such rules may introduce a bias towards a certain class of questions, they continue to retain a lot of variability in

[3] Available at https://github.com/dair-iitd/TourismQA

[4] The work in this chapter was done jointly with Aditi Partap and Krunal Shah. Aditi contributed to the scripts used for data extraction while Krunal contributed to the development and implementation of the models. The part of the work done by both appeared in their respective B.Tech theses.

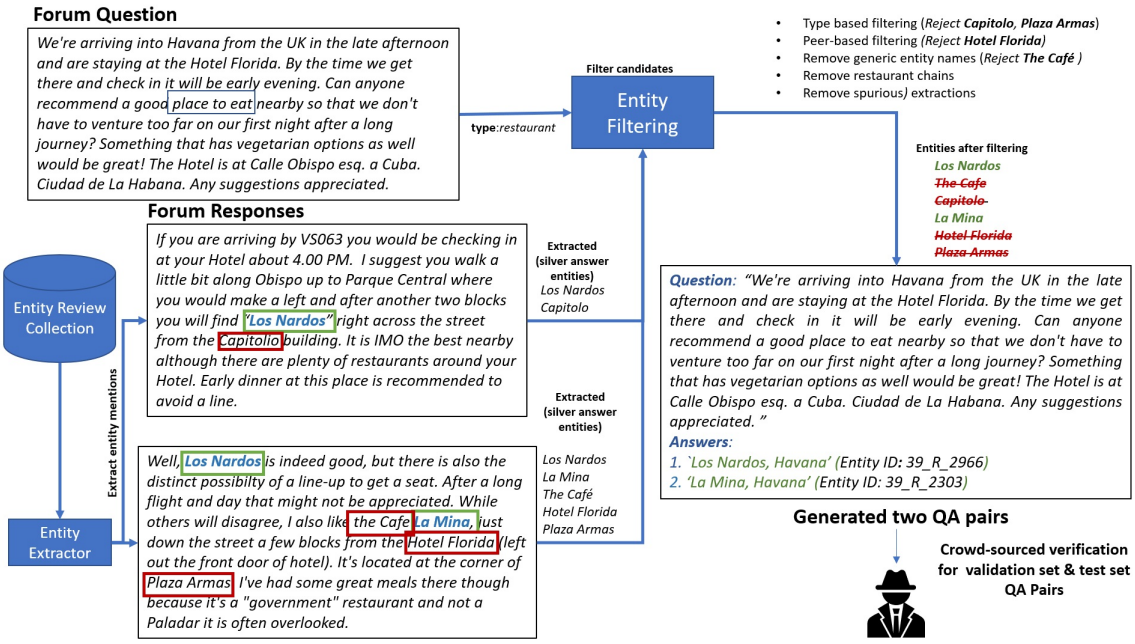


Figure 4.1: Entity Answers are extracted from forum post responses to generate QA Pairs. Entities marked in red indicate false positive extractions. Each entity in our collection has an ID of the form <city_id >_<POI type>_<number>. The dataset has three classes of POIs - restaurants (R), attractions (A) and hotels (H). Example forum question from https://bit.ly/2zIxQpj adapted for illustration.

language of expression (as Table 4.3 suggests), that still makes the task challenging. We further remove posts explicitly identified as "Trip Reports" or "Inappropriate" by the forum. Excessively long questions ($\geq$ 1.7X more than average) were also removed.

4.2.1 Answer Extraction

We create a list of entity names crawled for each city and use it to find entity mentions in user responses to forum posts. A high level entity class (hotel, restaurant, attraction) for each entity is also tagged based on the source of the crawl. Each user response to a question is tagged for part-of-speech, and the nouns identified are fuzzily searched[5] in the entity list (to accommodate for typographical errors). This gives us a noisy set of "*silver*" answer entities extracted from free text user responses for each question. We now describe a series of steps aimed at improving the precision of extracted silver answers, resulting in our gold QA pairs (summary in Figure 4.1).

[5]Levenstein distance < 0.05

4.2.2 Filtering of Silver Answer Entities

Type-based filtering: As a first step, we use the multi-sentence question understanding component developed in the previous chapter [Contractor et al., 2020] to identify phrases in the question that could indicate a target entity's "*type*"and "*attribute*". For instance, in the example in Figure 4.1 tokens "*place to eat*" will be identified as an *entity.type* and the phrase "*has vegetarian options as well*" will be identified as *entity.attribute*.

All entities collected from the online forums come with labels (from a set of nearly 210 unique labels) indicating the nature of the entity. For instance, restaurants have cuisine types mentioned, attractions are tagged as museums, parks etc. Hotels from the hotel booking website are simply identified as "hotels". We manually cluster the set of unique labels into 9 clusters.[6]

For a given question we use the phrase tagged with the *entity.type* tag, and determine its closest matching cluster using Word2Vec [Mikolov et al., 2013] embedding representations. Similarly, for each silver answer entity, we identify the most likely cluster by matching its meta-data attributes – we then use the cluster with the most number of attribute matches, as the cluster assigned for the silver answer entity. If the expected target entity cluster and the silver-answer entity cluster do not match, the silver-answer entity is removed from the pool. For example, in Figure 4.1, QA pairs that use entities *Capitolo* and *Plaza Armas* as answers, get discarded due to incorrect types.

Peer-based filtering: As mentioned previously, all entities have their type information (hotel, attraction or restaurant) indicated as part of meta-data. Using all *silver* (entity) answers for a question, we determine the frequency counts of each type encountered and remove any silver (entity) answer that does not belong to the majority type. For example, the QA pair with entity *Hotel Florida* with type "hotel" is discarded because the majority type, based on its remaining peers, is "restaurant". If there is no clear majority type, the question is discarded (i.e, all QA pairs are discarded).

Filtering entities with generic names: Some entities are often named after cities, or generic place types – for example "The Cafe" or "The Spa" which can result in spurious matches during answer extraction. We collect a list of entity types[7] from Google Places.[8] and remove any answer entity whose name matches any entry in this list.

Removing entities that are chains and franchises: Answers to questions can also be names of restaurants or hotel chains without adequate information to identify the actual franchisee referred. In such cases, our answer extraction returns all entities in the city with that name. We thus, discard all such QA pairs.

[6] https://github.com/dair-iitd/TourismQA/blob/master/data/common/cluster_categories.json

[7] Examples of types include "cafe", "hospital", "bar", etc.

[8] *https://developers.google.com/places/web-service/supported_types*

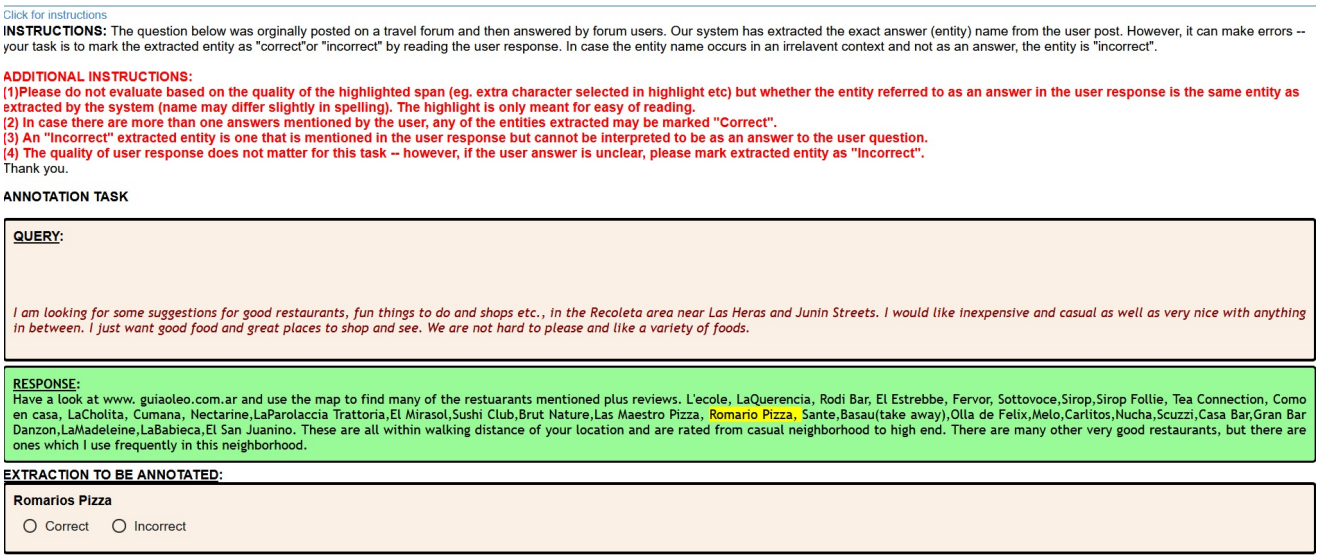


Figure 4.2: Human Intelligence Task (HIT) set up on Amazon Mechanical Turk to clean test and validation sets.

Removing spurious candidates: User answers in forum posts often have multiple entities mentioned not necessarily in the context of an answer but for locative references (e.g. "opposite Starbucks", or "near Wendys") or for expressing opinions on entities that are not the answer. We write simple rules to remove candidates extracted in such conditions (e.g.: if more than one entity is extracted from a sentence, we drop them all or if entity mentions are in close proximity to phrases such as "next to", "opposite". they are dropped).[9]

Additionally, we review the set of entities extracted and remove QA pairs with entity names that were common English words or phrases (eg: "August", "Upstairs", "Neighborhood" were names of restaurants that could lead to spurious matches).[10] We removed 322 unique entity names as a result of this exercise. Note that it is the only step that involved human annotation in the data collection pipeline thus far.

4.2.3 Qualitative Study: Data

We studied 450 QA pairs of the train-set,[11] representing approximately 1% of the dataset, for errors in the automated data collection process. The errors can be traced to one of four major causes (i) (16%) Entity name was a generic English word (e.g. "The Park") (ii) (27%) Entity matched another entity in the answer response which was not intended to be the answer entity to the original question. (e.g. Starbucks in "next to Starbucks")

[9]Full list: https://github.com/dair-iitd/TourismQA/blob/master/data/common/neighborhood_words.json
[10]https://github.com/dair-iitd/TourismQA/blob/master/data/common/common_names.json
[11]Note: this set is not cleaned by crowd-sourced workers

	#Ques.	QA pairs	Tokens per ques.	#QA Pairs with Hotels	#QA Pairs with Restr.	#QA Pairs with Attr.
Training	18,531	38,586	73.30	4,819	30,106	3,661
Validation	2119	4,196	70.67	585	3267	335
Test	2,173	4,342	70.97	558	3,418	366

Table 4.1: QA Pairs in train,validation and test sets

(iii) (31%) Entity matched another entity with a similar name but of a different target class (e.g. hotel with same name instead of restaurant). (iv) (13%) Failing to detect negations/negative sentiment (e.g. an entity mention in a post where the user says "*i wouldn't go there for the food*". (v) The remaining 13% of the errors were due to errors such as invalid questions (non-entity seeking), or incorrect answers provided by the forum users.

Crowd-sourced Data Cleaning: As our error study in the previous section shows, our automated QA pair extraction methods are likely to have some degree of noise. In order to facilitate accurate bench-marking, we crowd-source and clean our validation and test sets. We use the Amazon Mechanical Turk (AMT[12]) for crowd-sourcing. Workers are presented with a QA-pair, which includes the original question, an answer-entity extracted by our rules and the original forum-post response thread where the answer entity was mentioned. Workers are then asked to check if the extracted answer entity was mentioned in the forum responses as an answer to the user question. We spend \$0.05 for each QA pair costing a total of \$550. The crowd-sourced cleaning was of high quality; on a set of 280 expert annotated question-answer pairs, the crowd had an agreement score of 97%. As a result of the crowd-sourced cleaning, out of a total of $10,895$ QA pairs across the validation and test sets, 21.64% of the QA pairs were discarded indicating that our high precision rules for generating QA pairs have an answer extraction accuracy of 78.36%. The resulting dataset is summarized in Table 4.1.

We note that since workers are only asked to assess the extracted answers, our QA dataset is likely to contain false negatives, i.e. candidates that may be valid answers for a question but are not extracted by our automated methods (or are not mentioned by forum users in posts) as answers. However, due to the large candidate space it is infeasible to manually annotate each candidate with respect to a question. Thus, our task also shares challenges seen in evaluating recommendation systems [Valcarce et al., 2020], where the relevance judgements are sparse and incomplete (unlike traditional IR tasks). We discuss the impact of partial relevance judgements in more detail in Section 4.6. However, we note that the extraction accuracy (78.36%) is comparable to that seen in some existing datasets such as TriviaQA [Joshi et al., 2017] and is useful for training.

[12]http://requester.mturk.com

Avg # Tokens	3266
Avg # Reviews	69
Avg # Tokens per Review	47
Avg # Sentences	263

Table 4.2: Knowledge source consisting of 216, 033 entities and their reviews

Feature	%	Examples of Phrases in Questions
Budget constraints	23	*good prices,* *money is a bit of an issue* *maximum of $250 ish in total*
Temporal elements	21	*play ends around at 22:00 (it's so late!)* *.. dinner before the show,* *theatre for a Saturday night* *open christmas eve*
Location constraint	41	*dinner near Queens Theatre,* *staying in times square;would like it close,* *options in close proximity (walking distant)* *easy to get to from the airport*
Example entities mentioned	8	*found this one - Duke of Argyll* *done the Wharf and Chinatown,* *no problem with Super 8*
Personal preferences	61	*something unique and classy,* *am not much of a shopper,* *love upscale restaurants,* *avoid the hotel restaurants,* *Not worried about eating healthy* *out with a girlfriend for a great getaway*

Table 4.3: Classification of Questions. (%) does not sum to 100, because questions may exhibit more than one feature.

4.2.4 Data Characteristics

In our dataset, the average number of tokens in each question is approximately 73, which is comparable to the document lengths for some existing QA tasks. Additionally, our entity documents are larger than the documents used in existing QA datasets – they contain 3, 266 tokens on average. Lastly, answering any question requires studying all the possible entities in a given city – the average number of candidate answer entities per question is more than 5, 300, which further highlights the challenges of scale.

Our dataset contains QA pairs for 50 cities. The total number of entities in our dataset is 216, 033. Details about the knowledge source are summarized in Table 4.2 and additional statistics included in Appendix A. In almost every city, the most common entity class is restaurants. On average, each question has 2 gold answers extracted. Questions can include requirements based on physical location, budget, timings as well as, other *subjective* considerations related to ambience, quality of service, etc. A qualitative study of 100 random questions suggests that 61% of the questions contain personal preferences of users, 23% of the questions contain budgetary constraints, while 41% contain locative constraints (Table 4.3).

4.3 Problem Statement

Given a POI-recommendation question q (as in Figure 4.1) its target class t (for example, restaurant), the city c (for example, Havana), a candidate space $\mathbb{E}_c^t$ of target-class entities (POIs) for each corresponding city, and a collection of reviews $\mathbb{R}_e$ describing each entity e, the goal of our task is to score each candidate with respect to a question, for relevance.

4.4 Related Work: QA & IR

We compare and contrast our work against related work in QA and IR.

QA Tasks: Recent question answering tasks such as those based on reading comprehension require answers to be generated either based on a single passage, or after reasoning over multiple passages (or small-sized documents) (e.g. *SQuAD* [Rajpurkar et al., 2018], *HotpotQA* [Yang et al., 2018], *NewsQA* [Trischler et al., 2017] (see Table 4.4). Answers to questions are assumed to be stated explicitly in the documents [Rajpurkar et al., 2018] and can be derived with single or multi-hop reasoning over sentences mentioning facts [Yang et al., 2018]. Other variants of these tasks add an additional layer of complexity where the document containing the answer may not be known and needs to be retrieved from a large corpus before answers can be extracted/generated (e.g. *SearchQA* [Dunn et al., 2017], *MS MARCO* [Nguyen et al., 2016], *TriviaQA* [Joshi et al., 2017]). Models for these tasks typically use retriever-ranker architectures based on sparse vector representations like TF-IDF and BM25 ranking [Robertson and Zaragoza, 2009] to retrieve and sub-select candidate documents [Chen et al., 2017b]; deeper reasoning is then performed over this reduced space to return answers for scalability. However, we find that in our task, retrieval strategies such as BM25 perform poorly[13] and are thus not effective in reducing the candidate space (see Section 4.6). As a result, our task requires processing 500 times more documents per question (Table 4.4) and also requires reasoning over large entity review-documents that consist of noisy opinions. Further, traditional QA models such as BiDAF [Seo et al., 2016] or those based on BERT [Devlin et al., 2019] are infeasible[14] to train for our task. Thus, while existing tasks and datasets have been useful in furthering research in comprehension, inference and reasoning, we find that they do not always reflect the complexities of real-world question answering motivated in our task. We note that our work is also related to QA tasks defined for "Community Question-Answering (CQA)" [Harel et al., 2019]. However, in contrast to CQA tasks aimed at fetching ex-

[13] Hits@3 of 7%

[14] BiDAF requires 43 hours for 1 epoch (4 K-80 GPUs)

Dataset	Knowledge Source	Answer type	Avg. tokens in docs.	Answer doc. known	Multiple docs* required for answering	Reasoning over opinions
SQuAD [Rajpurkar et al., 2018]	Wikipedia paragraphs	Span	137	Y	N	N
NewsQA[Trischler et al., 2017]	CNN News articles	Span	≈ 300	Y	N	N
RACE [Lai et al., 2017]	Passages on topics	Choices	350	Y	N	N
SearchQA [Dunn et al., 2017]	Web snippets	Span	25-46	N	N	N
MSMARCO [Nguyen et al., 2016]	Web article snippets	Free text	10	N	Y	N
WikiReading [Hewlett et al., 2016]	Wikipedia articles	Infobox field	489	N	N	N
TriviaQA [Joshi et al., 2017]	Web articles	Span	800†	N	Y	N
HotPot-QA [Yang et al., 2018]	Wikipedia paragraphs	Span	≈ 800	N	Y	N
ELI5 [Fan et al., 2019]	Passages on topics	Free text	858	Y	N	N
TechQA [Castelli et al., 2019]	IT support notes	Free text	48	Y	Y	N
Dialog QA - **QuAC** [Choi et al., 2018]	Wikipedia passages	Span	401	Y	N	N
Dialog QA - **CoQA** [Reddy et al., 2018]	Passages on topics	Free text	271	Y	N	N
Our Dataset	Reviews	Entity (doc.)	3266	N	Y	Y

Table 4.4: Related datasets on Machine reading/QA and their characteristics. Unlike other existing datasets, our task requires us to reason over *opinions*. For reading comprehension tasks, the document containing the actual answer may not always be known. *"docs" refers to what the task would consider as its document (e.g., fact sentences for OpenBookQA). †Most questions in TriviaQA are answerable using only the first few hundred tokens in the document.

isting answers from forum threads or finding similar questions on a forum [Harel et al., 2019, Hoogeveen et al., 2018b], in our task, tourism POI-recommendation questions are answered by returning entity answers using a collection of reviews describing entities.

IR Tasks: Our QA task is one that also shares characteristics of information retrieval (IR), because, similar to document retrieval, answers in our task are associated with long entity documents, though they are without any additional structure. The goal of IR, specifically document retrieval tasks, is to retrieve documents for a given query. Neural models for IR use scalable architectures to create dense representations for queries and documents, and maximize mutual relevance in latent space [Mitra and Craswell, 2018]. To improve dealing with rare words, recent neural models also incorporate lexical matching along with semantic matching [Mitra et al., 2017]. However, unlike typical retrieval tasks, the challenge for answering in our task is not merely that of semantic gap, but that opinions need to be *reasoned* over and aggregated in order to assess relevance of the entity document. For instance, models for our task may need to understand that a restaurant with "*loud music*" or one "*without elevator access*" may not be most suitable to dine with an "*elderly grandparent*". This challenge is similar to other reading comprehension QA tasks that require deeper reasoning over text.

In summary, our task requires the scalability of IR models along with deeper reasoning required in QA tasks. Thus, in this chapter we use a coarse-to-fine architecture that sub-selects documents using dense representations from neural IR and then uses a deep reasoner over the selected subset to answer questions. In our work we use an existing neural IR model called Duet [Mitra et al., 2017, Mitra and Craswell, 2019] which is described below.

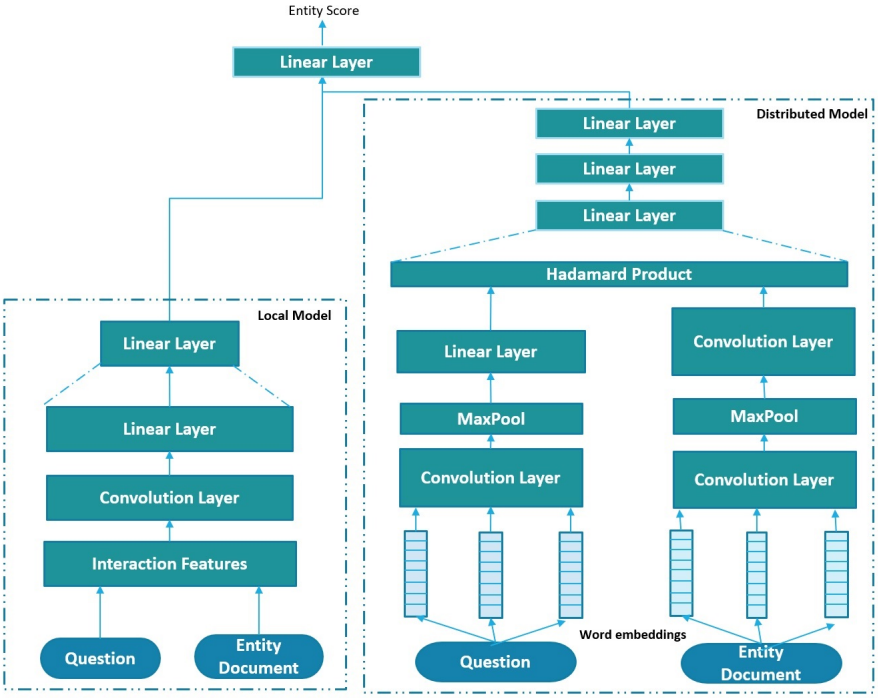


Figure 4.3: The Duet retrieval model [Mitra et al., 2017, Mitra and Craswell, 2019]

4.4.1 Duet – a Neural IR Network

Duet [Mitra et al., 2017, Mitra and Craswell, 2019] is an interaction-based neural network that compares elements of the question with different parts of a document and then aggregates evidence for relevance. It uses both local as well as distributed representations to capture lexical and semantic features. It is quite scalable for our task, since its neural design is primarily based on CNNs (Figure 4.3).

Local Model: The local representations are created using a term-document matrix (interaction features), which contains inverse-document-frequency (IDF) scores of words, for each term-position in the document. The interaction matrix $\mathbf{G}$, based on the i^{th} word (w_i^q) of the question q, and the j^{th} word w_j^e from the reviews of entity e, is given by:

$$\mathbf{G}_{i,j} = \begin{cases} IDF(\mathrm{w}_i^q), & \text{if } \mathrm{w}_i = \mathrm{w}_j^e \\ 0, & \text{otherwise} \end{cases} \tag{4.1}$$

This interaction matrix of size |q|x|e| is passed through a convolution network with f filters and a kernel size of |e|x1 and stride of 1. The output of this convolution operation is given by:

$$Z_i = ReLU(\mathbf{G}_i^T \mathbf{W}) \tag{4.2}$$

where the *ReLU* activation layer is applied element-wise and, **W** is a parameter matrix of size |e|xf. The output **Z** is a matrix of dimensions fx|q|.[15] This output is fed through a series of fully connected layers to return a score.

Distributed Model: The distributed model uses the Glove vector embeddings [Pennington et al., 2014] to create vector representations for words in both, questions and entity documents. These are independently passed through convolution layers with window-based max-pooling. The question representations are fed to a fully-connected linear layer while the entity document representations are further encoded using another convolution layer. The representations from the question and the entity document are combined together and then jointly processed using a series of fully connected linear layers to return a score. The hyper-parameter settings used are described in Section 4.6.1.2.

4.5 The Cluster-Select-Rerank Model

A model built for our task needs to address its novel challenges of reasoning at scale. As mentioned previously, each entity in our task is represented by a long, bag-of-reviews document; existing approaches, such as arbitrarily truncating documents to reduce length [Joshi et al., 2017], are not appropriate due to the lack of structure. Further, there are thousands of candidate entities for each question and reducing the candidate search space using TF-IDF style methods [Chen et al., 2017b] do not work well due to the nature of review documents (reviews express opinions about similar aspects/topics as opposed to the distinct, informative topics seen in typical QA/IR tasks). Finally, answering a question requires deep reasoning over the document for each candidate and the challenges of scale make the application of existing models intractable.

Our proposed baseline for this task consists of three major components designed to address these challenges: (1) a **clustering** module to generate representative entity documents that are smaller in size (in terms of document length), (2) a fast scalable neural retrieval model that uses dense representations of questions and entities to **select** candidate entities to reduce the search space, and (3) a QA-style **re-ranker** that reasons over the selected answers and scores them to return the final top-ranked answers. We refer to it as CsrQA and now describe each component in detail.

[15] We use $f = 128$

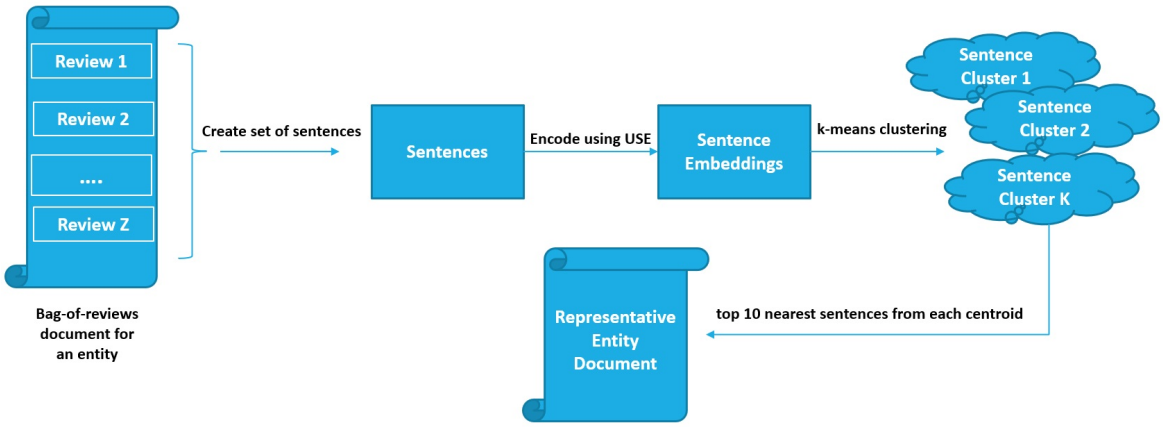


Figure 4.4: Representative documents created from Bag-of-Reviews entity documents, using clustering.

4.5.1 *Cluster: Representative Entity Document Creation*

We create entity documents $\bar{d}_e$ for each entity e by concatenating all reviews for an entity i.e, $\bar{d}_e = \text{concat}(m \in \mathbb{R}_e)$. As stated previously, these entity documents are much larger than documents used by previous QA tasks. In order to make training a sufficiently expressive neural model tractable, CsrQA first constructs smaller representative documents.[16] It encodes each review sentence in $\bar{d}_e$ using the pre-trained universal sentence encoder (USE) [Cer et al., 2018] to generate sentence embeddings. It then clusters sentences within each document $\bar{d}_e$ and uses the top-k sentences (nearest to the cluster centroid) from each cluster, to represent the entity e (process summarized in Figure 4.4). In our experiments we use $k = 10$ and generate 10 clusters per entity, thus reducing our document size to 100 sentences each. This constitutes an approximately 70% reduction in document size. We note that despite this reduction our problem continues to be large-scale. This is because our documents are still larger than those used in most QA tasks and before returning an answer to a question, the system still has to explore over 500 times[17] more documents, as compared to most existing QA tasks.

4.5.2 *Select:* Shortlisting Candidate Answers

In this step, CsrQA trains a neural retrieval model with the question as the query and representative entity documents as the text corpus. As its retrieval model, it uses the recently improved Duet network [Mitra and Craswell, 2019].

[16] representative documents are a set of review sentences

[17] Most QA tasks with large answer spaces are able to filter (reduce to top-10) candidates using TFIDF-style methods.

As described in Section 4.4.1, Duet is an interaction-based neural network that uses both local, as well as distributed representations, to capture lexical and semantic features.

Training: Duet is trained over the QA-pair training dataset and 10 randomly sampled negative examples and uses cross-entropy loss. Duet can be seen as ranking the full candidate answer space for a given question, since it scores each representative entity document. CsrQA selects the top-30 candidate entities from this ranked list for a deeper reading and reasoning, as described in the next section.

4.5.3 *Rerank:* Answering over Selected Candidates

In this step, our goal is to perform careful reading and reasoning over the shortlisted candidate answers to build the best QA system. The CsrQA implements a model for re-ranking based on Siamese network [Rao et al., 2016, Lai et al., 2018] with recurrent encoding and attention-based matching. The model is summarized in Figure 4.5.

Input Layer: It uses 128-dimensional Word2Vec embeddings [Mikolov et al., 2013], to encode each word of a question and a representative entity document. It uses a three layer bi-directional GRU [Cho et al., 2014], which is shared between the question and the review sentence encoder.

Self Attention Layer: It learns shared self-attention (intra-attention) weights [Cheng et al., 2016] for questions and representative entity documents and generates attended embedding representations for both. Let the hidden state of the sequence (question or entity sentence) be given by matrix $\mathbf{H}$ where the ith hidden state is represented by h_i. Then the attended representation (s) of the sequence is given by

$$\mathbf{A} = \text{softmax}(v_a \tanh(\mathbf{W}_a \mathbf{H}^T)) \quad \text{and} \quad s = \mathbf{AH} \tag{4.3}$$

where $\mathbf{A}$ is the attention matrix, $\mathbf{W}_a$ and v_a are attention parameters. We generate attended representations for both, the question as well as, for each sentence from the representative entity document. Let q be the attended representation of the question and let $\mathbf{E}_e$ be attended representation of the entity sentences as a matrix.

Question-Entity Attention (QEA) Layer: In order to generate an entity embedding, it attends over the entity sentence embeddings in matrix E_e with respect to the question [Luong et al., 2015]. This helps identify "important" sentences and the sentence embeddings are then combined based on their attention weights to create the entity embedding (which are thus, question-dependent). The question attended entity representation $\hat{e}_q$ is thus given by:

$$\mathbf{A}_e = \text{softmax}(q \mathbf{W}_E \mathbf{E}_e^T) \quad \text{and} \quad \hat{e}_q = \mathbf{A}_e \mathbf{E}_e \tag{4.4}$$

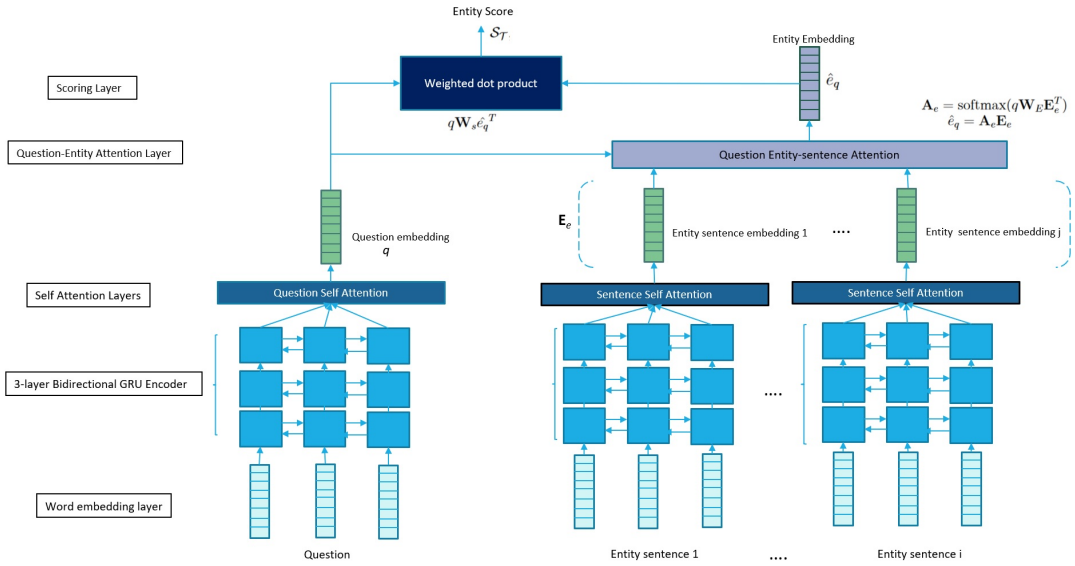


Figure 4.5: Reasoning network used to re-rank candidates shortlisted by the Duet model.

where $\mathbf{A}_e$ is the entity-sentence attention matrix and $\mathbf{W}_E$ is a parameter matrix.

Scoring Layer: Finally, given a question representation q and the entity embedding ($\hat{e}_q$), the model uses a weighted dot product between the two vectors q, $\hat{e}_q$ to generate the final score $\mathcal{S}_\mathcal{T}$, is given by $q\mathbf{W}_s\hat{e_q}^T$ ($\mathbf{W}_s$ is a parameter).

Training: The network is trained using max-margin loss by sampling 10 negative (incorrect answer) entities for each question-answer pair. We improve model training by employing curriculum learning and present harder negative samples returned by a simpler version of the ranker. One method of selecting harder candidates would be to use the gold entity and find entities *similar* to it in some latent space and then present the closest entities as negative samples. In neural settings, candidate embedding space serves as a natural choice for the latent space; negative samples could be generated by sampling from a probability distribution fitted over the distances from the answer embedding.

One could use task-specific embeddings from CrQA to model the candidate space, however, running our trained model on the the test data takes 2.5[18] days. Generating question-specific candidate embeddings for each instance while training (which is nearly 10 times larger) is thus, infeasible. We therefore, generate task-specific embeddings using our CrQA model but without the Question-Entity Attention (QEA) layer, i.e, this model learns question independent entity embeddings by self-attending on the sentence embeddings of each entity (as shown in Equation 4.5):

$$\mathbf{A'}_e = \text{softmax}(v_E\text{tanh}(\mathbf{W'}_E\mathbf{E}_e^T)) \quad \text{and} \quad e = \mathbf{A'}_e\mathbf{E}_e \tag{4.5}$$

[18]Using 4 K-80 GPUs

where $\mathbf{A'}_e$ is the entity-sentence attention matrix and $\mathbf{W'}_E$, v_E are parameters.

Once a model is trained, embeddings can be generated offline and used to generate a probability distribution (per answer entity) for negative sampling. We employ curriculum learning by slowly increasing the selection probability (p_{sel}) of hard negatives up to a maximum of 0.6 as shown in Equation 4.6.

$$p_{sel} = \begin{cases} \frac{epoch-4}{14} * 0.6, & \text{if } 4 \leq epoch \leq 14 \\ 0, & \text{otherwise} \end{cases} \tag{4.6}$$

4.6 Evaluation

We ask the following questions in our experiments: (1) What is the performance of the CsrQA model compared to other simpler baselines for this task?, (2) How does the CsrQA model compare with neural IR and neural QA models?, (3) What is the effect of the size of candidate space?, (4) What is the impact of false negatives?, (5) How important are our other modeling choices, viz: (i) Using Question-Attended Entity Embeddings via the QEA layer, (ii) Employing Curriculum Learning while training.

4.6.1 Models for comparison

We began by trying to adapt traditional reading comprehension QA models such as BiDAF [Seo et al., 2016] for our task, but we found they were infeasible to run – just 1 epoch of training using 10 negative samples per QA pair, and our representative entity documents, took BiDAF over 43 hours to execute on 4 K-80 GPUs. Running a trained BiDAF model on our test data would take even longer and was projected to require over 220 hours. Similarly, we also tried using models based on BERT fine-tuning, but again, it did not scale for our task. In the absence of obvious scalable QA baselines, we compare the performance of CsrQA with other task-specific baselines.

Random Entity Baseline: Returns a random ranking of the candidate answer space.

Ratings Baseline: Returns a global (question-independent) ranking of candidate entities based on user review ratings of entities.

BM25 Retrieval: We index each entity along with its reviews into Lucene.[19] Each question is transformed into a query using the default query parser that removes stop words and creates a disjunctive term query. Entities are scored and ranked using BM25 ranking [Robertson and Zaragoza, 2009]. Note that this baseline is considered a strong

[19] *http://lucene.apache.org/*

baseline for information retrieval (IR) and is, in general, considered better or at par with many neural IR models for typical IR tasks [McDonald et al., 2018].

Review-AVG Model: It uses averaged vector embeddings of the review sentences to represent each document - we use universal sentence embeddings (USE) [Cer et al., 2018] to pre-compute vector representations for each sentence and average them to create a document representation. Questions are encoded using a self-attended bi-directional GRU [Cheng et al., 2016] to generate a question representation. An entity is scored via a weighted dot product between question and document embeddings.

4.6.1.1 Ablation Models

RsrQA: This model highlights the value of the clustering step and the creation of representative entity documents. We replace the clustering phase of our CsrQA model and use 100 randomly-selected review-sentences to represent entities. Thus, this model is effectively CsrQA, but without clustering.

We also tried to create a model that creates document representations by selecting 100 sentences from an entity document by indexing them in Lucene and then using the question as a query. However, this method, understandably, returned very few sentences – the questions (query) are longer than a sentence on average and the lexical gap is too big to overcome with simple expansion techniques. Lastly, if we give the full entity document instead of a representative one, the neural select-rerank model cannot be trained due to GPU memory limitations.

CsQA: This model returns answers by running the neural information retrieval model, Duet, on the clustered representative documents. This model is effectively the CsrQA model but without re-ranking.

CrQA: This model returns answers by running the reasoner directly on the clustered representative documents. Thus, this model does not use neural IR to select and reduce the candidate search space. This model is effectively CsrQA, but without selection.

4.6.1.2 Hyper-parameter Settings

We summarize the hyper-parameter settings of Duet in Table 4.5. For all experiments we set $\delta = 1$ in our max-margin criterion. We used Adam Optimizer [Louizos et al., 2018] with a learning rate of 0.001 for training. The reasoning network (re-ranker) was trained for 5 days on 6 K80 GPUs (approx. 14 epochs) using 10 negative samples for each QA pair. We used 3-layer 128-dimensional bidirectional GRUs to encode questions and review sentences. Input word embeddings were updated during training and USE embeddings returned 512 dimension embeddings.

	Layer/Variable	Parameter Value
	Max document length	1600 words
	Max Query length	200 words
Local Model	Convolution Layer (1D)	Input Channel = 1600, Output Channel = 128, Kernel Size = 1, Stride = 1
	Linear Layer 1	Input Size = 25600, Output Size 128
	Linear Layer 2	Input Size = 128, Output Size = 128
Question Distributed Model	Convolution Layer (1D)	Input Channel = 128, Output Channel = 128, Kernel Size = 3, Stride = 1
	Max Pool	Kernel Size =198, Stride = 198
	Linear Layer 1	Input Size =128, Output Size = 128
Document Distributed Model	Convolution Layer 1 (1D)	Input Size =128, Output Size =128, Kernel Size = 3, Stride = 1
	Max Pool	Kernel Size = 100, Stride = 1
	Convolution Layer 2 (1D)	Input Size = 128, Output Size = 128, Kernel Size = 1, Stride = 1
Scoring Layer	Linear Layer 1	Input Size = 128, Output Size = 128
	Linear Layer 2	Input Size = 128, Output Size =128
	Linear Layer 3	Input Size = 128, Output Size = 1

Table 4.5: Hyper parameter values used in the Duet retrieval model. All layers are separated by ReLU activation units and a dropout layer with 0.5 probability.

4.6.2 Metrics for Model evaluation

The goal of our task is to return an entity (represented by a document) as the answer to a user question. Since our relevance judgements are incomplete and un-ranked, we only assess the relevance of a candidate answer, regardless of whether or not, there could be multiple better ranked answers. We therefore, use Hits@N scores for evaluating the QA system. For a question q, let the set of top ranked N entities returned by the system be Sys_N, and let the correct (gold) answer entities for the question be denoted by set $\mathbb{U}$. We give credit to a system for Hits@N if the sets Sys_N and $\mathbb{U}$ have a non-zero intersection (i.e., Hits@N = $\mathbb{1}((Sys_N \cap \mathbb{U}) \neq \phi)$, where $\mathbb{1}\{\}$ is the indicator function). We report Hits@3, Hits@5 and Hits@30 scores. In addition, we also use the standard mean reciprocal rank (MRR) metric. To compute MRR score we only consider the highest ranked gold answer (if multiple gold answers exist for a question).

Method	Hits@3	Hits@5	Hits@30	MRR
Random	0.32	0.58	3.78	0.007
Ratings	0.37	0.92	3.33	0.007
BM25	6.72	9.98	30.60	0.071
Review-AVG	7.87	11.83	30.65	0.084
RsrQA	10.22	14.63	36.99	0.104
CrQA	16.89	23.75	52.51	0.159
CsQA	17.25	23.01	52.65	0.161
CsrQA	**21.44**	**28.20**	**52.65**	**0.186**

Table 4.6: Performance of different systems including the CsrQA model on our task. Hits@N scores reported in % , (p-value <0.0005).

Candidate Space Size	No. of Questions	CsQA	CrQA	CsrQA
<=1000	631	28.69	30.27	**32.49**
>1000	1542	12.58	11.41	**16.93**

Table 4.7: Test set performance (Hits@3 in %) of ablation systems on questions with different candidate answer space sizes.

4.6.3 Results

Table 4.6 compares CsrQA against other models. We find that all non-neural baselines perform poorly on the task. Even the strong baseline of BM25 retrieval, which is commonly used in retrieval tasks, is not as effective for this dataset. Methods such as BM25 are primarily aimed at addressing challenges of semantic gap while in our task, answers require *reasoning* over subjective opinions in entity documents. We also observe that the performance of the neural model, Review-AVG, is comparable to that of BM25.

The RsrQA model that uses randomly sampled review-sentences, has a low Hits@3 score of 10.22 %. In contrast, both the CsQA and CrQA models, that use the clustered representative entity-documents have higher scores than RsrQA. Our final model CsrQA, has a Hits@3 score of approximately 21.44% (last row of Table 4.6).

We also find that CsrQA does better than CrQA. We hypothesize that since training the reasoner is compute intensive, it is unable to see many hard negative samples for a question even after a long time of training. As a result, it optimizes its loss on the negatives seen during training, but may not perform well when the full candidate set is provided at test-time. However, when the reasoner is used for re-ranking in CsrQA (at test time), the *select* module first shortlists good candidates and the reasoner's job is then just limited to finding the best ones from the small set of relatively good candidates.

We also note that the CrQA model without the QEA layer suffers a significant deterioration in performance as, building question-specific entity embeddings probably helps the model focus on the salient information necessary to answer a question.

Comparing CsrQA & CsQA suggests that, while the scalable matching of Duet is useful enough for filtering candidates, it is not good enough to return the best answer. On the other hand the CsrQA model has a reasoner specifically trained to re-rank a harder set of filtered candidates and hence performs better.

Overall, we find that each component of CsrQA is critical in its contributing towards its performance on the task. Moreover, strong IR only (CsQA) and QA only baselines (CrQA) are not as effective as their combination in CsrQA.

Effect of candidate space size: Table 4.7 breaks down the performance of systems based on size of the candidate space encountered while answering. In questions where the

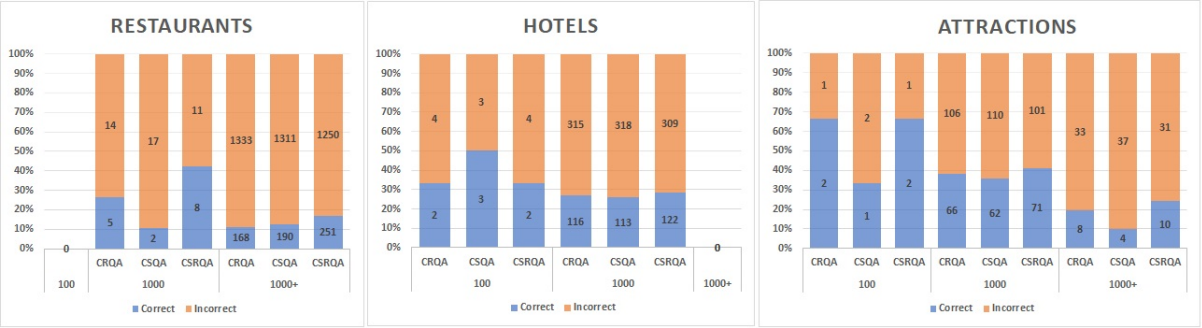


Figure 4.6: Entity class-wise break-up of the number of times (and %) a correct answer was within the top-3 ranks binned based on the size of candidate search space (<100, 100-1000, 1000+ entities) (X-axis).

	Human Scores	Machine Scores
Method	**Hits@3**	**Hits@3**
CrQA	50.0	19.79
CsQA	63.51	22.92
CsrQA	**65.63**	**33.33**

Table 4.8: Performance of different systems including the CsrQA model on our task as measured using human judgements (Human Scores) and gold-reference data (Machine Scores). Hits@N scores reported in %.

candidate space is relatively smaller (<1000), we find CrQA model has slightly better performance than the CsQA model. However, in large candidate spaces we find the CsQA model is more effective in pruning the candidate search space and performs better than the CrQA model. The CsrQA model outperforms both systems regardless of candidate space size, highlighting the benefit of our method.

Performance across different entity-classes: Figure 4.6 shows the number of times the gold answer was in the top-3 ranks for questions from each entity class.[20] The results have been binned based on the size of the candidate space (0-100, 100-1000, 1000+) We find that in hotels and attractions since the search space in most questions isn't as large, both the CsQA and CrQA models have comparable performance. However, using the full CsrQA model still shows considerable improvement (8% relative gain). Overall, we find that the reduction of search space is critical for this task and the use of a scalable shallow neural model to reduce the search space is an effective strategy to improve performance.

Effect of False Negatives: We assess whether metrics computed using the gold-entity answers as reference answers (machine scores) correlate with human relevance judgements (human scores) on the top-3 answers returned by a system. We conduct a blind human-

[20]Recall that each question has its own candidate space

Method	Hits@3	Hits@5	Hits@30	MRR
CrQA without QEA	14.91	19.97	47.58	0.141
CrQA	16.89	23.75	52.51	0.159

Table 4.9: The importance of question-specific entity embeddings generated using the QEA layer in CrQA

study using the CsQA, CrQA and CsrQA models on a subset of 100 questions (300 QA Pairs) from the validation-set. Two human evaluators (κ=0.79) were presented the top-3 answers from both models in random order and were asked to mark each answer for relevance – we ask the evaluators to manually query a web-search engine and assess if each question-recommendation pair (returned by a model) adequately matches the requirements of the user posting that question. As can be seen from Table 4.8, the absolute performance of the systems as measured by the human annotators is higher indicating the presence of false negatives in the dataset. Thus, the machine scores under-report actual performance. To assess the usefulness of the machine scores, we compute pair-wise correlation coefficients between CsrQA, CsQA and CrQA, and find there is moderately positive correlation [Akoglu, 2018] with high confidence between the human judgements and gold-data based measurements for Hits@3 ($\bar{\rho} = 0.39$, p-value<0.0009) and is thus useful to benchmark models.

Error Analysis: We conducted an error analysis of the CsrQA model using the results of the human evaluation. We found that nearly 35% of the errors made were on questions that involved location constraints while, 9% of the errors were due to either budgetary or temporal constraints not being satisfied.

Effect of question specific entity-embeddings: Instead of generating entity embeddings by question attention over entity-sentences (as in Equation 4.4), we could also generate question-independent, self-attended, entity embeddings (e) as given in Equation 4.5.

As can be seen in Table 4.9, the CrQA model without the QEA layer suffers a significant deterioration in performance. We hypothesize that building question-specific entity embeddings, helps the model focus on the salient information necessary to answer a question, and thus helps improve performance.

Effect of Size of Ranking Space: The performance improvement of the CsrQA model over the CrQA model suggests that the re-ranker gets confused as the set of candidate entities increases. We studied the performance of the CsrQA model by varying the number of candidates it had to re-rank. As expected, as we increase the number of candidates available for re-ranking, the Hits@3 scores begin to drop finally settling at approximately 15% (on validation data) when the full candidate space is available (Table

top-k	Hits@3	Hits@5	Hits@30	MRR
10	19.39	25.86	33.88	0.160
20	19.53	26.85	47.33	0.171
30	19.01	26.66	54.32	0.171
40	18.59	26.76	57.24	0.172
50	18.68	26.85	57.95	0.171
60	18.64	25.77	58.66	0.169
80	18.26	25.34	58.94	0.169
100	18.26	25.02	58.75	0.167
Full	14.67	21.43	53.56	0.147

Table 4.10: Performance of CsrQA on the validation data reduces, as the size of candidate space (selected by CsQA) to be re-ranked increases.

4.10). However, we find that the variation in Hits@30 scores is small, indicating that there are only a few candidates (approx. 30-40) per question that the model is most confused about. Thus, since max-margin ranking models can be sensitive to the quality of negative samples, identifying harder candidates and making them available at training, could help models better distinguish between candidates.

4.6.4 Sampling Strategies for Curriculum Learning

As mentioned earlier, we created negatives samples by using the gold entity to find the most similar entities to it, in embedding space. We created the embedding representation of entities using Equation 4.5. We also experimented with the following alternative methods of creating entity embeddings: (i) Using the averaged sentence embeddings of the representative documents (AVG. Emb in Table 4.11) (ii) Doc2Vec [Le and Mikolov, 2014] (iii) Use the ranking returned by CsQA (Duet) to generate negative samples.

As can be seen in the last row of Table 4.11, training CrQA with harder negative samples with curriculum learning helps train a better model. Interestingly, the negative samples derived by using the ranking from Duet [Mitra et al., 2017, Mitra and Craswell, 2019] results in comparable performance but using Duet as *selection* mechanism results in significantly improved performance as shown previously.

Method	Hits@3	Hits@5	Hits@30	MRR
CrQA (No CL)	16.06	22.18	**53.04**	0.155
CrQA (CL) Doc2Vec Emb.	16.38	22.14	52.97	0.149
CrQA (CL) AVG Emb.	16.24	22.14	51.68	0.157
CrQA (CL) Duet Ans.	16.06	21.95	52.88	0.155
CrQA (CL) Task Emb.	**16.89**	**23.75**	52.51	**0.159**

Table 4.11: Curriculum learning (CL) with different entity embedding schemes

4.7 Summary

In order to create a model for answering questions seeking POI-recommendations, we first harvested a dataset of over 47,000 QA pairs and then used it to formulate the novel task of returning a POI (answer) by reasoning over a collection of unstructured reviews describing entities. In contrast to the previous chapter which used supervision to build a query parser, here we directly train an answering system using QA pairs. Due to the nature of questions and the review documents, one of the biggest challenges in this dataset is that of scalability. Our task requires processing 500 times more documents per question than most existing QA tasks, and individual documents are also much larger in size. We thus presented a cluster-select-rerank architecture based method that brings together neural IR and QA models, and serves as a strong baseline for this task. It first clusters text for each entity to identify exemplar sentences describing an entity. It then uses a scalable neural information retrieval (IR) module to select a set of potential entities from the large candidate set. A reranker uses a deeper attention-based architecture to pick the best answers from the selected entities. While our final system registers a 25% relative improvement over other simpler baseline models, a correct answer is in top-3 for only 21% of the questions, which points to the difficulty of the task. We believe that further research on this task will significantly improve the state-of-the-art in question answering. Our error analysis found that nearly 35% of the questions with errors involved reasoning on location constraints. Therefore, in the next chapter, we attempt to better answer questions that express location constraints. Resources from this chapter are available at http://ibm.biz/TourismQA.

Chapter 5

Improving QA with Spatio-Textual Reasoning

In the previous chapter we developed a QA system, that used entity reviews to answer POI-recommendation questions. However answering such questions can also require models to use other knowledge, such as location data. Consider the example in Figure 5.1, which shows a real-world[1] Points-of-Interest (POI) seeking question. Answering such a recommendation question is a challenging problem as, it not only requires reasoning over a text corpus describing potential restaurants (eg. reviews), but it also requires resolving spatial constraints ("near Hotel Florida") over the physical location of a restaurant. In addition, the question is also under-specified and subjective (eg, "dont have to venture too far") making the spatial-inference task harder.

In this chapter, we present our joint spatio-textual QA model for returning answers to questions that require textual as well as spatial reasoning. Note that the word "textual" here does not refer to the fact that questions are textual (which indeed they are); we,

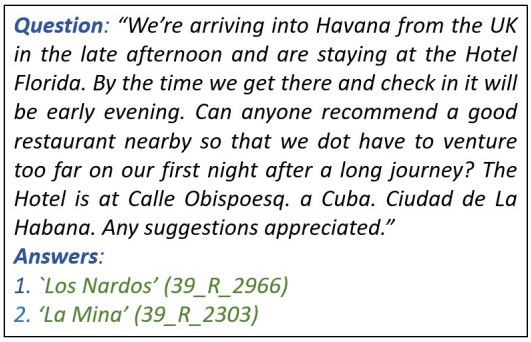


Figure 5.1: A sample POI recommendation question from our dataset created in Chapter 4. The answers correspond to POI IDs of the form <city_id >_<POI type>_<number>.

[1]https://bit.ly/2zIxQpj

instead use "textual reasoning" to imply that, to answer questions, a model has to also reason over a textual corpus (in this case, a corpus of POI reviews). We first develop a modular spatial-reasoning network that uses geo-coordinates of location names mentioned in a question, and, of candidate answer entities, to reason over only spatial constraints (if any). It learns to associate contextual *distance-weights* with each location-mention in the question – these weights are combined with their respective spatial-distances from a candidate answer, to generate a 'spatial relevance' score for that answer. We then combine the spatial-reasoner with the CrQA textual QA system (Chapter 4) which answers questions by reasoning over entity reviews, to develop a joint spatio-textual QA model. To the best of our knowledge, we are the first to develop a joint QA model that combines reasoning over external geo-spatial knowledge along with textual reasoning.

5.1 Contributions

Our work in this chapter makes the following contributions.

- We develop a spatial-reasoner that uses geo-coordinates of locations and POIs to reason over spatial constraints specified in a question.
- We demonstrate, using a simple artificial dataset, that our spatial-reasoner is not only able to reason over "near", "'far" constraints but is also able to determine location references that are not useful for reasoning (Eg: a location reference mentioning where a user last went on vacation).
- We develop a spatio-textual QA model, which fuses spatial knowledge (geo-coordinates) with textual knowledge (POI reviews) using sub-networks designed for spatial and textual reasoning.
- We demonstrate that our joint spatio-textual model performs significantly better than models employing only spatial- or textual-reasoning. It also obtains state-of-the-art results the POI-recommendation dataset described in the previous chapter, with substantial improvement in answering location questions.[2]

[2]Joint work with Shashank Goel, who contributed significantly to the development and implementation of spatial-reasoner, as well as the maintenance of the source code repositories. The part of the work done by him appeared in his B.Tech thesis.

5.2 Related Work

Recently, there has been work on QA models that fuse knowledge from multiple sources; for example, by combining data from knowledge bases with textual passages [Xia et al., 2019a, Bi et al., 2019], or incorporating multi-modal data sources [Guo et al., 2018, Vo et al., 2019]. But, we do not know of systems that fuse geo-spatial knowledge with text. In addition, there exist several geo-spatial IR systems (eg, [Santos and Cabral, 2009, Scheider et al., 2020]), however, to the best of our knowledge, none of them perform joint-reasoning over geo-spatial and textual knowledge sources. Our work is related to four broad areas of question answering and information retrieval:

Geographical Information Systems: There is significant prior work on Geographical Information systems where standard IR models are augmented with spatial knowledge [Ferrés Domènech, 2017, Purves et al., 2018]. Models have been developed to address challenges in adhoc-retrieval tasks with locative references [Gey et al., 2006, Mandl et al., 2008, Santos and Cabral, 2009]. However, such models deal primarily with inference problems in toponyms (eg, "Beijing is located in China"), location disambiguation and use of topographical classes (eg, "Union lake is a water-body") etc. Methods for IR involving locative references use three strategies (i) a pipeline of filtering based on spatial information followed by text-based IR (ii) a pipeline of filtering based on text-based IR followed by ranking based on geo-spatial ranking or coverage, and (iii) a weighted or linear combination of two independent rankings [Leidner et al., 2020]. Our work builds on the third strategy by jointly training a model with both geo-spatial and textual components. To the best of our knowledge, joint reasoning over text and geo-spatial data has not been investigated in geographical IR literature.

Geo-Spatial Querying: There has been considerable work in research areas of geo-parsing (toponym discovery and disambiguation) [Kew et al., 2019], geo-spatial query processing over structured or RDF knowledge bases (KB) [Vorona et al., 2019, Scheider et al., 2020], geocoding and geo-tagging documents [De Rassenfosse et al., 2019, Lim et al., 2019, Huang and Carley, 2019] etc. However, such querying methods require KB & task-specific annotations for training and are thus specialized in application and scope [Scheider et al., 2020].

Numerical Reasoning for Question Answering: Spatial reasoning in our task is effectively a form of numerical reasoning over distances between location mentions in a question and a candidate entity (POI). Recently introduced tasks such as DROP [Dua et al., 2019] and QuaRTz [Tafjord et al., 2019] require reasoning that includes addition, subtraction, counting, etc. for answering reading comprehension style questions. Other tasks such as MathQA [Amini et al., 2019] and Math-SAT [Hopkins et al., 2019] present

high school and SAT-level algebraic word problems.

Models developed for numerical reasoning tasks such as NAQANet [Dua et al., 2019] and NumNet [Ran et al., 2019] reason over the explicit mentions of numerical quantities within a question or passage. In contrast, the questions in our task do not explicitly mention geographical coordinates, and also do not contain all the information required for numerical reasoning (since the distances need to be computed with respect to a candidate answer under consideration). Further, in contrast to algebraic word problems and numerical reasoning questions, answers in the POI-recommendation task are also heavily influenced by text-based reasoning on subjective POI-entity reviews.

Points-of-Interest (POI) Recommendation: Existing models for POI recommendation typically rely on the presence of structured data, including geo-spatial coordinates. Queries may be structured or semi-structured and can consist of both spatial as well as textual arguments. Textual arguments are usually associated with the structured attributes or may serve as filters. Approaches include efficient indexing for 'spatial' and 'preference' features along with specialized data-structures as IR-Trees, [Cong et al., 2009, Zhang et al., 2016, Tsatsanifos and Vlachou, 2015, Li et al., 2016], methods based on Matrix Factorization [Yiu et al., 2007] for user-specific recommendations, click-through logs used for recommendations from search engines [Zhao et al., 2019a] etc.

This chapter is related to our work in Chapters 3 and 4. We had developed two approaches for answering MSRQs: semantic parsing of unstructured user questions to query a semi-structured knowledge store [Contractor et al., 2020], and the use of trainable neural model operating over a corpus of unstructured reviews to represent POIs [Contractor et al., 2019]. Neither of these approaches explicitly reason on spatial constraints, even though the questions contain them.

5.3 Spatio-Textual Reasoning Network

Answering a question as in Figure 5.1 requires reasoning on spatial knowledge (for aspects of location mentioned in the question) and POI reviews (for other preferences/constraints specified in the question). Our model design (Figure 5.2) is motivated by this reasoning - we have a geo-spatial reasoner which uses the question and spatial knowledge (Section 5.3.1), a textual reasoner based on the work in Chapter 4 [Contractor et al., 2019] that uses the question and POI reviews (Section 5.3.2), and a joint layer (Section 5.3.3) that combines the reasoning from these two sub-components to do *joint* reasoning.

Given a POI-recommendation question q (as in Figure 4.1) its target class t (for example, restaurant), the city c (for example, Havana), a candidate space $\mathbb{E}_c^t$ of target-

class entities (POIs) for each corresponding city, and a collection of reviews $\mathbb{R}_e$ describing each entity e, and a knowledge base $\mathbb{L}$ containing the geo-spatial co-ordinates of each entity, the goal of our task is to score each candidate with respect to a question, for relevance.

5.3.1 Geo-Spatial Reasoner

Our geo-spatial reasoner consists of the following components: (1) **Distance-aware Question Encoder** – to encode questions along with geo-spatial distances between location mentions (in the question) and a candidate entity, (2) **Distance Reasoning layer** – to enable reasoning over geo-spatial distances with respect to the spatial constrains mentioned in the question, (3) **Spatial Relevance Scorer** – to score and rank candidates for spatial-relevance.

Distance-aware Question Encoder: We generate question representations by using embedding representations of their constituent tokens along with embedding representations of their location-mentions. A question token can be represented by traditional word-vector embeddings, or contextual embeddings such as BERT [Devlin et al., 2019]. Each token representation is further appended with a two dimensional one-hot encoding representing Begin (**B**), Intermediate (**I**) or Other (**O**) labels, indicating the presence of location tokens identified using a location tagger (See Figure 5.2). Thus, for the question, "*Find me a mexican restaurant within walking distance from the Eiffel Tower*", all token positions except those corresponding to '*Eiffel*' and '*Tower*' have *B-I* vectors with two zeros – the token position corresponding to '*Eiffel*' will have the first bit on, while the position corresponding to '*Tower*' will have the second bit on. The *B-I* labels help the model recognize a single continuous location-mention. In addition, we concatenate the distance[3] of the candidate entity e, from a location-mention to each location-token representation. For example, if the candidate entity being considered by the system is '*Chipotle Mexican Grill, 2 Rue Linois, 75015 Paris*' with geo-coordinates $(lat_e, long_e)$, then the distance between this restaurant and the '*Eiffel Tower*' $(lat_{Eiffel}, long_{Eiffel})$ is appended wherever the *B-I* vector corresponding to this location-mention is non-zero (we append 0 everywhere else). Thus, the question representations are distance-aware and candidate-dependent.

Formally, let the token embedding representations in a question be given by v_i $(v_1 \dots v_i \dots v_{|Q|})$, where $|Q|$ is the length of a question. Let the distance between the kth location-mention lm_k and c be denoted by d_k. Further, let $POS(lm_k)$ be a function that returns the set of position indices occupied by location mention lm_k, i.e. it returns

[3]Manhattan Distance

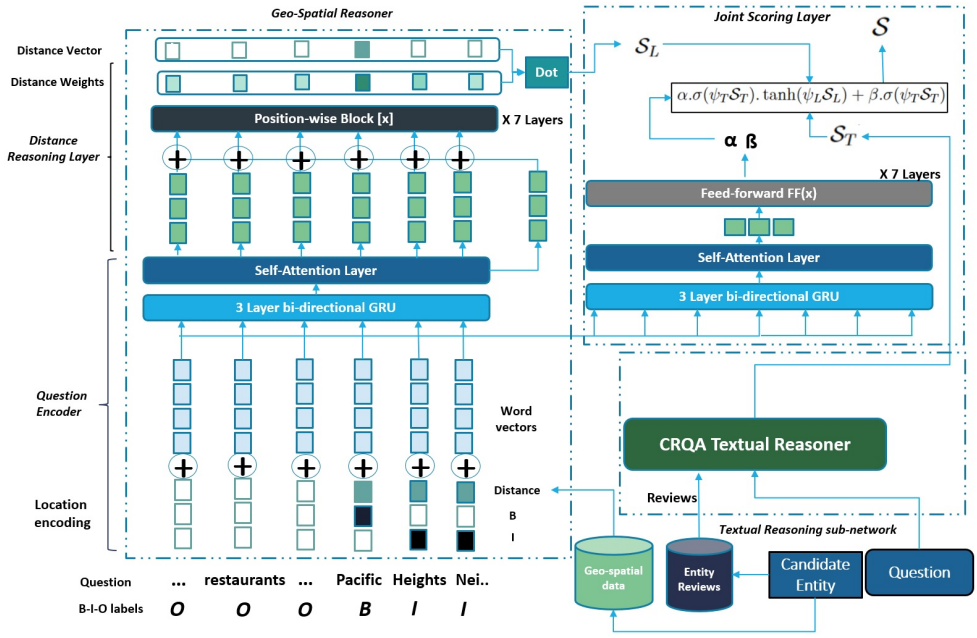


Figure 5.2: Spatio-Textual reasoning network consisting of (i) Geo-Spatial Reasoner (ii) Textual-Reasoning subnetwork (iii) Joint Scoring Layer

the set of position indices of question tokens that have been assigned the B or I label from the B-I encoding for location mention lm_k, $(POS(lm_k) \subset \{1, \ldots, |Q|\})$. We create a $|Q|$-dimensional distance vector $\mathbf{d}'$ where each element d'_i of the vector is given by:

$$d'_i = \begin{cases} d_k & \text{if i} \in POS(lm_k) \\ 0, & \text{otherwise} \end{cases} \tag{5.1}$$

Let the one-hot vector (two dimensional) of the B-I labels for the ith position be g_i. The input question embedding t_i, $(t_1 \ldots t_i \ldots t_{|Q|})$ is then given by:

$$t_i = concat[v_i, d'_i, g_i] \tag{5.2}$$

We encode the question using a bi-directional GRU [Cho et al., 2014] which results in output states q_i, q_1 ...q_i ...$q_{|Q|}$.

Distance-Reasoning Layer (DRL): We first used a series of down-projecting feed-forward layers applied to the output state of the GRU, to generate the final score for each candidate, but we found this was not effective (Section 5.4.1.2). We therefore include a component designed for distance-reasoning referred to as the 'Distance Reasoning Layer' which uses the representations generated by the distance-aware question encoder.

A model could score candidate-entities for relevance if, for each location mentioned

in the question, it is able to (i) learn *whether* a location-mention needs to be considered for answering, and (ii) learn *how* a location-mention needs to be used for answering. Our design of the DRL is motivated by this insight – it learns a function which, for each location-mention lm_k in the question, outputs a *distance-weight* w_k. Here, w_k captures the contribution of the spatial-distance between lm_k and the candidate entity c, under the constraints mentioned in the question. For instance, a question may include location-mentions that could be involved in simple 'near' or 'far' constraints or other complex constraints such as "within driving distance" or "within walking distance" etc. The DRL layer uses the distance-aware question encoding to understand the nature of the constraint being expressed, as well as, figure out how to compute distance-reasoning weights to express those constraints.

To compute distance-weights, we use a series of position-wise feed-forward blocks [Vaswani et al., 2017] that consist of a linear layer with ReLU activation applied at each output position of the Question Encoder:

$$q_i^l = Block_l(q_i^{l-1}) = max(0, A_l q_i^{l-1} + b_l) \tag{5.3}$$

where q_i^l is the output of the Block layer at layer l, A_l is a weight matrix and b_l the bias term.

The initial block input uses the output state of the GRU (q_i) concatenated with the final hidden state ($q_{|Q|-1}$). Thus, the output q_i^1 from the application of the first block layer, corresponding position i in the input is given by:

$$q_i^1 = Block_1(concat[q_i, q_{|Q|}]) \tag{5.4}$$

The blocks apply the same linear transformations at each position but we vary the parameters across layers (see Appendix B). The final layer gives us a single dimension output for each position resulting in an $|Q|$-dimensional vector $\mathbf{r}$ $(r_1...r_i..r_{|Q|})$.

Let $\bar{B}$ be a $|Q|$-dimensional multi-hot vector based on the position indices that have been assigned only the B label from the *B-I* encoding used in the input layer (an element of $\bar{B}$ is 1 whenever it corresponds to a position index indicating the *start* of a location mention in a question). The distance-weight vector $\mathbf{w}$ for a question is given by:

$$\mathbf{w} = tanh(\mathbf{r} \odot \bar{B}) \tag{5.5}$$

Since an element in $\bar{B}$ is 1 at the beginning position of every *unique* location mention in the question, $\mathbf{w}$ thus contains weights corresponding to each unique location mention.

Spatial Relevance Scorer: We use the distance-weights for scoring - the final score $\mathcal{S}_L$

of a candidate c is given by:

$$\mathcal{S}_L = \mathbf{w}\dot{\mathbf{d}}' \tag{5.6}$$

Note that since we concatenate the distance values along with token embeddings while encoding locations as part of the Question Encoder (Equation 5.2), it helps learn distance weights w which are dependent on the distance value as well as the semantic constraints (e.g: 'within walking distance' vs 'within driving distance', 'near' vs 'far') present in the question. Thus, the spatial relevance score is not just a simple linear combination of distances and makes the model representationally more powerful (see experiments in Section 5.4.2). We refer to the Geo-Spatial Reasoner as SPNET for brevity in the rest of the chapter.

5.3.2 Textual-Reasoning Sub-network

We use the CRQA [Contractor et al., 2019] model from Chapter 4 as our textual-reasoning sub-network. Recall that it consists of a Siamese-Encoder [Lai et al., 2018], which uses question representations to attend over entity-review sentences and generate question-aware entity-embeddings. These entity embeddings are combined with question representations to generate an overall relevance score ($\mathcal{S}_T$). For scalability, instead of using full review documents, the model uses a set of representative sentences from reviews after clustering them in USE-embedding space [Cer et al., 2018]. As before, we continue to use k-means to cluster sentences in USE embedding space. We set k=10, and select 10 sentences per cluster, thus creating a document with 100 sentences or less, to represent an entity. In order to build a model that is capable of joint spatio-textual reasoning, our model learns question-specific combination weights that combine textual and spatial-reasoning scores.

5.3.3 Joint Scoring Layer

Let the score generated by the textual-reasoner be $\mathcal{S}_T$ and let the rescaling weights for $\mathcal{S}_T$ and $\mathcal{S}_L$ (spatial-reasoning score) be ψ_T and ψ_L respectively. Then, the overall score $\mathcal{S}$ is given by:

$$\alpha.\sigma(\psi_T\mathcal{S}_T).\tanh(\psi_L\mathcal{S}_L) + \beta.\sigma(\psi_T\mathcal{S}_T)$$

where σ is the Sigmoid function and α,β are combination weights. The weights are computed by returning a two dimensional output (corresponding to each weight), after a series of feed-forward operations on the self-attended representation [Cheng et al., 2016], of the question using the outputs of a Question Encoder with the same architecture as in SPNET (see Appendix B for hyperparameters). Note that the first term of scoring

equation uses $\mathcal{S}_L$ as a *selector* – for questions where there are no locations mentioned, the spatial score of a question with no location-mentions will be 0 (due to the equation for $\mathbf{w}$). This lets the model rely only on textual scores for these cases.

Training: We first independently[4] pre-train the spatial-reasoning sub-network using max-margin loss, where the correct answer entity is scored higher than a random negative entity. During this pre-training step any entity within a 100m radius from the gold-answer entity, is considered a correct answer (this helps the spatial-reasoner become sensitive to distances). We then initialize the parameters of the spatial-reasoner (in the joint-model) using the pre-trained weights and train the full network consisting of both, the spatial and textual reasoning sub-networks, using the curriculum-learning based setup described in Section 4.5.3. We find that in the absence of pre-training of the spatial-reasoning sub-network, the joint-model tries to ignore spatial reasoning.

5.4 Evaluation

We first present a detailed study of the spatial-reasoner using a simple artificially generated dataset. This allows us to probe and study different aspects of spatial-reasoning in the absence of textual reasoning. We then present our experiments with the joint spatio-textual model using the real-world POI-recommendation QA dataset collected in the previous chapter.

5.4.1 Detailed Study: Geo-Spatial Reasoner

We conduct this study on a simple artificial dataset, that was generated using linguistically diverse templates. These templates specify spatial constraints and location names, chosen at random from a list of 200,000 entities across 50 cities.

Question 1: "Suggestions for a place of tourist interest close to **Hilal Park** *and* **Mahatta Palace Museum***"*
Question 2: "Hey I will be staying at **Pinati***. Please suggest a coffee shop far from the* **Cellar Bar***."*
Question 3: "Please propose an eatery close to **Udipi Palace** *but not in the neighbourhood of* **Nico***. Thanks!"*

Figure 5.3: Sample questions from the artificial dataset. The dataset has questions from three categories: (1) close to set X, (2) far from set X (3) Combination.

[4]the textual-reasoner is not trained

Models	Hits@3	Hits@5	Hits@30	MRR	D_g
			Close to Set X		
SPNet w/o DRL	62.60	66.00	79.00	0.608	2.88
SPNet	90.20	**92.80**	**97.60**	0.873	0.86
BERT SPNet w/o DRL	63.60	67.60	82.60	0.616	3.68
BERT SPNet	**91.40**	**92.80**	97.20	**0.896**	**0.78**
			Far from Set X		
SPNet w/o DRL	89.00	90.80	96.40	0.858	15.24
SPNet	**98.00**	**98.40**	**99.20**	0.975	13.88
BERT SPNet w/o DRL	90.80	92.00	95.80	0.881	15.32
BERT SPNet	97.80	98.00	98.80	**0.978**	**13.87**
			Combination		
SPNet w/o DRL	23.40	28.00	50.60	0.229	9.72
SPNet	52.80	60.20	82.00	0.486	3.90
BERT SPNet w/o DRL	26.80	32.60	59.00	0.242	12.96
BERT SPNet	**59.20**	**65.80**	**86.20**	**0.551**	**3.02**
			Aggregate		
SPNet w/o DRL	58.33	61.60	75.33	0.565	9.28
SPNet	80.33	83.80	92.93	0.778	6.21
BERT SPNet w/o DRL	60.40	64.07	79.13	0.579	10.65
BERT SPNet	**82.80**	**85.53**	**94.07**	**0.808**	**5.89**

Table 5.1: Results of SPNet on the artificial spatial-questions dataset (t-test p-value $< 10^{-33}$ for Hits@3)

5.4.1.1 Artificial Dataset

Template Classes: We create templates that can be broadly divided into three types of proximity queries based, on whether the correct answer entity is expected to be: (1) close to one or more locations (mentioned in the question), (2) far from one or more locations, (3) close to some and far from others (combination). We create different templates for each category with linguistic variations. Figure 5.3 shows a sample question from each category. See Appendix B for more details, including the list of templates.

Use of distractor-locations: In order to make the task more reflective of real-world challenges we also randomly insert a *distractor* sentence that contains a location reference which does not need to be reasoned over (e.g the location "Pinati" in Question 2 in Figure 5.3).

Gold-entity generation: The gold answer entity is uniquely determined for each question based on its template. For example, consider a template T, "*I am staying at $A! Please suggest a hotel close to $B but far from $K.*" The score of a candidate entity X is given by $dist_T(X) = -(dist(X, B) - dist(X, K))$ (distances from B needs to be reduced, while distance from K needs to be higher). A is a distractor. The candidate with the $max(dist_T(X))$ in the universe is chosen as the gold-answer entity for that question. We use the geo-coordinates of locations to compute the distance.

Dataset Statistics: The train, dev and test sets consist of 6000, 1500 and 1500 questions respectively generated using 48 different templates, split equally across all 3 template categories. Each question consists of location-names from only one city and thus the candidate search space for that question is restricted to that city. The average search space for each question is 1250, varying between 10-16200 across cities. The dataset includes questions containing distractor-locations (52.33% of dataset) distributed evenly across all template classes.

5.4.1.2 Results

We study SPNET using the artificial dataset to answer the following questions: (1) What is the model performance across template classes? (2) How does the network compare with baseline models that do not use the DRL? (3) How well does the model deal with distractor-locations, i.e locations not relevant for the scoring task? For all experiments in this section we use perfectly tagged location-mentions.

Metrics: We study the performance of models using Hits@N (N=3,5,30) which requires that any one of the top-N answers be correct, Mean Reciprocal Rank (MRR) and the average distance of the top-3 ranked answers from the gold-entity $Dist_g$. $Dist_g$ is helpful in quantifying the spatial goodness of the returned answers (lower is better).

We use the following models in our experiments: (i) SPNET (ii) SPNET without DRL (iii) BERT-SPNET (iv) BERT-SPNET without DRL. Models without DRL use the final hidden states of the Question Encoder and a series of down-projecting feed-forward layers to generate the final score.

Performance across template classes: As can be seen in Table 5.1, all models perform the worst on the template class that contains a combination of both 'close-to' and 'far' constraints. Models based on SPNET perform exceeding well on the 'Far' templates because the difference between the $dist_T(X)$ scores of the best and the second best candidate is almost always large enough for every model to easily separate them.

Importance of Distance-Reasoning Layer: As can be seen in Table 5.1 the performance of each configuration (with and without BERT) suffers a serious degradation in the absence of the DRL. Recall, that all models have access to spatial knowledge in their input layer via the question encoding. This indicates that the DRL is an important component required for reasoning on spatial constraints. To further assess whether our model is able to do distance reasoning, we computed the correlation between ranking-by-distances (appropriate ranking order for each template-class) and SPNET's ranking on the artificial dataset. We found the correlation to be a high 0.97 suggesting that the model is able to use physical distance to compute the best answer.

	Without Distractors		With Distractors	
Models	**Hits@3**	**MRR**	**Hits@3**	**MRR**
SPNET	82.58	0.800	78.30	0.758
BERT SPNET	**84.13**	**0.820**	**81.60**	**0.797**

Table 5.2: Performance of spatial-reasoning networks degrades in the presence of location-distractor sentences.

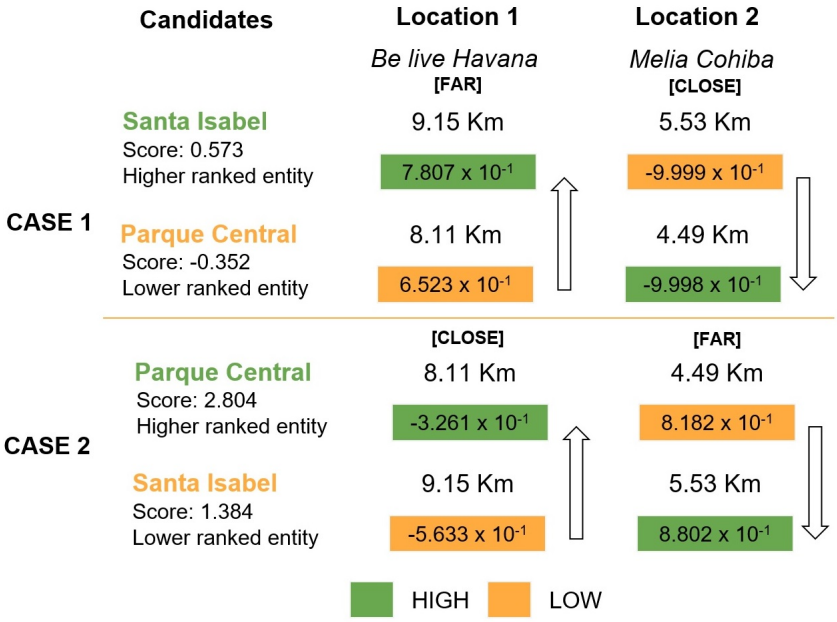


Figure 5.4: Probing study of the Distance Reasoning Layer (DRL) using the question: *"I came from Tropicoco today. Any nice ideas for a coffee shop [`far from/close to`] 'Be Live Havana' but [`close to/far from`] 'Melia Cohiba'?"*. The coloured boxes indicate the relative magnitude of weights assigned; each candidate entity assigns a *higher* weight (column-wise comparison), as compared to the other candidate, on the distance property it is most likely to benefit from, with respect to the spatial-constraint

Effect of distractor-locations: We report results on two splits of the test set: Questions with and without distractor-locations. We report the aggregate performance over all template classes due to space constraints. As can be seen in Table 5.2, models suffer a degradation of performance in the presence of distractor-locations. We hypothesize that this is because the reasoning task becomes harder; models now need to also account for location-mentions that do not need to be reasoned over.

Probing Study: We conduct a probing study (Figure 5.4) on SPNET to get some insights into the reasoning process employed by the trained network. We use a question that has both 'near' and 'far' constraints (case 1) and then interchange the constraints (case 2). In both the cases we study the corresponding distance-weights assigned to the location-mentions with respect to two candidates "Santa Isabel" and "Parque Central".

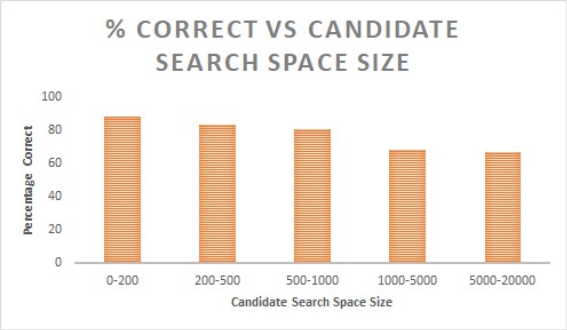


Figure 5.5: Performance of SPNET decreases with increase in universe size.

Consider the first case; as can be seen, each candidate entity assigns a *higher* weight (column-wise comparison) as compared to the other candidate, on the distance property it is most likely to benefit from, with respect to the spatial-constraint. For example, when the spatial-constraint requires an answer to be *close* to "Melia Cohiba", the candidate "Parque Central" assigns a higher weight to this location as compared to candidate "Santa Isabel", since "Parque Central" has a *smaller* distance value to this location. On the other hand, with respect to the "*far*" constraint, candidate "Santa Isabel" has a *larger* distance value from "Be live Havana" as compared to candidate "Parque Central", thus assigning a *higher* distance weight for this location-mention.

When we interchange the constraints (Case 2) we see the same pattern and the comparative weight trends (at each location-mention) invert due to inversion of spatial-constraints. This suggests, that DRL is learning to transform the inputs and generate weights based on the spatial constraint at hand.

Effect of Candidate Space Size: We find that as the candidate space increases, the errors made by the SPNET model also increases (Figure 5.5). Approximately 25% of the test-set contains questions with large (> 1000) candidate spaces.

Effect of the No. of Location-mentions: The complexity of the spatial-reasoning task increases as the number of location-mentions (including distractor-locations) in the question increase. As can be seen in Figure 5.6 , as the number of location-mentions in a question increases, the performance of the model drops.

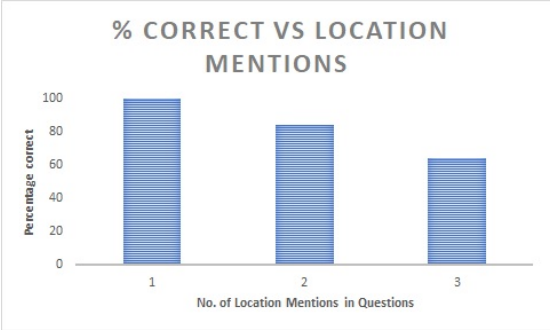


Figure 5.6: Performance of SPNet decreases with increase in the number of location mentions in the question.

5.4.2 Spatio-Textual Reasoning Network

For the joint model, we investigate the following research questions: (i) Does joint spatio-textual ranking result in improved performance over a model with only spatial-reasoning or only textual-reasoning? (ii) How do pipelined baseline models that use spatial re-ranking perform on the task? (iii) Does distance-aware question encoding help in spatio-textual reasoning? (iv) Is the spatio-textual reasoning model more robust to distractor-locations as compared to baselines? (v) What kind of errors does the model make?

Dataset: We use the dataset of Tourism Questions created in Section 4.2. As described previously, it consists of over 47,000 real-world POI question-answer pairs along with a universe of nearly 200,000 candidate POIs; questions are long and complex, as presented in Figure 5.1, while the recommendations (answers) are represented by an ID corresponding to each POI. Each POI comes with a collection of reviews and meta-data that includes its geo-coordinates. The training set contains nearly $38,000$ QA-pairs and about $4,200$ QA-pairs each in the validation and test sets. The average candidate space for each question is $5,300$. We use the non-BERT based SPNet subnetwork in these experiments due to challenges of scale on this dataset.[5]

Location Tagging in Questions: In order to get mentions of locations in questions, we manually label a set of 425 questions from the training set for location mentions.

	Precision	**Recall**	**F1**
Micro Average	87.59	87.56	87.58
Macro Average	88.24	87.83	88.03

Table 5.3: Performance of the BERT-BiLSTM CRF for tagging locations on a small set of 75 questions.

[5]Recall that CrQA was also not based on BERT due to this reason

	Location		Non-location	
Dataset	**Questions**	**QA pairs**	**Questions**	**QA pairs**
Train	9,617	21,396	10,342	22,150
Dev	1,065	2,209	1,054	1,987
Test	1,086	2,198	1,087	2,144

Table 5.4: Distribution of questions with location-mention across train, dev & test sets.

We then use a BERT-based sequence tagger[6] trained on this set to label locations. The tagger has a macro-$F1$ of 88.03 (see Table 5.3). It is possible that a question may contain **only** distractor-locations, i.e., locations-mentions that do not need to be reasoned over the answering task.

Once the location-mentions are tagged, we remove the punctuations and stopwords from the tagged-location span. We then query the Bing Maps Location API[7] using the location-mention along with the city (known from question meta-data) to get the geo-tags. To reduce noise in geo-tagging, we ignore the location-mention if the remaining text has a length of less than 4 characters or is identified as a popular acronym, continent, country, city or state (lists from Wikipedia). We further reduce noise by ignoring a location mention: (1) if no results were found from BING, or (2) If the geo-tag is beyond 40km from the city center. We found that on the set of 75 questions referred to in Table 5.3, this processing step resulted in a macro-precision of 100 and a macro-F1 of 78.97. Further, the location-mention geo-tagging accuracy on a small set of 83 location-mentions was found to be 96%.

We label all questions in the full dataset using this tagger, resulting in approximately 49.54% of the QA pairs containing at least one location-mention (see Table 5.4). In all our experiments, we use the Manhattan distance as our distance value, because it is generally closer to real-world driving/walking distance within a city, as opposed to straight-line distance.

5.4.2.1 Baselines

Apart from the textual-reasoning model CrQA we also use the following baselines in our experiments:

Sort-by-distance (SD): Given a set of tagged-locations in a question and their geo-coordinates, rank candidate entities by the *minimum* distance from the set of tagged locations.

SPNet: Use only the spatial-reasoning network for ranking candidate entities using their

[6]github.com/codedecde/BiLSTM-CCM/tree/allennlp

[7]https://bit.ly/36Vazwo

	Location Questions				
Models	**Hits@3**	**Hits@5**	**Hits@30**	**MRR**	**Dist$_g$**
SD	2.49	3.41	14.29	0.029	3.07
SPNET	1.47	2.11	8.47	0.019	2.97
CRQA	14.83	21.27	50.65	0.143	3.41
CRQA→SD	13.73	19.26	50.65	0.125	**2.23**
CRQA→SPNET	10.13	15.65	50.64	0.104	2.47
Spatio-textual CRQA	**18.32**	**25.69**	**56.17**	**0.168**	2.62

Table 5.5: Comparison of the joint Spatio-Textual model with baselines on questions that have location mentions (t-test p-value< 0.009)

	Location Questions					Non-location Questions			
Models	**Hits@3**	**Hits@5**	**Hits@30**	**MRR**	*Dist$_g$*	**Hits@3**	**Hits@5**	**Hits@30**	**MRR**
CRQA	14.83	21.27	50.65	0.143	3.41	18.95	26.22	54.37	0.177
Spatio-Textual CRQA	**18.32**	**25.69**	**56.17**	**0.168**	**2.62**	**20.42**	**26.77**	**56.49**	0.18
Spatio-textual CRQA without (w/o) distance-aware QE	16.85	23.39	53.04	0.159	2.84	20.06	26.86	56.49	**0.185**

Table 5.6: Comparison of Spatio-Textual CRQA (with and without (w/o) distance-aware question encoding) and CRQA (t-test p-value < 0.03 for Hits@3)

geo-coordinates. No textual-reasoning performed.

CrQA → SD: Rank candidates using CRQA and then re-rank the top-30 answers using SD.

CrQA → SPNet: Rank candidates using CRQA and then re-rank the top-30 answers using SPNET.

Training: We pretrain SPNET on this dataset by allowing entities within a radius of 100m from the actual gold-entity to be considered as gold (only for pretraining). To train the joint network we initialize model parameters learnt from component-wise pretraining of both SPNET as well as CRQA.

	Full Set			
Models	**Hits@3**	**Hits@5**	**Hits@30**	**MRR**
CRQA	16.89	23.75	52.51	0.159
Spatio-Textual CRQA	**19.37**	**26.23**	**56.33**	**0.175**
Spatio-textual CRQA (w/o distance-aware QE)	18.45	25.13	54.76	0.172

Table 5.7: Comparison of Spatio-Textual CRQA (with and without (w/o) distance-aware question encoding) and CRQA on the full set

5.4.2.2 Results

We present our experiments on two slices of the test-set – questions with tagged location-mentions (called *Location-Questions*) and those without any location mentions (*Non-Location Questions*). As can be seen in Table 5.5 sorting-by-distance (SD) performs very poorly indicating that simple methods for ranking based on entity-distance do not work for such questions. Further, the poor performance of SPNET also indicates that the task cannot be solved just by reasoning on location data.

In addition, pipelined re-ranking using SD or SPNET over the textual reasoning model decreases the average distance ($Dist_g$) from the gold-entity but does not result in improved performance in terms of answering (Hits@N) indicating the need for spatio-textual reasoning. Finally, from Tables 5.5 & 5.6 we note that the spatio-textual model performs better than its textual counterpart on the Location-Questions subset, while continuing to perform well on questions without location mentions.

Effect of distance-aware question encoding: In order to demonstrate the importance of distance-aware question encoding, we present an experiment where we remove the distance values from the input encoding. Thus, Equation 5.2 changes to $t_i = concat[v_i, g_i]$. As Tables 5.6 and 5.7 show, the performance of the Spatio-Textual CRQA model in the absence of distance-aware encoding drops (last row), but it still performs better than the text-only CRQA model (first row). This indicates that the distance-aware question encoding helps learn better distance weights for spatio-textual reasoning.

Effect of distractor-locations: As mentioned earlier, we use a location-tagger that is oblivious to the reasoning task, to tag locations in the dataset.We manually create a small set of 200 questions, randomly selected from the test-set, but ensuring that half of it contains *at least* one non-distractor location mentioned in the question while the other half contains questions with *only* distractor-locations.

As can be seen from Table 5.8, all models including the spatio-textual model deteriorate in performance if a question only contains distractors; the spatio-textual model however, suffers a less significant drop in performance.

Qualitative Study: We randomly selected 150 QA pairs with location-mentions from the test-set, to conduct a qualitative error analysis of Spatio-textual CRQA (Table 5.9). We find that nearly 37% of the errors can be traced to the textual-reasoner, 22% of the errors were due to a 'near' constraint not being satisfied, while about 13% of the errors were due to the model reasoning on distractor-locations. Lastly 8% of the errors were due to errors made by the location-tagger and incorrect geo-spatial data.

	Questions requiring Spatial-reasoning				
Models	**Hits@3**	**Hits@5**	**Hits@30**	**MRR**	$Dist_g$
SD	5.00	7.00	22.00	0.053	2.10
SPNET	1.00	1.00	8.00	0.013	2.64
CRQA	15.00	17.00	51.00	0.132	3.53
CRQA→SD	15.00	22.00	51.00	0.142	**1.96**
CRQA→SPNET	16.00	23.00	51.00	0.134	2.41
Spatio-textual CRQA	**22.00**	**28.00**	**54.00**	**0.182**	2.62
	Questions with distractor-locations only				
SD	2.00	3.00	17.00	0.025	4.12
SPNET	1.00	2.00	9.00	0.016	4.14
CRQA	19.00	26.00	51.00	0.162	3.62
CRQA→SD	13.00	17.00	51.01	0.108	3.26
CRQA→SPNET	13.00	17.00	51.00	0.113	**3.24**
Spatio-textual CRQA	**20.00**	**28.00**	**53.00**	**0.187**	3.50

Table 5.8: Experiments on two subsets from the test-set: (i) Questions requiring Spatial-reasoning (ii) Questions with distractor-locations only.

Error Type	**Percentage**
Textual Reasoning Error	37.9
Far from the required location	22.3
Influenced by Distractor	12.6
Not in requested Neighbourhood	10.7
Location Tagger Error	5.8
Repeated Location Names	4.9
Error in Geo-Spatial Data	2.9
Invalid Question	2.9

Table 5.9: Spatio-Textual CRQA: Classification of Errors

	Location Questions				
Models	**Hits@3**	**Hits@5**	**Hits@30**	**MRR**	$Dist_g$
CSRQA	19.89	26.43	**51.47**	0.168	2.70
Spatio-textual CSRQA	**21.36**	**28.36**	**51.47**	**0.183**	2.27
	All Questions				
CSRQA	21.45	28.21	**52.65**	0.186	2.47
Spatio-textual CSRQA	**22.41**	**28.99**	**52.65**	**0.193**	**2.32**

Table 5.10: Comparison with current state-of-the-art CSRQA on (i) Location Questions (ii) Full Task

	Automated evaluation		Human evaluation	
	Location	**Non-location**	**Location**	**Non-location**
CsrQA	28.00	36.00	64.00	70.00
Spatio-textual CSRQA	32.00	32.00	84.00	72.00

Table 5.11: Hits@3 results on a blind-human study using 100 randomly selected questions from the test-set

5.4.2.3 Spatio-Textual CsrQA

In the previous chapter we improved overall task performance by employing a neural IR method to reduce the search space [Mitra and Craswell, 2019], and then using the CrQA textual-reasoner to re-rank only the top 30 selected candidates (pipeline referred to as CsrQA). We therefore create a spatio-textual counterpart to CsrQA, by using spatio-textual reasoning in re-rank step. We find that this final model results in a 1 pt (Hits@3) improvement overall (see Table 5.10), and a 1.5 pt improvement on location questions (Hits@3).

Effect of False Negatives: To supplement the automatic evaluation, we additionally conducted a blind human-study using the top-ranked CsrQA and spatio-textual CsrQA models on another subset of 100 questions from the test-set. Two human evaluators (κ=0.81) were presented the top-3 answers from both models in random order and were asked to mark each answer for relevance. The manual annotation resulted in Hits@3 for CsrQA and spatio-textual CsrQA at a much higher, 67% and 78% respectively. As Table 5.11 shows, on the subset of location questions, the accuracy numbers are 64% and 84%. This underscores the value of joint spatio-textual reasoning for the task, and signifies a substantial improvement in the overall QA performance.

Spatio-Textual Retriever for CsrQA: The Spatio-Textual CsrQA model used the CsQA model to select candidates before re-ranking them with Spatio-Textual CrQA. However, we could also create a Spatio-Textual *retriever* by using CsQA as the textual reasoning network in our joint model 5.3. We refer to the Spatio-Textual CsrQA model that uses Spatio-textual CsQA for selection, and Spatio-Textual CrQA for re-ranking, as Spatio-Textual CsrQA+.

We begin by studying the performance of Spatio-Textual CsQA as compared to corresponding baseline models defined in Section 5.4.2.1. As can be seen in Table 5.12, Spatio-Textual CsQA performs better than baseline models including vanilla CsQA. This demonstrates that spatio-textual architecture is not dependent on a specific textual reasoning model can inject spatial-reasoning in other existing textual models. Comparing the performance of Spatio-Textual CsrQA+ with Spatio-Textual CsrQA, we find there

	Location Questions				
Models	**Hits@3**	**Hits@5**	**Hits@30**	**MRR**	D_g
CSQA	15.84	20.26	51.47	0.149	2.61
CSQA → SD	11.34	17.26	51.47	0.118	**2.18**
CSQA → LocNet	8.38	13.72	51.47	0.097	2.27
Spatio-Textual CSQA	16.11	21.27	54.51	0.159	2.52
Spatio-Textual CSRQA	21.36	28.36	51.47	0.183	2.27
Spatio-Textual CSRQA+	**21.46**	**29.09**	**54.51**	**0.187**	2.45
	All Questions				
CSRQA	21.45	28.21	52.65	0.186	2.47
Spatio-Textual CSRQA	22.41	28.99	52.65	0.193	**2.32**
Spatio-Textual CSRQA+	**22.69**	**29.73**	**55.22**	**0.198**	2.51

Table 5.12: Comparison of re-ranking models operating on a reduced search space returned by CSQA on Location Questions (ii) Comparison of spatio-textual CSRQA+ with CSRQA and spatio-textual CSRQA on the full task.

is nearly a 1pt improvement in Hits@5 and 3pt improvement in Hits@30.

5.5 Summary

In this chapter we presented the first joint spatio-textual QA model that combines spatial and textual reasoning. Experiments on an artificially constructed (spatial-only) QA dataset show that our spatial reasoner effectively learns to satisfy spatial constraints. We also presented detailed experiments on our POI recommendation task for tourism questions. When compared against textual-only and spatial-only QA models, the joint model obtains significant improvements. The design of our system is modular; we how the the use of our spatial-reasoner with two different textual reasoners helps improve their performance. Our final model spatio-textual CSRQA+ establishes a new state of the art on the task. Resources from our work are available at: https://ibm.biz/SpatioTextualQA

Part III

Comparison Questions

Chapter 6

Automated Entity Comparison

In the previous part of this thesis, we studied questions where users seek POI recommendations; answers to these questions were names of entities and they were returned by analyzing review documents describing those entities. Instead of seeking new recommendations, users may sometimes require a *comparison* of existing choices they have in mind. For example, in the question "*I will traveling to Spain and then to the US for two weeks in the summer. I really wanted to visit Granada in Spain and also NYC but due to some time and flight constraints I will only be able to pick one of the two. Could anyone help me choose?*", the user is not seeking an entity-answer but requires help in choosing between two destinations - Granada and New York City.

The proliferation of Web 2.0 has enabled ready access to large amounts of community created content, such as status messages, blogs, wikis, and reviews which form an important source of knowledge in our day to day decision making. Unfortunately, such content typically focuses on one real world entity at a time, whereas, a user deciding between alternatives is most interested in a *comparative* analysis of strengths and weaknesses of each. There have been some recent attempts to create comparisons using expert knowledge, but generating such comparisons manually does not scale – even pairwise comparisons are quadratic in the number of entities. Few automated comparisons for specific products with pre-defined attributes (e.g., laptops, cameras) exist; they are typically powered by existing structured knowledge bases. To the best of our knowledge, prior work on automatically generating comparisons for arbitrary domains from unstructured text, does not exist.

In this chapter, we define the novel task of generating *entity comparisons* from textual corpora, in which each document describes one entity at a time. For broad applicability, we do not restrict ourselves to a pre-defined ontology; instead, we use textual phrases that describe entities as our unit of information. We call these *descriptive phrases* –

Cluster Label	Granada	New York City
art, architecture	moorish-style architecture religious art fine art beautiful architecture ornamental architecture [....More]	contemporary art modern art medieval art egyptian art 19th century american art [....More]
courtyard,palace	brick-walled courtyard lovely courtyard area nasrid royal palace alhambra palace [....More]	
museum,finest	alhambra museum archaeological museum unesco world heritage site splendid arabic shops and restaurants isabella art collection and scepter [....More]	fine art museums guggenheim museum american museum metropolitan museum worlds finest islamic art collection [....More]
gardens,park	partal gardens palace gardens pleasant gardens moorish style gardens [....More]	famous flushing meadows park central park renowned gardens gateway national recreational area [....More]

Figure 6.1: Sample comparison for two cities - Granada (Spain) and New York City (United States) generated using our system. A quick look reveals that that both cities have a nice set of museums and gardens to visit, while palaces and courtyards are only in Granada. Granada's art and architecture are more ornamental, whereas New York's might be more contemporary.

they encompass general attribute-value phrases, opinion phrases, and other descriptions of the facets of an entity. We generate entity comparisons in a tabular form where the phrases are organized *topically*, thus, allowing for direct comparisons. Figure 6.1 shows a sample city comparison generated by our system for tourism. For the purpose of the work presented in this chapter, we do not identify the entities of interest from questions, and instead, make a simplifying assumption, that the entities for comparison are known; entities of interest may be tagged and identified from comparison questions by employing approaches similar to the ones used in Chapter 3.

Our comparison generation algorithm extracts descriptive phrases per entity and clusters them into semantic groups. We perform clustering via a topic model, where phrases from an entity are combined into one document. The topics identify prominent facets of the entities. Unfortunately, since the number of entities being compared is usually small, just statistical co-occurrence of words and phrases is not sufficient to identify good topics. In response, we use vector embeddings of descriptive phrases and employ a *Gaussian extension* of probabilistic Latent Semantic Analysis (pLSA)[1] over these vectors.

We also modify Gaussian pLSA to additionally incorporate an *entity-balance* term, preferring topics in which phrases from the entities are represented in a proportionate

[1]See Section 2.2.2 for an overview of pLSA.

measure. The balance term trades off the discovery of unique facets for each entity with that of common facets. This enables direct comparison between entities leading to an overall improved comparison table. Since the balance term is only a preference (not a constraint), it still allows the algorithm to exhibit clusters which may be sparsely represented (or not represented at all) in one of the entities.

We demonstrate the usefulness of our ideas on two domains – tourism (city comparison) and movies. Based on user experiments, we find that the entity-balanced model outputs much better comparisons as compared to an entity-oblivious model such as GMM.

6.1 Contributions

In summary, our work makes the following contributions:

- We define a novel task of generating entity comparisons from a corpus that describes entities individually.
- We present the first system to output such a comparison. Our system runs Gaussian pLSA over the vector embeddings of extracted phrases, and preferentially tries to balance the entities in each topic.
- Human subject evaluations using Amazon Mechanical Turk (AMT) demonstrate that AMT workers overwhelmingly prefer comparisons generated using entity-balanced Gaussian pLSA compared to entity-oblivious clustering.

6.2 Related Work

Recently, the internet has seen a growth in websites offering comparisons for different entities. Product websites such as eBay maintain comparisons for products. Google also outputs pre-built comparisons between common entities when queried with the word "vs." between them. Both of these output purely structured attribute-value information and are unable to compare along more qualitative and descriptive dimensions such as ease of living or quality of nightlife when comparing cities, for example. Other websites such as WikiVS[2] contain user-contributed comparisons that have been categorized based on the nature of the entities being compared. These are manually curated and therefore do not scale to the quadratic number of entity pairs.

Perhaps the most closely related work to ours is the field of contrastive opinion mining and summarization [Kim et al., 2011, Liu and Zhang, 2012]. Examples include extraction

[2] http://www.wikivs.com/wiki/Main_Page

of contrastive sentiments on a product [Lerman and McDonald, 2009] and summarization of opinionated political articles [Paul et al., 2010]. Contrastive opinion mining extracts contrasting view points about a *single* entity or event instead of comparing multiple ones. A recent preliminary study extends this for comparing reviews of two products [Sipos and Joachims, 2013]. It uses a supervised method for learning sentence alignments per product-type, and does not organize various opinions for an entity via clustering.

Other related work includes comparative text mining tasks where document collections are analyzed to extract shared topics or themes [Zhai et al., 2004]. Since such methods only identify latent topics for the full document collection, they can't be directly used for a specific comparison task.

Since our system is a combination of IE and clustering, we briefly describe related approaches for these subtasks.

Information Extraction: Our work is related to the vast literature in information extraction, in particular Open IE [Banko et al., 2007]. Our use of POS patterns for extracting domain-specific descriptive phrases is similar in spirit to ReVerb's patterns for relation extraction [Etzioni et al., 2011] and adjective-noun bigrams for fine grained attribute extraction [Huang et al., 2012, Yatani et al., 2011]. Adapting the literature on entity set expansion [Pantel et al., 2009, Voorhees, 1994, Natsev et al., 2007], our system expands seed nouns for broader coverage. We use Wordnet and distributional similarity-based approaches for this [Curran, 2003, Voorhees, 1994].

Clustering: Our entity-balanced clustering algorithm is related but different from previous work on *balanced* clustering. Prior work [Banerjee and Ghosh, 2006, Yuepeng et al., 2011, Lin et al., 2019] has focused on generating different clusters to be equi-sized. Other work [Zhu et al., 2010, Ganganath et al., 2014, Jitta and Klami, 2018] enforces size constraints on clusters. Our idea of balance, on the other hand, is targeted towards a better comparison and prefers that entities are well represented (balanced) in each cluster.

6.3 Task & System Description

Our motivation is to concisely compare two or more entities to aid a user's decision making. We make several choices in our task definition to help with this goal. First, we decide to output comparisons using a succinct tabular representation (see Figure 6.1). It has higher information density compared to, say, writing a natural language comparison summary.

Second, our unit of information is a *descriptive phrase*. We define it as any short phrase that describes an entity – these include attribute-value pairs (e.g., "Greek art"),

opinion phrases (e.g., "spectacular views"), as well as other descriptions (e.g., "oldest church of Europe").

Third, for better readability, our table must organize the information coherently along various aspects relevant for a comparison. We achieve this by grouping related descriptive phrases. The choice of aspects should be dependent on the specific entities being compared, e.g., the facet of "beaches" may split into "water activities" and "beach types" for Jamaica v.s. Hawaii, but not for San Francisco v.s. Bombay.

Moreover, comparisons are meant to highlight both the similarities and the differences between entities. We therefore need to trade-off the discovery of unique facets of an entity with those which are common to the entities being compared. Thus, while clusters that balance the entities are preferable, it is also acceptable to have clusters where one of the entities is sparsely represented (or not represented at all). This would happen in situations where that entity does not express a particular aspect and other entities do. Comparisons must trade off semantic coherence of facets with entity-balance in each facet.

Last, but not the least, since the comparisons are targeted to aiding user's decision making, understanding their intent is important. As an example, the user may be interested in city-comparison for the purpose of tourism, or for choosing a city to live in. Descriptive phrases for the former could be related to sightseeing, shopping, etc., but for the latter they may cover aspects such as living expenses, transportation, and pollution. We accommodate this necessity by allowing minimal human supervision for specifying user intent. This supervision can come in forms such as an intent-relevant seed noun list, or topic-level annotation following unsupervised topic modeling, etc. This supervision further guides descriptive phrase extraction.

6.4 Architecture

Our system consists of a pipeline of information extraction, clustering, cluster labeling and phrase ordering. IE extracts descriptive phrases relevant to user-intent and we develop a new clustering algorithm that produces better comparisons by balancing the entities in each cluster. We identify cluster labels based on the most frequent words in a cluster. We order phrases within a cluster based on the distance from the centroid. We now describe our IE and clustering techniques in detail.

6.4.1 Information Extraction

Our IE pipeline works in two steps. We first extract descriptive phrases via POS patterns and then filter out the non-topical phrases. For filtering, first we create a seed list of

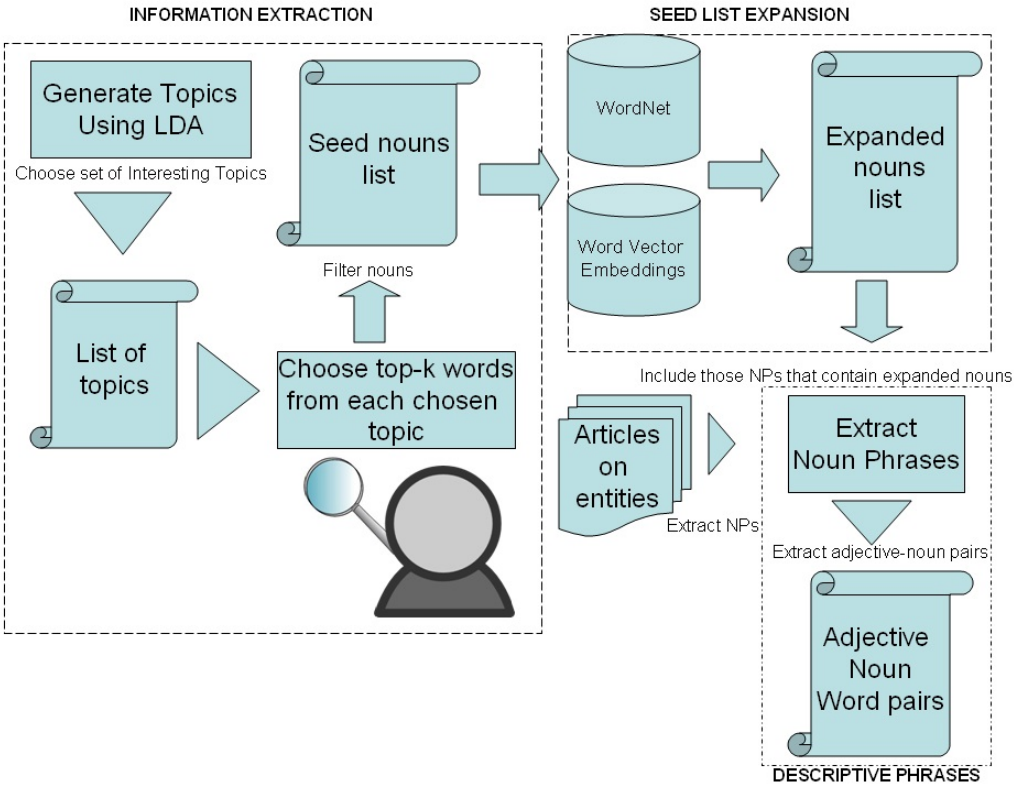


Figure 6.2: Information Extraction pipeline based on a seed list generated using LDA

relevant nouns via minimal human supervision, which are then expanded by item-set expansion. Descriptive phrases with a noun in the expanded list are retained, and rest are filtered.

Preliminary analysis on a devset revealed that a large fraction of descriptive phrases are noun phrases (NPs). We first extract all NP chunks from the collection and, additionally, using POS tags, extract any adjective-noun bigrams that are part of a bigger NP chunk, or missed due to chunking errors. This forms the initial set of descriptive phrases.

Filtering for User Intent: These descriptive phrases include those that are not relevant for user intent such as "excellent schools" for tourism. We filter these phrases by matching them to a list of intent-specific nouns. This list is created by first curating a seed list and then expanding it using item-set expansion. We employ two methods to obtain a seed list for specifying user intent: (1) a list of user-specified seed nouns, and (2) a labeling of LDA topics based on top words in each topic.

In the first approach we get the seed nouns directly from the domain expert. Our system supports the process by identifying frequent nouns and showing those to the annotator to annotate. For our tourism system, we spent about three hours to produce a list of 100 seed nouns.

Since this process requires significant effort per user intent, we also investigate a semi-

Method	Precision	Recall	F1
All nouns	0.53	**0.67**	**0.59**
Seed Nouns only (Manual)	**0.77**	0.32	0.44
Seed (Manual) + WN	0.71	0.35	0.46
Seed (Manual) + WV	0.70	0.40	0.49
Seed Nouns only (LDA)	0.74	0.19	0.30
Seed (LDA) + WN	0.74	0.20	0.31
Seed (LDA) + WV	0.74	0.26	0.38

Table 6.1: Quality of extracted descriptive phrases on a devset

automatic approach in which we run Latent Dirichlet Allocation (LDA) [Blei et al., 2003] on the whole phrase list. We then show the top 20 words in each topic and ask the annotator to provide only topic-level annotations. We treat the top 15 words from each positively labeled topic to be in the seed set. Since the number of topics is usually not that large, this significantly reduces the time required for annotation. E.g., we ran LDA with 20 topics and it took about 10 minutes to annotate them. However, the seed nouns are noisier due to noise in LDA. We summarize these steps in Figure 6.2.

Seed List Expansion: Finally, we use ideas from item-set expansion to expand the seed list for improved coverage. We implement two approaches for this step. In the first method (WN) we use Wordnet [Miller, 1995] to include words that are a direct hop away from the seed nouns. In the second approach (WV), we use word-vector embeddings [Collobert et al., 2011] and include top 10 neighbors of each seed in our expanded list. The expansions capture near-synonyms and topically related words.

IE Experiments: We now present comparisons of various IE methods on a small development set. We selected seven WikiTravel[3] articles (each article is on one city) and manually annotated an exhaustive set of descriptive phrases. This forms our devset for IE comparisons.

We chose various parameters in our IE systems so that our precision never drops below 0.70. For example, we used k=15 for choosing the top words from LDA into seed list. We use this target precision, because we believe that for any human-facing system the precision needs to be high for it to be considered acceptable by people.

Table 6.1 compares the performance of the various IE methods. Not surprisingly, we find that manual seed lists obtain a much higher recall as compared to LDA seeds, at approximately the same level of precision. Both Wordnet and word-vector improve the recall substantially, though vectors are more effective. The recall of all nouns is only 0.67 because a large number of descriptive phrases were larger n-grams (not just adjective-noun bigrams) and were missed due to chunking errors.

[3] www.wikitravel.com

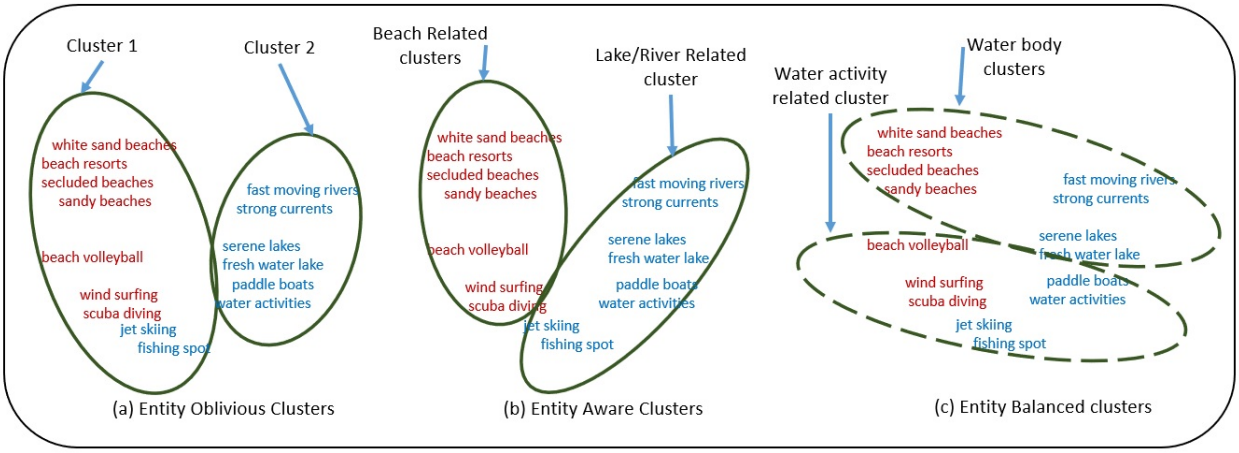


Figure 6.3: Three alternative clusterings (a), (b), (c) for descriptive phrases from two cities – each color is a different city. We prefer clusters shown in (c) as they balance information from both entities

6.4.2 Building Clusters for Comparison

Our next task is to construct meaningful comparisons using these phrases. A useful comparison of entities should organize the available information in a way that is easy to comprehend by the user. Towards this goal, we group the related descriptive phrases across a number of clusters. But simply having a good clustering of descriptive phrases may not be enough. We would like to have a clustering that explicitly captures the individual characteristics of each of the entities as well as makes the relative strengths and weaknesses of each entity apparent. For example, Figure 6.3 shows three different clusterings of phrases from two cities; phrases from each city are in a different color. Here, the third clustering is most appropriate for comparison, because not only is it a good clustering of descriptive phrases from each city considered separately, but the clusters produced also have *entity-balance*, i.e., the clusters produced have a good *balance* of both cities; both of these are key elements of comparison.

We first observe that a topic model such as Probabilistic Latent Semantic Analysis (pLSA) is a good fit to our clustering problem. In pLSA documents are characterized as mixtures of topics and topics as distributions over words. For our problem, we could combine all phrases for an entity into one document, and run pLSA to identify a coherent set of topics, which can then be used as clusters. Such a model will allow different entities to express topics in different proportions.

We note that LDA, which is a strict generalization of pLSA (LDA with uniform Dirichlet prior is equivalent to pLSA), is, in general, not a good fit for our task. LDA typically uses a *sparse* Dirichlet prior on document-topic distribution, which would not

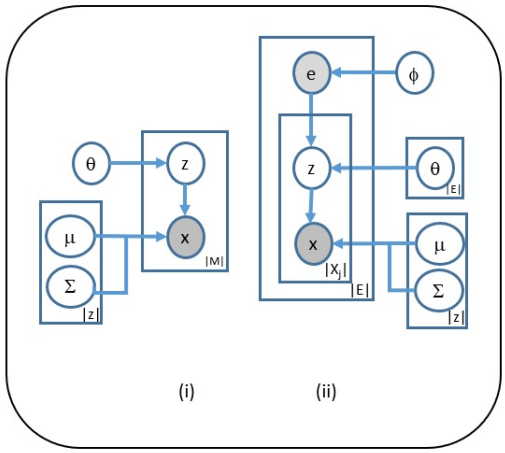


Figure 6.4: Plate Notation of (i) Standard Gaussian Mixture Model (ii) Gaussian pLSA (and entity balanced Gaussian pLSA)

be appropriate since for comparison we would like to represent each entity in as many topics as possible.

Unfortunately, a direct application of pLSA may not yield good results. This is because typically the number of entities being compared (i.e., the number of documents in pLSA) is very small (often 2), therefore, there isn't enough statistical regularity to find good coherent topics. The alternative proposition of learning topics on the whole corpus isn't very appealing either, since that will learn global topics and not the topics particularly meaningful for the current comparison at hand.

In response, we exploit the availability of pre-trained word vectors as a source of background semantic knowledge for every phrase, and generalize the pLSA model to *Gaussian pLSA* (G-pLSA). We construct a vector representation for each descriptive phrase by averaging the word-vectors of individual words in a phrase [Mikolov et al., 2013].[4] Thus, this model is pLSA with each topic-word distribution represented as a Gaussian distribution over descriptive phrases in the embedding space. This model is also similar to the Gaussian LDA model [Das et al., 2015], but without LDA's Dirichlet priors as discussed above.

Gaussian pLSA has several advantages for our task. First, it can meaningfully learn topics only for the entities being compared, instead of needing to learn a global topic model over the whole corpus. Second, due to additional context from word vectors, the topics are expected to be much more coherent compared to traditional topic models for cases when the underlying corpus is small, as in our case. Finally, in our model the vectors are generated from a Gaussian distribution and that helps capture the *theme* of

[4]We use the pre-trained 300 dimension vectors available at http://code.google.com/p/word2vec/

the cluster directly by enabling a centroid computation in the embedding space. This is especially useful for identifying and ranking important descriptive phrases per cluster while generating the comparison table.

Let $x_j^{(i)}, z_j^{(i)}$ denote the values of the i^{th} phrase and the corresponding cluster (topic) id, respectively, for the j^{th} entity e_j. Then, the log-likelihood $L(\Theta)$ of the observed data can be written as:

$$\sum_{j=1}^{|\mathbb{E}|}\sum_{i=1}^{|\mathbb{X}_j|} log\Big[\sum_{z_j^{(i)}=1}^{|Z|} P(x_j^{(i)}|z_j^{(i)};\Theta)\cdot P(z_j^{(i)}|\mathrm{e}_j;\Theta)\cdot P(\mathrm{e}_j;\Theta)\Big] \tag{6.1}$$

Here, $|\mathbb{X}_j|$ and $|Z|$ are the total number of phrases and clusters[5] respectively, for a given entity e_j and, $|\mathbb{E}|$ is the total number of entities being considered for comparison. Θ denotes the vector of all the parameters. We optimize the expression $L(\Theta)$ using EM and estimate the parameters of the model. As can be seen, the clusters are shared across entities, and the phrases generated are independent of the entity given, a cluster and the entities themselves are free to exhibit clusters in different proportions.

We also note just as pLSA can be seen as a natural extension of mixture of unigrams [Blei et al., 2003], Gaussian pLSA is an extension from the Gaussian Mixture Model (GMM) which is entity-oblivious. GMM generates each phrase independent of the entity it came from and hence, distributes entity phrases arbitrarily across clusters. We use GMM as a baseline for our experiments. Figure 6.4 illustrates the two models in plate notation.

Entity-Balanced Gaussian pLSA: Vanilla Gaussian pLSA may not always lead to a good clustering for comparison since the expression above does not involve any term to balance the entity-information in clusters, as motivated earlier. Thus, we incorporate a regularizer term to have a good balance (proportion) of entities in each cluster (see Figure 6.3 (c)) resulting in our final model for comparison called *Entity-Balanced Gaussian pLSA (EB G-pLSA)*. The plate notation for EB G-pLSA is identical to G-pLSA.

Our regularizer is a function of the KL-divergence between multinomial distributions for every pair of entities. KL-divergence $KL(P||Q)$ between two discrete distributions $P(x)$ and $Q(x)$ is defined as $\sum_l P(x_l) log\left(\frac{P(x_l)}{Q(x_l)}\right)$. Its an asymmetric measure of similarity and is equal to 0 when the two distributions are identical (and greater than 0 otherwise). Symmetric KL-divergence is defined as $Sym\text{-}KL(P,Q) = KL(P||Q) + KL(Q||P)$.

Let $P_{\theta_j}(z|\mathrm{e}_j)$ and $P_{\theta_k}(z|\mathrm{e}_k)$ denote the multinomial distributions for generating the cluster id z given the entities e_j and e_k, respectively. Here, θ_j and θ_k denote the respective multinomial parameters. We add a regularizer term to the log-likelihood minimizing the sum of symmetric KL-divergence between the distributions $P_{\theta_j}(z|\mathrm{e}_j)$ and $P_{\theta_k}(z|\mathrm{e}_k)$ for

[5]Note that number of clusters for all entities will be the same i.e, $|Z_j| = Z$ for all j

every pair of entities e_j and e_k. Adding this regularizer requires the multinomial distributions to be similar to each other, thereby preferring balanced clusters over unbalanced ones. Our regularized average log-likelihood can be written as:

$$L_{reg}^{avg}(\Theta) = \frac{1}{M}L(\Theta) - \eta \cdot \left[\sum_{j,k=1|j<k}^{|\mathbb{E}|} Sym\text{-}KL(P_{\theta_j}, P_{\theta_k}) \right] \tag{6.2}$$

$L(\Theta)$ is the total log-likelihood as defined in the previous equation. $M = \sum_{j=1}^{|\mathbb{E}|} |\mathbb{X}_j|$ and $|\mathbb{E}|$ is the total number of entities being compared. η is a constant controlling the weight of the regularizer. Note that we add the regularizer term to the *average* log-likelihood (instead of the total log-likelihood) in order to have the same regularizer value for comparisons having varying number of data points (descriptive phrases). This is important to obtain a single value of η which would work well across different entity comparisons. In our experiments, η was tuned using held-out data and was found to be robust to small perturbations.

We use standard EM to optimize the regularized log-likelihood. Since the regularizer does not have any hidden variables, E-step is identical to the one for the unregularized case. During M-step, the values maximizing the mean parameters μ_z and the ϕ parameter can be obtained analytically. There is no closed form solution for the parameters θ_j, θ_k. We perform gradient descent to optimize these parameters during the M-step. In our experiments, we did not estimate the co-variance matrices Σ_z and kept them fixed as a diagonal matrix with the diagonal entry (variance) being 0.1. We did not learn the co-variance matrices as that would have increased the number of parameters substantially, and thus, had the danger of over fitting. The small value of the variance chosen was to ensure less overlap between different clusters.

Clustering Experiments: We conducted preliminary experiments to compare the performance of GMM (vanilla Gaussian mixture modeling using word vectors) with G-pLSA and EB G-pLSA on a development set consisting of 5 random city pairs. The descriptive phrases were constructed using the automated seed list as described in IE Section. We manually created the gold standard clusterings. The number of clusters was set to the number in the gold set for each of the city pairs.

We used f-measure and pairwise accuracy to evaluate the deviation from the gold standard for the clusterings produced by each of the algorithms. Table 6.2 shows the results. EB G-pLSA performs better than the other two algorithms on both the metrics, and especially on pairwise accuracy. Performance of G-pLSA is very similar to GMM.

	GMM	G-pLSA	EB G-pLSA
f-measure	0.42	0.43	0.44
pairwise accuracy	0.66	0.65	0.76

Table 6.2: Comparing clustering methods on development set

6.5 Evaluation

In order to evaluate the usefulness of our system we conducted extensive experiments on Amazon Mechanical Turk (AMT). Our experiments answer the following questions. (1) Are comparisons generated using our clustering methods G-pLSA and EB G-pLSA preferred by users against the entity oblivious baseline of GMM? (2) Are our system-generated comparison tables helpful to people for the task of entity comparison?

Datasets & System Settings: We experiment on two datasets – tourism and movies. For tourism, we downloaded a collection of 16,785 travel articles from WikiTravel. The website contains articles that have been collaboratively written by Web users. Each article describes a city or a larger geographic area that is of interest to tourists. In addition, all articles contain sections[6] describing different aspects of a city from a tourism point of view (e.g., places to see, transportation, shopping and eating). For our proof of concept, we performed IE only on the 'places to see' sections.

For Movies dataset, we used the Amazon review data set [Leskovec and Krevl, 2014]. It has over 7.9 million reviews for 250,000 movies. We combined all the reviews for a movie, thus, generating a large review document per movie. This dataset is much noisier compared to WikiTravel due to presence of slang, incorrect grammar, sarcasm, etc. In addition, users also tend to compare and contrast while reviewing movies so there are even references to other movies. As a result, the descriptive phrases extracted were much more noisy.

For the time consuming manual seed list setting of our IE system, we only use the

Domain	Total pairs	EB G-pLSA Win		GMM Win		EB G-pLSA Win		G-pLSA Win		G-pLSA Win		GMM Win	
		4-0	3-1	1-3	0-4	4-0	3-1	1-3	0-4	4-0	3-1	1-3	0-4
Tourism	30	20%	33%	13%	0%	13%	30%	30%	0%	17%	27%	13%	3%
Movies	20	20%	35%	10%	0%	5%	35%	15%	5%	5%	45%	15%	5%

Table 6.3: User preference win-loss statistics for different clustering methods on both city and movie comparison task using the same IE system. Both EB G-pLSA and G-pLSA significantly outperform the baseline GMM model. EB G-pLSA has some edge over the G-pLSA model. Note: Ties have not been shown in the table.

[6] http://wikitravel.org/en/Wikitravel:Article_templates/Sections

tourism dataset. For movies, we generate seeds using annotation over LDA topics only. For all systems we use word-vectors to expand the seed list.

For each table, we generated k clusters where k was determined using a heuristic,[7] [Mardia et al., 1980] and we displayed at most 30 phrases per cluster. We did not display any cluster that had less than 4 phrases.

6.5.1 Evaluation of Clustering Algorithms

In order to examine whether clustering using EB G-pLSA indeed produces best comparison tables, we conducted a human evaluation task on Amazon Mechanical Turk (AMT) where users of our system were asked to indicate their preference between two comparison tables. Since we have three systems we performed this pairwise study thrice. In each study, two comparison tables were generated from different systems. For each entity-pair we asked four workers each to select which comparison table they preferred. The order of the tables was randomized to remove any biasing effect. We paid \$0.3 for each table comparison. Table 6.3 reports the results for both domains where descriptive phrases were generated using LDA+WV. The list of pairs used in our experiments is available in Appendix C.

On 30 city-pairs in the Tourism domain, workers preferred the comparison tables generated using EB G-pLSA 53% of the time and GMM was preferred only 13% (the rest were ties). It is worthwhile to note that whereas in 20% of the comparisons, EB G-pLSA had a clear 4-0 margin, there was no such comparison where all the workers preferred the GMM model. We also requested users to provide the reasons for their preferences. While most users specified a non-informative reason such as "like it better", some users gave specific reasons such as "subdivides the parts I find useful into more specific categories" and "easy to understand and more specific points of comparison". Our results also show that G-pLSA is a distinct improvement over GMM (44% vs. 16%). EB G-pLSA had a marginal edge over G-pLSA (43% vs. 30%).

On movies domain, we report results on 20 movie-pairs and we again found an overwhelming preference for the system using EB G-pLSA for clustering. 55% of the time, the output of EB G-pLSA was preferred over GMM's 10%. Other comparisons between G-pLSA and GMM, and between our G-pLSA and EB G-pLSA systems also follow trends similar to tourism domain. Figure 6.5 shows an example of a comparison generated for two movies - Batman and Gandhi, using EB G-pLSA.

The performance of EB G-pLSA is statistically significantly better than GMM for both the tourism and the movie datasets, with p values being less than 0.00004 and

[7] No. of clusters = square root of half the number of phrases

Cluster Label	Batman (1989)	Gandhi (1982)
movies,films	comic-book movies batman films superhero films mega-blockbuster movie 1989 batman film t [....More]	biographical films poignant movies legendary films best picture movie most important historically-based biographical filmt [....More]
india,africa		1947 india africa and india india / pakistan riots south africa gandhi [...More]
comic,book	comic book charecter comic book movie plot comic book movies [....More]	modern indian history little indian history movie gandhi references [...More]
performance,job	academy-award-nominated performance excellent direction job good deal performance [....More]	awardwinning performance great job brilliant performance [....More]

Figure 6.5: Sample comparison for two movies - Batman (1989) and Gandhi (1982), generated using our system.

0.002, respectively, using a one-sided students t-test. This strong preference suggests that the clustering induced by incorporating entity balance in the clusters produces much better comparison tables.

6.5.2 Value of Comparison Tables

The goal of our experiments in this section was to assess whether our comparison tables add value to some realistic task and to understand the overall usefulness of our system. To our knowledge there are no other automated systems comparing cities for tourism (or movies), hence we could not evaluate our system against existing approaches. Therefore, we decided to evaluate the benefit of the output generated by our system (i.e., comparison tables) against reading the original WikiTravel articles. For fairness we only use the 'places to see' sections from WikiTravel, since that was the raw text used in generating comparison tables in the first place.

Since the comparisons are generated automatically, people may not find them understandable, or there may be missing valuable information. We test this in a human subject evaluation. We adapt the evaluation methodology developed recently for contrasting multiple ways of presenting information and testing the overall learning of the subjects [Shahaf et al., 2012, Christensen et al., 2014]. The evaluation is divided into two parts. In the first part the workers are given a limited time to read the information provided (articles or comparison tables) for an entity-pair. They are then asked to write a short 150-300 word summary contrasting different aspects of the two entities. Each user writes two summaries, one based on articles and the other based on our table. Our study pairs two users such that if user1 read the articles for city pair 1 and the table for city pair 2, their partner user will see the reverse. The workers were additionally asked

which knowledge source they preferred and why.

Making a worker create summaries using both information sources helps reduce the effect of worker comprehension and skill in the evaluation of our task, as each worker contributes to summaries created using our system as well as the baseline. In order to reduce the effects of any sequence bias, half the mechanical turk workers were first shown the output of our system followed by the articles and the other half (partners) were shown content the other way around.

In the second part of this experiment we directly compare the knowledge acquisition of these workers. In particular, we ask a different set of workers to evaluate the summaries created by the partnered workers. In each task, a worker has to compare two summaries for the same entity-pair, one created using tables by one worker and other created using articles by their partner. Each summary pair was shown to four different users and each of them was asked to select the summary they preferred for comparing and contrasting the entities. Since we perform this experiment on Tourism data, the MTurk task descriptions explained that the intent of the comparison is tourism and their summaries or preferences must be from that perspective.

6.5.2.1 Results

We performed this evaluation on twenty city pairs using both our information extraction methods i.e. Manual+WV expansion (referred as TABLE-M) and LDA+WV expansion (referred as TABLE-LDA) along with the EB G-pLSA method for clustering. The city pairs were chosen such that the cities are related but not too similar, and the workers would likely not have thought of the specific comparisons before.

We found that in the first part where workers were given 10 minutes to create the summaries, they on average asked for 30% more time to create the summaries when information was presented as article. This supports our belief that our system-generated tables successfully reduce information overload. It also suggests that the structure added by the system (clusters) was useful for the comparison task and reduced workers' cognitive load.

We now present the results for the second part of the study in which workers evaluated the comparison summaries written by the workers in the first part. Within 20 city-pairs, summaries for 5 city pairs (25%) generated based on TABLE-M were preferred and 5 (25%) generated based on original articles were chosen. The workers were indifferent in 10 of the city pairs (both summaries got two votes each). This shows that despite having a very high compression ratio, workers still managed to create summaries that were comparable in quality to those created by reading original documents. We repeated

the same study using TABLE-LDA and found that summaries for 8 city pairs (40%) generated based on TABLE-LDA were preferred and 5 (25%) generated based on original articles were chosen. The workers were indifferent in 6 of the city pairs (both summaries got two votes each).

We did not repeat this experiment using the Movies data set as the source articles were concatenated reviews with no structure and it would not be surprising that users prefer our system. In summary, we find that both our systems convey adequate and useful information in the comparisons and the summaries generated by users using our systems were found to be as good as the ones created by users reading the full articles.

6.6 Summary

In this chapter, we defined the novel task of automatically generating tabular entity comparisons from unstructured text. We also implemented the first system for this task that first extracted descriptive phrases from text, and then clustered them to generate comparison tables. Our clustering algorithm is a Gaussian extension of p-LSA, where the descriptive phrases are represented using embeddings in the word vector space. In order to have a better comparison between entities, we incorporated a balance term which prefers clusters where entities are proportionately represented.

We performed extensive human-subject evaluations for our systems over Amazon Mechanical Turk (AMT) on two datasets – tourism and movies. We found that AMT workers overwhelmingly preferred EB G-pLSA based comparisons over GMM-based. We also assessed the value of our generated comparisons over reading the original articles. We found that while both sets of workers learned as much, the workers viewing tables asked for less additional time to narrate a comparison in words. Overall, we believe that comparison tables add value for users deciding between multiple entities.

Part IV

Epilogue

Chapter 7

Conclusion & Future Work

Recent progress has helped improve the state-of-the-art in QA, leading to its use in a variety of applications – from helping school students practice math word problems, to systems that answer questions based on facts presented in text. In the spirit of promoting further research on challenges associated with real-world QA, in this thesis, we contributed a series of new QA problems using data from the tourism domain. We focused our attention on two types of questions –*recommendation* questions and *comparison* questions.

We first introduced the task of understanding *recommendation* questions that often express vague and subjective constraints based on location, budget, etc. We formulated the problem as a semantic labeling task over an open representation that made minimal assumptions about schema or ontology-specific semantic vocabulary. At the core of our model, we used a BiLSTM (bi-directional LSTM) CRF, and to overcome the challenges of operating with low training data, we supplemented it by using BERT embeddings, hand-designed features, as well as hard and soft constraints spanning multiple sentences. This helped develop a pipelined QA model, that first parsed a question and then retrieved an answer entity from a downstream knowledge store (Google Places in our experiments). We found that the use of our labels helped answer 36% more questions with 35% more (relative) Hits@3 scores as compared to keyword-querying based baselines. While this model had the advantage of requiring very little training data and it relied on the use of a black-box knowledge source which gave us no control over the reasoning process employed for answering.

Therefore, we then studied the QA problem by answering questions directly using a collection of reviews and a set of labeled QA pairs. We harvested a novel real-world QA dataset containing 47,124 paragraph-sized real user questions from travelers seeking Points-of-Interest (POI) recommendations for hotels, attractions and restaurants. We also associated each entity answer with a collection of unstructured reviews sourced from

travel websites. Since questions could have thousands of candidate answers to choose from, the task presented novel challenges of reasoning at scale. We found that existing architectures were infeasible to run on this dataset and we thus developed a scalable three-stage *cluster-select-rerank* pipeline which worked better than a neural information retrieval model or a pure attention-based re-ranker. Our model now serves as strong baseline for this task.

POI-seeking recommendation questions also express various kinds of spatial and non-spatial constraints. We developed the first joint spatio-textual reasoning model, which combines geo-spatial knowledge with information in textual corpora to answer questions. We created a modular spatial-reasoning network that used geo-coordinates of location names mentioned in a question, and of candidate answer POIs, to reason over only spatial constraints. We then combined our spatial-reasoner with the text-based reasoner from the three-stage pipeline, as a joint model. We demonstrated that our joint spatio-textual model performed significantly better than models employing only spatial- or textual-reasoning.

Finally, we defined the novel task of automatically generating entity comparisons from text. We presented comparisons in the form of a table that semantically clustered descriptive phrases about entities. We developed a novel clustering algorithm that balances information about entities in each cluster, to generate comparison tables. We tested our system's effectiveness on two domains, travel articles describing cities and movie reviews, and found that entity-balanced clusters were overwhelmingly preferred by users.

We now discuss some open problems and additional directions for future work.

7.1 Improving Joint-Reasoning

7.1.1 Question Answering

In the new POI-recommendation dataset (Chapter 4) that we created, we found that nearly 23% of the questions express budgetary constraints while nearly 21% of the questions contained temporal constraints (Section 4.2.4). Similar to our work in Chapter 5, where we incorporated joint reasoning using geo-spatial and textual data, methods that reason over other types of constraints could be developed. For instance, to resolve budgetary constraints, models may need to utilize pricing information (numeric data or ranges) along with information in reviews to answer questions. Similarly, temporal constraints would require incorporating knowledge about open and closing times along with the ability to understand the elements in a calendar. These would then need to be combined with textual information for joint reasoning. Improvements to spatio-textual

reasoning can also be made – for instance, our model cannot support reasoning on questions that require directional or topographical inference (eg.“north of X”, “on the river beach”). These types of constraints require resolving toponyms and reasoning over topographical classes. Alternative approaches for joint-reasoning could also be developed - for instance, one could extend ideas from Graph-neural network based approaches, such as NumNet [Ran et al., 2019] where each entity could be viewed as a node in a graph for reasoning. However, we note that methods will need to be made more scalable for them to be useful. The entity space (and thus nodes in the graph) would run into thousands of nodes per question making current message-passing based inference methods prohibitively expensive.

Finally, one could also envision a unified joint model that supports reasoning on spatial, budgetary, temporal constraints along with textual information. Our work provides a dataset, a strong baseline method, a joint model that incorporates one such constraint (location) and we hope that it will serve as a strong foundation for further research in this area.

7.1.2 Clustering

In our work in Chapter 6, we used a list of seed nouns to capture a user's intent for comparison. These nouns were used to extract descriptive phrases from articles describing entities, and clustering was then applied on these phrases to generate comparisons. In future work, it could be interesting to explore systems that jointly extract descriptive phrases and then generate clusters guided by the “intent” or “goal” of the comparison task. Along with using descriptive phrases, it could be useful to explore methods for joint clustering that incorporate knowledge from structured data sources that contain complementary information.

7.2 Improving Textual Reasoning

Many existing QA tasks that rely on unstructured knowledge are largely formulated as variants of reading comprehension tasks, which assume that answers are stated explicitly in the documents [Rajpurkar et al., 2018, Joshi et al., 2017] or require inference over text from one or more passages/articles to generate the exact correct answer [Yang et al., 2018, Reddy et al., 2018, Choi et al., 2018, Fan et al., 2019]. In contrast, our POI-recommendation questions differ from traditional factoid or semi-factoid questions, since POI-recommendation questions may be vague or have under-specified requirements, resulting in subjective answers. Users may also express preferences and constraints in

questions, which requires deeper reasoning and the use of external knowledge sources for answering. The task is further complicated by the nature of the knowledge source which consists of subjective opinions often contradictory and could contain sarcasm, references to other entities (for example as a comparative mention). While the CsrQA method we developed in Chapter 4 overcomes some of the novel challenges of reasoning at scale posed by our task, there is a lot of room for improvement – a significant number of errors (61%) were due to the model not fulfilling user preferences of cuisine, age appropriate and/or celebration activities, hotel preferences etc. Our current approaches do not explicitly reason over the *intensity* of constraints (eg: " *love* chinese food" vs "*dont mind*" a chinese restaurant. The use of sentiment analysis and emotion recognition [Liu et al., 2010, Xu et al., 2019] along with explicit reasoning on qualitative constraints could also be helpful. For instance, methods based on Logical Neural Networks [Riegel et al., 2020], which internally use weights to reason over derived logical representations, could use sentiment-aware representations of constraints for reasoning. Applying multi-objective optimization methods [Deb and Deb, 2014] over such derived but explicit constraint representations could also be an interesting direction of future work. Exploring methods that incorporate query annotations (such as those in Chapter 3) along with deep reasoning over documents (such as those in Chapters 4 and 5) could help in our task. Lastly, developing scalable architectures that are pre-trained or use transfer learning, could also be helpful in our task.

An interesting extension to our baseline CsrQA model could be to make the clustering step question-dependent – this could help representative entity documents better reflect important information for each question. In addition, it could be interesting to explore ideas similar to the recent use of dense phrase vectors [Seo et al., 2019, Lee et al., 2020] to overcome some challenges of scale. Finally, developing QA methods that are capable of reasoning over subjective phrases, to account for contradictions and sarcasm, along with multi-hop reasoning across review documents, could be helpful in making significant progress on this task.

For the task of entity comparison presented in Chapter 6, we chose to view comparisons in the form of tables consisting of descriptive phrases. With the development of large pre-trained transformer models, it could also be interesting to explore the generation of *comparative summaries* where documents describing entities are summarized to highlight their similarities and differences. This could be viewed as another flavour of existing multi-document summarization tasks [Goldstein et al., 2000, Liu and Lapata, 2019], where information from more than one document is jointly summarized.

7.3 Task Extensions

As described in Chapter 1, apart from recommendation and comparison questions travel forums contain questions that seek time-sensitive information (e.g bus schedule details), or require validation/recommendations of itineraries that not only account for transportation and commutes, but also places to visit, stay along with the duration of each visit etc. These are challenging problems because, not only would models need to determine the POI-recommendations, they also need to organize them to fulfil higher order constraints of travel schedules, duration of visits, prices that may vary across dates, routing etc. Developing such models would be a leap forward in the capabilities of current QA systems that support tourism questions. Further, such models could also be extended to support conversational queries, similar to existing work on task oriented dialogs [El Asri et al., 2017, Eric et al., 2020], where systems communicate in natural language with users to make flight, hotel, restaurant reservations etc. Models would not only need to address neuro-symbolic reasoning challenges discussed previously, they would also need to be able to *fuse* and incorporate information from multiple knowledge sources. For instance, travel reservation systems could return ticketing and pricing information, reviews could help make recommendations based on user preferences, structured attributes and FAQs could help provide details about timings, facilities, restrictions, etc. Additionally, with time-sensitive application domains such as tourism, temporal reasoning also becomes important – certain venues may be open on some days of the week, areas may be closed due to natural disasters, sporting events, law and order situations and as 2020 has shown, even a pandemic!

During the COVID-19 pandemic we found that many restaurants and hotels that we had used as part of our POI-recommendation dataset had suspended operations temporarily or had changed the nature of the services – for example, fine dining restaurants served only takeout food. In such circumstances, information in blogs and reviews may become out of date very quickly. How should QA systems adapt to situations when an important knowledge source suddenly becomes irrelevant due to external factors? Perhaps it could be useful to consider the development of reasoning models that not only aggregate information across knowledge sources, but are also aware about the *nature* of knowledge within them. Such models could then prioritize one knowledge source over the other, either by user input or a trigger (for example, based on monitoring and classification of tweets or news stories to detect that a knowledge source may be stale or needs to be plugged out of the system).

7.4 Extension to other domains

We believe that our work on answering recommendation questions has parallels in other domains – for example, questions seeking consumer product recommendations such as "*a laptop with a responsive keyboard and* 14*-inch screen*" will require a system to reason over screen size using a database of product properties along with textual reasoning about, whether or not, the keyboard is responsive (likely found in user reviews). Similarly, users often post questions seeking recommendations online when making purchase decisions about big-ticket items such as houses and cars. For example, this is an actual forum question[1] *"I'm looking to buy a car for the first time, my boyfriend has been driving me to work everyday so he'll be happy once I get a car. Even though I'm inexperienced I'm not young I'm 32yrs old for many years my priority was to buy our first home together and now I'm in a financially stable sitisituan were I can now afford a car of my own , but I will be looking at used cars as I don't want to commit to finance deals I have saved over the years what I hope is a good budget of 10k. I am hoping you can all give recommendations of which models or brands I should be looking at. I don't have any kids or pets just me and my boyfriend of 12yrs, and we won't be having any kids as due to medical reasons I can't have kids. So it's just me and my boyfriend. I am thinking of something small and easy to run as well as cheap to run but there's two snags why I'm struggling and am hoping I can get help on here as my boyfriend is useless as he's driven the same car for the past 10yrs he drives a lexus RX (he has a better salary than me , he's a doctor, my salary is peanuts compared to his you don't earn much as a receptionist) so as much as I Love his car I can afford that and he doesn't know much about cars apart from knowing a fair bit about Lexus's. The first issue is when I was learning I tried to do my lessons in a manual car but due to my fibromyalgia I struggled with my pain to keep up with changing gears and switching between the pedals and it was my instructor who advised I learn automatic which was much eaiser but also disappointed as I know it will limit my car choice. So any recommendations have to be a good automatic. The second is my fibromyalgia the car has to be comfty on long journeys as I can get really stiff were in my boyfriend's RX the ride is smooth and is really comfty and doesn't affect my fibromyalgia but a friend's mini cooper caused a lot of pain. I do about 250 miles a week to work (well until I get my car my boyfriend does 240 miles to get me to work and back every week) so it's really important it's a comfortable make/model and one I can buy with my budget. Looking forward to hear from you all".* Similar to the challenges faced in answering tourism recommendation questions, the question here is long, with irrelevant sentences and references (eg: financial priorties). Further, answering such questions would require a repository of vehicles along

[1] https://www.bogleheads.org/forum/viewtopic.php?t=268597

with details about them over which reasoning needs to be performed. In Chapter 3, we demonstrated how our open semantic representation could apply to the task of book recommendations. Answering such questions would again require a query-able knowledge source; if a parsing based approach were to be taken for QA.

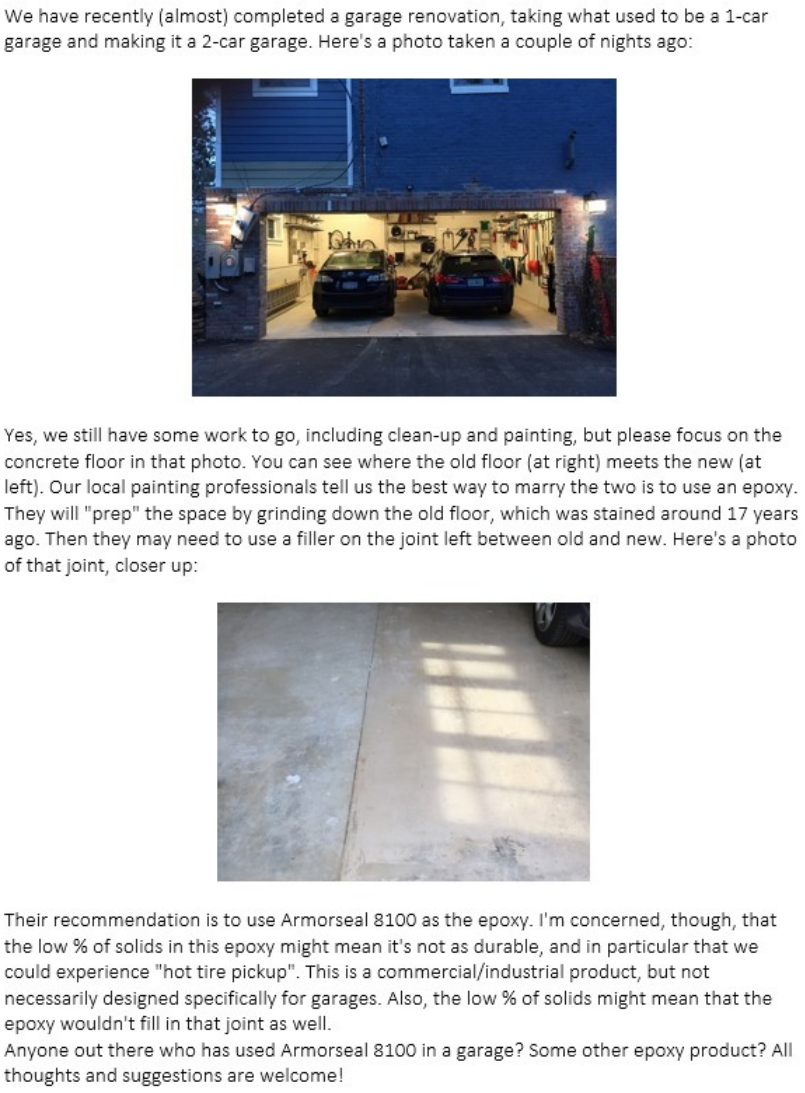


Figure 7.1: Example of a multi-modal recommendation question. Answering this question, requires understanding the information encoded in the images. Question and image source: https://www.houzz.com/discussions/5643190/best-garage-floor-epoxy#n=15

In our work so far, we have assumed that the questions express information in one modality – text. However, in some cases, real-world recommendation questions can also require multi-modal reasoning for both question understanding and answering. For example, Figure 7.1 shows a question asking for recommendations of an epoxy to be used in a garage. As can be seen, the question also requires the information encoded by the

image to be used for answering. This is a challenging problem – not only does answering this question require multi-modal reasoning, it also perhaps requires an understanding of deeper domain specific concepts in chemistry which are unlikely to be expressed in free text. Recent work on combining text with images, including those that learn shared representations of images and text [Tsai et al., 2019, Hu and Singh, 2021], as well as models, that operate on chemical representations [Honda et al., 2019] may be helpful for such questions.

This thesis explored a novel set of real-world challenges using data from tourism domain. We conclude this work be noting there are numerous possible directions for future work – solving more domain specific challenges (e.g, making travel plan recommendations), fusing more knowledge types (e.g, joint reasoning over graph topology and text, image and text), addressing similar challenges in other domains (e.g, product recommendation, car recommendation), supporting conversational QA for recommendations (e.g, for automated travel agents), developing robust information-fusing reasoning methods that are aware about the purpose of each knowledge source, developing methods for comparative multi-document summarization, etc.

Appendix A

POI-Recommendation Dataset Statistics

City ID	#Attractions	#Restaurants	#Hotels	Total Entities	Avg #Reviews	Avg #Tokens	Avg #Tokens per Review
New York	846	8336	562	9744	83.16	4570.5	54.9
Washington	351	2213	220	2784	100.7	5403.2	53.6
Chicago	471	5287	174	5932	51.44	2833.4	55.1
San Francisco	426	3661	302	4389	60.36	3170.6	52.5
Mexico City	290	2607	318	3215	26.6	1173.5	44.1
Miami	168	2283	191	2642	52.5	2416.5	46.0
Vancouver	243	2518	118	2879	63.17	3183.9	50.4
Sao Paulo	248	3336	232	3816	9.1	370.0	40.7
Buenos Aires	324	2385	334	3043	27.17	1283.4	47.2
Rio De Janeiro	290	2320	205	2815	24.54	1118.14	45.5
London	1466	16212	710	18388	130.46	7243.7	55.5
Dublin	387	1938	270	2595	160.6	8667.8	53.9
Paris	767	11379	711	12857	58.6	3172.4	54.1
Rome	850	6393	402	7645	77.546	4115.2	53.
Stockholm	200	2168	125	2493	56.39	2646.8	46.9
Oslo	211	1061	74	1346	73.06	3362.0	46.0
Zurich	144	1434	97	1675	47.81	2162.0	45.2
Vienna	412	2761	332	3505	82.33	3724.7	45.2
Berlin	518	5147	593	6258	64.99	2958.4	45.5
Budapest	340	2225	170	2735	123.26	5762.8	46.7
Bucharest	212	1424	196	1832	47.76	2043.1	42.7
Moscow	544	3291	259	4094	21.73	946.1	43.5
Amsterdam	358	3055	422	3835	116.8	5769.9	49.4
Beijing	509	2234	0	2743	18.87	1067.4	56.6
New Delhi	350	5102	671	6123	31.72	1317.7	41.5
Mumbai	432	7159	383	7974	22.45	881.6	39.3
Agra	66	250	203	519	93.27	3979.7	42.6
Bangkok	435	5778	793	7006	54.48	2457.0	45.1
Karachi	62	219	14	295	22.31	869.4	38.9
Singapore	28	7616	453	8097	42.52	1965.2	46.2
Jakarta	194	3853	602	4649	25.64	887.3	34.6
Tokyo	0	0	781	781	157.04	6215.8	39.6
Seoul	354	3747	611	4712	25.53	1098.8	43.0
Bukhara	38	23	14	75	24.4	1112.3	45.5
Ulaanbaatar	56	222	47	325	23.75	1103.1	46.4
Kathmandu	111	588	178	877	54.07	2356.9	43.6
Melbourne	324	3030	162	3516	49.31	2464.2	49.9
Sydney	353	4100	362	4815	64.57	3125.5	48.4
Auckland	175	1733	238	2146	48.5	2330.7	48.0
Havana	183	657	23	863	44.07	2394.8	54.3
Honolulu	218	1561	117	1896	88.2	4762.8	54.0
Kingston	39	159	54	252	68.6	2721.9	39.7
Seychelles	0	1	0	1	20.0	1306.0	65.3
Dubai	247	5786	347	6380	54.14	2419.6	44.7
Cairo	155	1232	111	1498	33.27	1344.9	40.4
Amman	41	499	143	683	58.52	2264.7	38.7
Jerusalem	227	561	31	819	55.93	2651.8	47.4
Johannesburg	111	929	180	1220	58.559	2262.5	38.6
Cape Town	139	822	287	1248	121.638	4969.0	40.8
Nairobi	72	477	107	656	42.2698	2013.7	47.6

Table A.1: City Wise - Knowledge Source Statistics

City Name	#Questions	#QA Pairs	#QA Pairs With Hotel	#QA Pairs With Restaurants	#QA Pairs With Attractions	Avg #Tokens in Question
New York	5891	14673	1030	12841	802	77.1
Washington	861	1886	168	1591	127	73.2
Chicago	1189	2888	129	2583	176	76.2
San Francisco	1621	4079	410	3417	252	74.0
Mexico City	127	216	65	137	14	68.4
Miami	98	134	28	97	9	68.2
Vancouver	498	874	223	554	97	74.9
Sao Paulo	16	25	7	16	2	75.7
Buenos Aires	268	493	140	325	28	77.2
London	3387	8265	569	6572	1124	75.9
Dublin	621	1103	196	810	97	72.5
Rome	1004	1782	234	1292	256	72.4
Stockholm	160	280	56	190	34	78.1
Oslo	67	114	43	65	6	78.2
Zurich	95	147	41	97	9	69.8
Vienna	292	465	89	320	56	66.0
Berlin	386	652	68	453	131	71.8
Budapest	317	655	23	605	27	75.3
Bucharest	22	46	3	41	2	59.8
Moscow	64	106	26	74	6	70.2
Amsterdam	669	1299	207	1002	90	70.6
Beijing	54	71	0	57	14	70.7
New Delhi	28	55	24	18	13	54.0
Mumbai	166	334	98	198	38	63.8
Agra	40	52	36	14	2	54.8
Bangkok	743	963	313	482	168	65.1
Singapore	515	821	332	471	18	67.6
Jakarta	25	44	15	15	14	64.7
Tokyo	16	22	22	0	0	64.6
Seoul	70	82	39	29	14	69.4
Kathmandu	23	39	26	13	0	75.7
Melbourne	33	65	5	56	4	63.7
Sydney	344	508	100	340	68	67.0
Havana	37	52	8	39	5	70.0
Honolulu	61	93	24	61	8	56.0
Kingston	5	6	1	4	1	87.6
Cairo	48	57	10	36	11	77.8
Amman	9	10	3	7	0	56.7
Jerusalem	44	58	4	43	11	57.9
Johannesburg	17	20	14	4	2	66.3
Cape Town	40	57	26	31	0	65.3
Nairobi	25	28	5	19	4	66.3

Table A.2: City Wise Training Dataset Statistics

City Name	#Questions	#QA Pairs	#QA Pairs With Hotel	#QA Pairs With Restaurants	#QA Pairs With Attractions	Avg #Tokens in Question
New York	627	1445	116	1243	86	77.0
Washington	104	243	18	213	12	80.9
Chicago	141	324	16	295	13	74.2
San Francisco	185	439	38	360	41	73.9
Mexico City	14	20	7	9	4	62.4
Miami	13	16	2	9	5	54.7
Vancouver	53	99	26	57	16	74.3
Sao Paulo	1	1	1	0	0	65.0
Buenos Aires	39	82	15	66	1	68.7
London	342	634	76	469	89	75.2
Dublin	62	122	20	97	5	76.9
Rome	118	185	25	139	21	72.7
Stockholm	24	46	9	29	8	82.6
Oslo	9	12	5	7	0	82.9
Zurich	11	16	3	13	0	53.4
Vienna	29	47	9	32	6	55.7
Berlin	39	60	13	37	10	82.6
Budapest	48	96	3	87	6	66.5
Bucharest	2	7	0	7	0	44.5
Moscow	9	14	7	5	2	65.7
Amsterdam	72	113	30	74	9	75.5
Beijing	7	8	0	7	1	65.9
New Delhi	1	3	0	3	0	75.0
Mumbai	20	38	20	14	4	64.0
Agra	4	7	5	2	0	37.5
Bangkok	56	68	32	26	10	64.8
Singapore	46	72	30	41	1	62.2
Jakarta	3	4	3	1	0	43.3
Tokyo	1	1	1	0	0	42.0
Seoul	4	4	3	1	0	50.8
Kathmandu	1	3	3	0	0	12.0
Melbourne	2	2	0	0	2	31.5
Sydney	39	50	12	32	6	80.4
Havana	3	4	0	3	1	75.0
Honolulu	8	8	2	5	1	73.2
Kingston	1	1	0	0	1	78.0
Cairo	12	15	0	14	1	77.8
Amman	3	4	0	3	1	83.3
Jerusalem	6	8	0	5	3	87.3
Johannesburg	4	5	1	4	0	47.5
Cape Town	7	13	6	7	0	70.6
Nairobi	3	3	1	2	0	68.3

Table A.3: City Wise Test Dataset Statistics

City Name	#Questions	#QA Pairs	#QA Pairs With Hotel	#QA Pairs With Restaurants	#QA Pairs With Attractions	Avg #Tokens in Question
New York	621	1362	119	1169	74	75.6
Washington	114	236	20	202	14	74.0
Chicago	140	334	20	293	21	71.6
San Francisco	171	413	55	328	30	78.2
Mexico City	16	20	10	8	2	77.3
Miami	7	8	3	5	0	60.6
Vancouver	61	102	27	65	10	74.2
Sao Paulo	3	8	2	6	0	83.7
Buenos Aires	25	46	13	33	0	79.5
London	334	657	81	494	82	74.5
Dublin	71	125	34	85	6	72.5
Rome	108	166	25	119	22	71.1
Stockholm	17	32	7	18	7	62.9
Oslo	8	9	5	4	0	78.5
Zurich	17	26	12	14	0	73.2
Vienna	37	59	12	36	11	72.8
Berlin	28	46	12	28	6	73.8
Budapest	34	58	3	54	1	66.8
Bucharest	1	2	2	0	0	89.0
Moscow	6	14	4	10	0	57.3
Amsterdam	72	140	11	121	8	72.2
Beijing	3	5	0	4	1	36.0
New Delhi	5	6	1	3	2	24.2
Mumbai	15	32	9	21	2	71.5
Agra	3	5	4	1	0	33.3
Bangkok	55	71	26	32	13	66.3
Karachi	1	1	0	0	1	78.0
Singapore	53	81	37	42	2	69.1
Jakarta	3	8	5	3	0	55.7
Seoul	8	8	6	0	2	57.9
Kathmandu	4	6	6	0	0	86.8
Melbourne	2	4	0	4	0	74.5
Sydney	35	56	4	44	8	62.0
Havana	4	5	1	4	0	72.8
Honolulu	13	15	3	11	1	62.2
Cairo	8	13	2	7	4	64.4
Jerusalem	6	6	0	5	1	40.5
Johannesburg	3	3	1	1	1	77.7
Cape Town	4	5	1	2	2	71.2
Nairobi	3	3	2	0	1	81.7

Table A.4: City Wise Validation Dataset Statistics

Appendix B

Joint Spatio-Textual Reasoning

This appendix is organized as follows.

- Section B.1 provides more details about the artificial dataset used in Chapter 5 and supplementary experimental information that includes additional tables referred to in the chapter.
- Section B.2 gives details about the model hyper-parameters used in Chapter 5.

B.1 Artificial Dataset

We generate a simple, artificial dataset using linguistically diverse templates specifying spatial constraints and locations chosen at random from across 200,000 entities. These entities were sourced from the dataset created in Chapter 4. Each POI entity is labeled with its geo-coordinates apart from other meta-data such as its address, timings, etc. Further, each entity in a city has a specific type viz. Restaurant(R), Attraction(A) or Hotel(H). Table B.1 shows the list of templates used for generating the dataset. These templates have been to make the artificial dataset reflective of real-world challenges. For instance, templates #41-#48 include the possibility of injecting *distractor locations*. To generate questions, $LOCATION and $ENTITY values are updated by randomly selecting values from the POI-set for each entity as described in the next section.

B.1.1 Dataset Generation

To generate a question, a city c, type t and a template T are chosen at random. The "ENTITY" token in each template is replaced by a randomly chosen *metonym* of the type t. Table B.2 shows the list of metonyms for each type. Each instance of the "LOCATION" token is replaced by a randomly chosen entity from the city c and type t. The candidate

Id	Description
1	Do you have any recommendations of *ENTITY* near the *LOCATION*?
2	Does anyone have ideas on *ENTITY* close to *LOCATION*? Thank you!
3	Hello! Could anyone please suggest *ENTITY* in the neighborhood of *LOCATION*?
4	Good Morning! Can someone please propose *ENTITY* not very far from *LOCATION*?
5	Suggestions for *ENTITY* close to both *LOCATION* and *LOCATION*?
6	Some good ideas of *ENTITY* between *LOCATION* and *LOCATION*? Thanks much!
7	Please advise *ENTITY* close to *LOCATION* and not very far off the *LOCATION*.
8	Any ideas for *ENTITY* near *LOCATION* and also close to *LOCATION* would be welcomed?
9	I once lived around *LOCATION*. Does anyone have ideas of *ENTITY* close to the *LOCATION*? Thanks!
10	Any nice suggestions of *ENTITY* near the *LOCATION*? I will be going to *LOCATION* the next day.
11	I just came from *LOCATION*. Someone, please recommend *ENTITY* in the neighborhood of *LOCATION*.
12	Could anyone propose *ENTITY* not far from the *LOCATION*? I need to leave for *LOCATION* urgently.
13	We came from *LOCATION* this morning. Suggestions for *ENTITY* close to both *LOCATION* and *LOCATION*?
14	Any ideas of *ENTITY* between *LOCATION* and *LOCATION*? I would be going to *LOCATION*. Thanks.
15	We might be staying around *LOCATION*. Please advise *ENTITY* close to *LOCATION* and not far from *LOCATION*.
16	Could anyone suggest ideas for *ENTITY* close to *LOCATION* and around *LOCATION*? We could be going to *LOCATION* soon.
17	Any suggestions for *ENTITY* quite far from the *LOCATION*? Thank you very much!
18	Somebody please suggest *ENTITY* cut off from *LOCATION*. Have a good day!
19	Does anyone have suggestions for *ENTITY* away from *LOCATION*? Thanks a lot!
20	Good Afternoon! Any proposals for *ENTITY* not very close to the *LOCATION*?
21	Suggestions on *ENTITY* far from both *LOCATION* and *LOCATION*? Thank!
22	Hi! Any idea of *ENTITY* far away from *LOCATION* and *LOCATION*?
23	Could anyone please propose *ENTITY* not close to *LOCATION* and also far from *LOCATION*?
24	Does anyone have any suggestions for *ENTITY* far from *LOCATION* and not around *LOCATION*?
25	Hey! I will be staying at *LOCATION*. Please suggest *ENTITY* cut off from *LOCATION*.
26	Any pleasant ideas of *ENTITY* far off the *LOCATION*? I might then be visiting *LOCATION*.
27	I came from *LOCATION* this afternoon. Any proposal for *ENTITY* not close to the *LOCATION*?
28	Does anyone have a suggestion for *ENTITY* distant from *LOCATION*? By the way, I came from *LOCATION* yesterday.
29	We will be staying near the *LOCATION*. Suggestions for *ENTITY* far from both *LOCATION* and *LOCATION* will be welcomed.
30	Any idea of *ENTITY* far away from *LOCATION* and *LOCATION*? I would then be visiting *LOCATION*.
31	Hi, I will be staying near the *LOCATION*. Could anyone propose *ENTITY* not very close to *LOCATION* and far from *LOCATION*?
32	Does anyone have suggestions for *ENTITY* far from *LOCATION* and also far from *LOCATION*? I will then be visiting *LOCATION* too.
33	Any good ideas of *ENTITY* far from *LOCATION* but close to *LOCATION* would be appreciated? Best Regards.
34	Anyone having ideas of *ENTITY* close to *LOCATION* but far from *LOCATION*?
35	Someone please advise *ENTITY* far from *LOCATION* but not very far from *LOCATION*.
36	Suggest *ENTITY* close to *LOCATION* but not in the neighborhood of *LOCATION*. Thank you so much!
37	Does anyone have good ideas of *ENTITY* far from *LOCATION* but near *LOCATION*? Regards.
38	Please suggest ideas of *ENTITY* in the neighborhood of *LOCATION* but far from *LOCATION*.
39	Could anyone advise *ENTITY* far from *LOCATION* but not too far from *LOCATION*?
40	Any nice ideas of *ENTITY* close to *LOCATION* but not in the neighborhood of *LOCATION*. Thanks!
41	Tomorrow, I would be coming to stay at *LOCATION*. Anyone having ideas of *ENTITY* close to *LOCATION* but far from *LOCATION*?
42	Please propose *ENTITY* far from *LOCATION* but not far from *LOCATION*. I will then be exploring *LOCATION*.
43	I came from *LOCATION* this evening. Any nice ideas for *ENTITY* far from *LOCATION* but close to *LOCATION* would be appreciated?
44	Suggest *ENTITY* close to *LOCATION* but not near *LOCATION*. Tomorrow, I will be leaving for *LOCATION*.
45	Yesterday, I came to stay at *LOCATION*. Any ideas of *ENTITY* close to *LOCATION* but far from *LOCATION*?
46	Suggestions of *ENTITY* far from *LOCATION* but not very far from *LOCATION*. I will then be moving to *LOCATION*.
47	I came from *LOCATION* today. Any good ideas for *ENTITY* far from *LOCATION* but near to *LOCATION* would be welcomed?
48	Advise *ENTITY* close to *LOCATION* but not close to *LOCATION*. I might be leaving for *LOCATION* soon.

Table B.1: Templates used for generating the artificial dataset

set consists of the entities from the city c and type t. The entities used as location mentions are sampled without replacement and removed from the candidate set.

The gold answer entity is uniquely determined for each question based on its template. For example, consider a template T, "*I am staying at $A! Please suggest a hotel close to $B but far from $C.*" The score of a candidate entity X is given by $dist_T(X) = -(dist(X, B) - dist(X, C))$ (distances from B needs to be reduced, while distance from C needs to be higher). A is a distractor. The candidate with the $max(dist_T(X))$ in the universe is chosen as the gold entity for that question.

Each question further consists of 500 negative samples (35% hard, 65% soft). The negative samples are generated as a part of the gold generation process. A hard negative sample has a $dist_T(X)$ value closer to the gold as compared to a soft negative sample. We release the samples used for training along with the dataset for reproducibility.

Entity type	Metonyms
R (Restaurant)	a restaurant, an eatery, an eating joint, a cafeteria, an outlet, a coffee shop, a fast food place, a lunch counter, a lunch room, a snack bar, a chop house, a steak house, a pizzeria, a coffee shop, a tea house, a bar room
H (Hotel)	a hotel, an inn, a motel, a guest house, a hostel, a boarding house, a lodge, an auberge, a caravansary, a public house, a tavern, an accomodation, a resort, a youth hostel, a bunk house, a dormitory, a flop house
A (Attraction)	an attraction, a tourist spot, a tourist attraction, a popular wonder, a sightseeing place, a tourist location, a place of tourist interest, a crowd pleaser, a scenic spot, a popular landmark, a monument

Table B.2: List of metonyms for each entity type in the artificial dataset

B.1.2 Template classes

We create templates (Table B.1) that can be broadly divided into three different categories based on whether the correct answer entity is expected to be: (1) close to one or more locations [1-16] (2) far from one or more locations [17-32] (3) close to some and far from others (combination) [33-48]. To make the task more reflective of real-world challenges we also randomly insert a *distractor* location that does not need to be reasoned. The second-half for each category (i.e. [9-16], [25-32], and [41-48]) consists of templates that have a distractor locative reference. Further, for the close (or far) category, the templates could contain one location ([1-4] + [9-12]) or two locations ([5-8] + [13-16]) that need to be reasoned for close (or far).

B.2 Model settings

B.2.1 Experiments on artificial dataset

The hyperparameters for the best performing configurations of all models were identified through manual testing on the validation set (Table B.3). The models were trained on a 2x NVIDIA K40 (12GB, 2880 CUDA cores) GPU on a shared cluster. The BERT models were trained with a learning rate (LR) of 0.0002 and non-BERT models with 0.001.

Hyperparameter	Value
Negative samples	40
Batch size	20
Optimizer	Adam
Loss	MarginRankingLoss
Margin	0.5
Max no. of epochs	15
GRU Input dimension	131
GRU Output dimension	32
DRL Block Layer 1	64 (Input) 64 (Output)
DRL Block Layer 2	64 (Input) 64 (Output)
DRL Block Layer 3	64 (Input) 64 (Output)
DRL Block Layer 4	64 (Input) 1 (Output)

Table B.3: Hyperparameter settings for experiments on the artificial-dataset

B.2.2 Spatio-textual Reasoning Network

Hyperparameter	Value
Word embeddings size	128
Dropout	0.2
Optimizer	Adam
Loss	Hinge Loss
Margin	1.0
Batch Size	200
SPNet GRU input dimension	131
SPNet GRU output dimension	256
Textual GRU input dimension	128
Textual GRU output dimension	256
DRL Block Layer 1	512 (Input) 256 (Output)
DRL Block Layer 2	256 (Input) 256 (Output)
DRL Block Layer 3	256 (Input) 128 (Output)
DRL Block Layer 4	128 (Input) 128 (Output)
DRL Block Layer 5	128 (Input) 50 (Output)
DRL Block Layer 6	50 (Input) 10 (Output)
DRL Block Layer 7	10 (Input) 1 (Output)
α,β FF Linear Layer 1	256 (Input) 50 (Output)
α,β FF Linear Layer 2	50 (Input) 50 (Output)
α,β FF Linear Layer 3	50 (Input) 10 (Output)
α,β FF Linear Layer 4	10 (Input) 10 (Output)
α,β FF Linear Layer 5	10 (Input) 10 (Output)
α,β FF Linear Layer 6	10 (Input) 2 (Output)

Table B.4: Hyperparameters used for experiments on the end-task

The hyperparameters for the best performing configuration were identified through manual testing on the validation set. The Spatio-Textual Reasoner was trained on 4 K-80 GPUs on a shared cluster.

Appendix C

Pairs used for Comparisons

City Pairs
NYC vs Rome
Mumbai vs Sydney
Lisbon vs Stockholm
Honolulu vs NYC
Granada vs Seville
Granada vs NYC
Edinburgh vs Prague
Edinbugh vs Dublin
Darjeeling vs Munnar
Beijing vs Mumbai
ChicagoVsRome
Bucharest vs Prague
Boston vs NYC
Rio vs Mexico
NYC vs Bombay
Moscow vs Berlin
Honolulu vs Sydney
Honolulu vs Santorini
Honoulu vs Kingston
Rome vs Paris
Seoul vs Bejing
Seoul vs NYC
Sydney vs Canberra
Vancouver vs Chicago
Prague vs Beijing
Vancouver vs SFO
Sydney vs Vancouver
Shillong vs Mumbai
Rome vs Beijing
Rome vs Athens

Table C.1: City Pairs used for comparing clustering algorithms

Movie Pairs
Batman (1989) and Gandhi
Ace Ventura (Pet Detective) and The Mask
Deep Impact and Twister
Ace Ventura (Pet Detective) and Gandhi
The Santa Claus and Mrs. Doubtfire
Deep Impact and Men In Black
Deep Impact and Back to the Future
Back to the Future II and Blindness (Ceguera)
Alladin and Men In Black
Back to the Future and Godzilla
My name is Nobody and 40 days and 40 nights
Nadia 2 and Frank Spandone
The Crusades: Terry Jones and Lilies of the Field
Marked for Death and Runaway Bride
The Triangle and Slipstream
The Adventures of Pete & Pete - Season 2 and Molly: An American Girl on the Home Front
The Adventures of Pete & Pete - Season 2 and WWE 2
Due South Season 2 and The Canterville Ghost
The Great Race and Barnyard

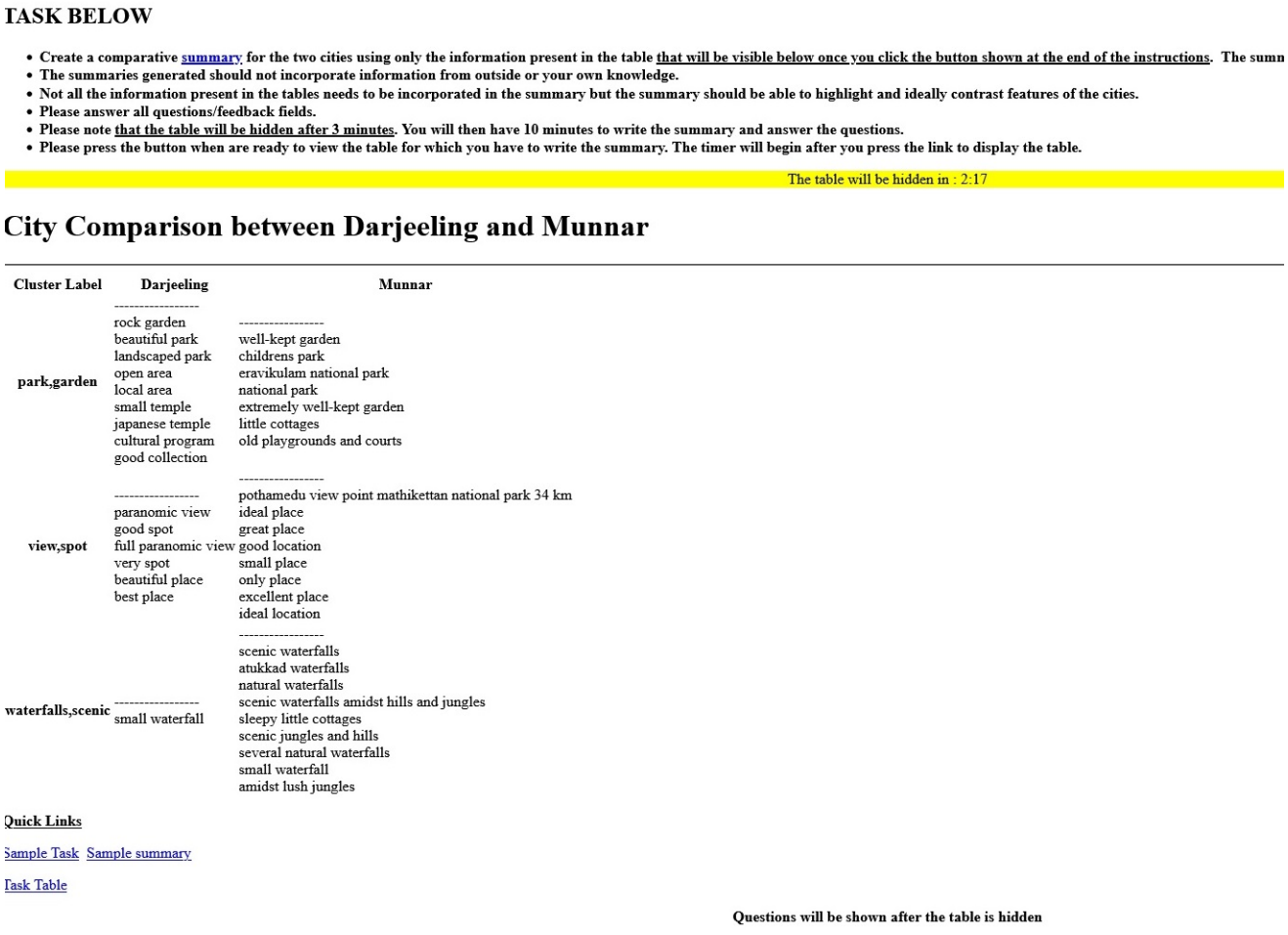
Table C.2: Movie Pairs used for comparing clustering algorithms

City Pairs
Honolulu vs Santorini
Darjeeling vs Munnar
Edinburgh vs Prague
Edinburgh vs Dublin
Rio vs Mexico
Lisbon vs Stockholm
Moscow vs Berlin
Granada vs NYC
Sydney vs Canberra
NYC vs Rome
Bucharest vs Prague
Vancouver vs Chicago
Seoul vs Beijing
Chicago vs Rome
Sydney vs Vancouver
Rome vs Paris
Beijing vs Mumbai
Honolulu vs NYC
Prague vs Beijing
Shillong vs Mumbai

Table C.3: City Pairs used for evaluating summaries created by crowd source workers using the comparisons outputs from EB G-pLSA and by using full Wikipedia articles

Appendix D

Answering Comparison Questions: Screenshots of Crowd-worker Tasks



TASK BELOW

- **Create a comparative summary for the two cities using only the information present in the table that will be visible below once you click the button shown at the end of the instructions. The sumn**
- **The summaries generated should not incorporate information from outside or your own knowledge.**
- **Not all the information present in the tables needs to be incorporated in the summary but the summary should be able to highlight and ideally contrast features of the cities.**
- **Please answer all questions/feedback fields.**
- **Please note that the table will be hidden after 3 minutes. You will then have 10 minutes to write the summary and answer the questions.**
- **Please press the button when are ready to view the table for which you have to write the summary. The timer will begin after you press the link to display the table.**

The table will be hidden in : 2:17

City Comparison between Darjeeling and Munnar

Cluster Label	Darjeeling	Munnar
park,garden	----------	----------
	rock garden	well-kept garden
	beautiful park	childrens park
	landscaped park	eravikulam national park
	open area	national park
	local area	extremely well-kept garden
	small temple	little cottages
	japanese temple	old playgrounds and courts
	cultural program	
	good collection	
view,spot	----------	----------
	----------	pothamedu view point mathikettan national park 34 km
	paranomic view	ideal place
	good spot	great place
	full paranomic view	good location
	very spot	small place
	beautiful place	only place
	best place	excellent place
		ideal location
waterfalls,scenic	----------	----------
	----------	scenic waterfalls
	small waterfall	atukkad waterfalls
		natural waterfalls
		scenic waterfalls amidst hills and jungles
		sleepy little cottages
		scenic jungles and hills
		several natural waterfalls
		small waterfall
		amidst lush jungles

Quick Links

Sample Task Sample summary

Task Table

Questions will be shown after the table is hidden

Figure D.1: Sample task screenshot where users were shown the comparison tables before writing summaries. A live timer displayed current time left for task. Screenshot truncated for ease of presentation.

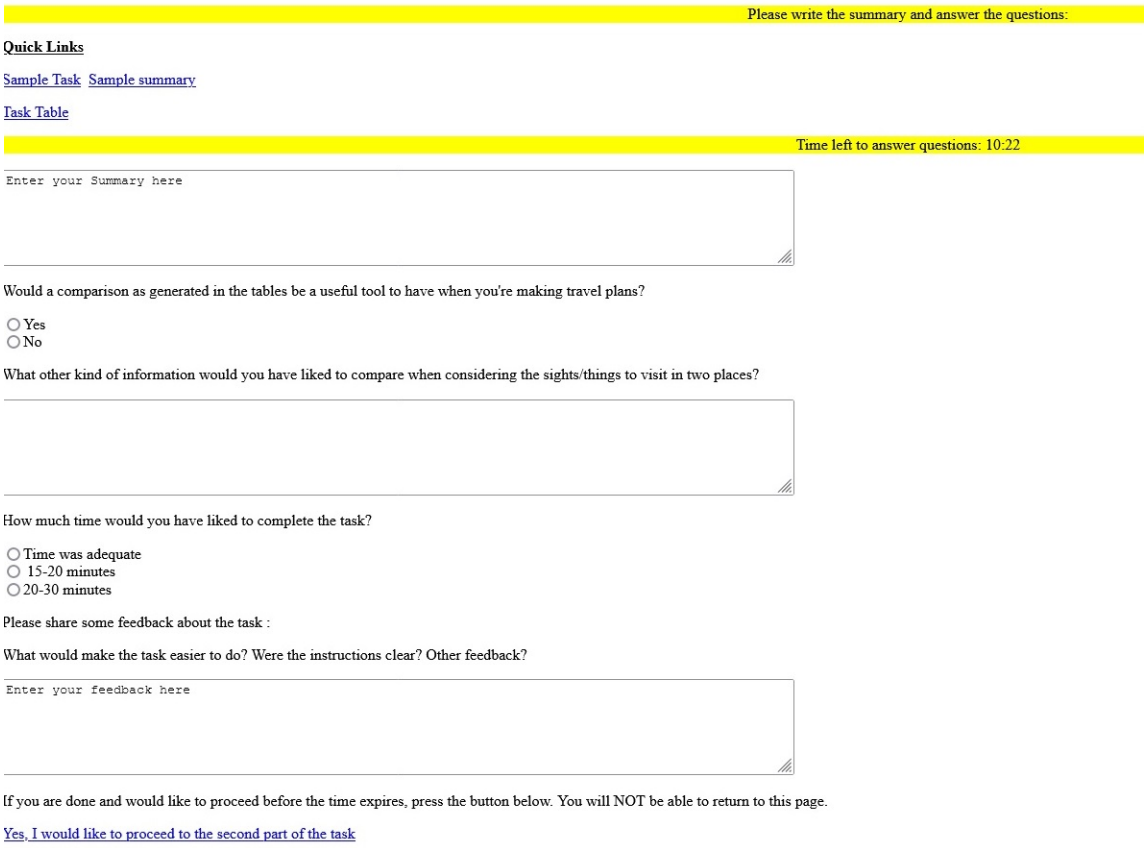


Figure D.2: Sample task screenshot where users were asked to write summaries after viewing the comparison table. A live timer displayed current time left for task. Screenshot truncated for ease of presentation.

TASK BELOW

- **Create a comparative summary for the two cities using only the information presented which will be visible below once you click the button shown at the end of the instructions. The summary created should be between 100-300 words.**
- **The summaries generated should not incorporate information from outside or your own knowledge.**
- **Not all the information presented needs to be incorporated in the summary but the summary should be able to highlight and ideally contrast features of the cities.**
- **Please answer all questions/feedback fields.**
- **Please note that the text will be hidden after 3 minutes. You will then have 7 minutes to write the summary and answer the questions.**
- **Please press the button when are ready to view the cities' information for which you have to write the summary. The timer will begin after you press the button.**

The text will be hidden in : 2:27

Sydney

Landmarks

Sydney Harbour

Sydney cityscape at dusk, viewed from the North Head lookout

The Sydney Harbour Bridge crosses the harbour from the The Rocks to North Sydney. There are many different experiences centred around the bridge. You can walk or cycle across, picnic under, or climb over the Harbour Bridge. See the details in The Rocks.

The Sydney Opera House. The Sydney Opera House is simply one of the most famous structures ever built. It is in the city centre.

Darling Harbour is a large tourist precinct and includes a range of activities, restaurants, museums and shopping facilities.

Sydney Olympic Park. Home of the 2000 Olympics and now parklands and sporting facilities.

Luna Park, 1 Olympic Dr, Milson's Point, tel. 02 9033 7676. Is a large theme park situated near the Sydney Harbour Bridge. Its mouth-shaped entrance can be seen from many areas of Sydney as well as the large Ferris Wheel.

Sydney Tower also called Centrepoint Tower or AMP Tower. The tallest structure in Sydney, the tower contains a buffet, cafe and a rather large restaurant and attracts many visitors a year. The tower is in the City Centre

St Mary's Cathedral. Sydney's main catholic cathedral. Corner of St Mary's Road and College St. The cathedral is in the City Centre.

Royal Botanic Gardens- The Royal Botanic Gardens were first established in Sydney by Governor Bligh in 1816. The gardens cover 30 hectares and adjoin the 35 hectares making up the Domain, there are over 7500 species of plants represented here. The gardens are at the north eastern corner of the City Centre and overlook Sydney harbour.

Historical areas

La **Perouse**

The Rocks has sites preserved from Sydney's early settlement.

Parramatta to the west of Sydney is the site of many of Sydney's oldest buildings from colonial times.

Macquarie Street in the City has a string of historical sites, from the first hospital in the colony, to the Mint to Hyde Park Barracks, to the Conservatorium which was the original government house stables. Sydney Hospital was first known as "The Rum Hospital", it was the first major building established in the colony.

La **Perouse**, near Botany Bay, in Sydney's Eastern Suburbs contains the grave of an early French explorer, museum, and old fort.

The walk from Manly to Middle Head passes many coastal artillery fortifications built into the cliffs of Sydney Harbour during the late nineteenth century.

Mrs Macquarie's Chair and walk near the Botanical Gardens in the City

Anzac War Memorial at the eastern end of Hyde Park in the City Centre. The memorial commemorates the memory of those Australians who lost their lives during war. It houses a small museum, an impressive statue and the Pool of Remembrance. Sydney's Anzac War Memorial was built in the 1930s.

Museums and galleries

Some of Sydney's museums are free to enter including the Art Gallery of New South Wales and the Museum of Contemporary Art. You may be charged to enter certain exhibitions. Sydney Museums generally do not have 'free days' that you can find in other parts of the world but some historic houses may be free on certain public holidays, though tend to attract large crowds.

The Australian Museum is much the old style natural history museum. Usually a special exhibition on as well. The museum is near Hyde Park in City Centre.

The Australian National Maritime Museum has inside and outside exhibitions - much of the history of Australia is a maritime one, and much of it is in this museum in Darling Harbour.

The Art Gallery of NSW has mostly classical, but some modern and Aboriginal art. Near the Botanical Gardens in the city centre.

The Powerhouse Museum has some buttons to push, some technology, but some interesting displays of Sydney in the 1900s, in the City West in Ultimo, right on the boundary with Darling Harbour. Exhibits designed for children also.

The Museum of Contemporary Art [24] in the city centre, near Circular Quay.

The Museum of Sydney [25] in the city centre.

Or see one of the smaller chic Art Galleries in East Sydney.

Wildlife

In captivity

Kangaroos

Taronga Zoo :Taronga Zoo Large zoo whose animals have the best view in the world, a short ferry trip from the City on the North Shore. The Koala Park Sanctuary in the Outer West. Sydney Aquarium [26] in Darling Harbour. Sydney Wildlife World' adjacent to the aquarium in Darling Harbour. Featherdale Wildlife Park in Western Sydney and just out of Sydney, the Australian Reptile Park [27], about an hour north of Sydney, has kangaroos, wallabies, dingos, and more.Symbio Park in Helensburgh. In the wild Whale Watching see whales migrating the Pacific coast. There are boats from Darling Harbour or Circular Quay. Bats (Flying foxes) nest next to the fernery in the Botanic Gardens in the city, and fly to feed over the city buildings and Harbour Bridge at dusk, you can see them on the eastern side of the Opera House at sunset. Rainbow Lorikeets swarm around the trees in many suburbs at dusk, making a tremendous chatter Sulphur Crested Cockatoos are commonly seen in the leafier suburbs all day. Ibis are an unusual wader bird, that has made its home in the suburbs, especially in Hyde Park in the city Possums are a native marsupial at home in the urban environment. Look up carefully in tree lined streets, or in Hyde Park after dark. Locals regard these critters as somewhat of a nuisance as they have a habit of nesting in the warmth of house roofs and love to brawl noisily at about 2am above your bedroom. Kangaroos & Wallabies. These can be spotted with patience in most of the Sydney National Parks, including the Royal National Park, ask the local rangers where they tend to be seen in the late afternoons. This is a great way to experience Australia's native wildlife in their natural habitat compared to seeing these amazing animals confined in zoos, but requires considerably more time and patience.

Sydney Harbour

Sydney Opera House

Yachts in Sydney Harbour; business district in background

Sydney's large natural harbour was the reason that the original penal settlement was established in the area, near what is now known as Circular Quay. It is now well developed, with skyscrapers, highrises, and houses all around its shores, but it is still very beautiful.

The harbour is served by ferry services that transport passengers around the harbour. An excellent way to see both the harbour and Sydney attractions is to take a ferry east from Circular Quay to Taronga Zoo or Manly or west under the Harbour Bridge towards Parramatta. These are reasonably priced and a favourite for tourists. If time is short, for a shorter route, the ferry between Circular Quay and Darling Harbour will let you ride under the Harbour Bridge and see the central part of the harbour.

Catch a ferry from Circular Quay to Manly. Before returning to the Sydney CBD, walk from the Manly ferry wharf along the Manly Corso to famous Manly Beach. A great day, afternoon or evening out at a fraction of the price of a commercial harbour cruise.

You can take a cruise on Sydney Harbour. There are many cruises to choose from and they depart from Darling Harbour or Circular Quay. For a bigger adrenalin rush, try the jet boats that zip around the harbour [28] at breakneck speeds.

Sydney Harbour can be viewed from the city or from on of the many walks next to it, most of which are easily accessible by ferry or bus.

The world famous Sydney to Hobart Yacht Race begins every year on Boxing Day, on Sydney Harbour. Thousands of spectator craft take to the water to farewell the yachts as they set off on their grueling journey to Hobart. Seaworthy craft can follow the yachts through the Sydney Heads into the open ocean. You can also see the race from a harbour vantage point like Watsons Bay. where you can see them sail towards you across the harbour, and then cross to the gap to see them sail down the coast.

You can visit the Harbour Islands by ferry or water taxi.

Swing by the Royal Botanic Gardens [29] and the Art Gallery of New South Wales [30] on the edge of the gardens. While you're in the area visit Mrs Macquarie's Chair for a picture postcard view of the Sydney Harbour Bridge and Opera House in one picture. You may have to compete with the numerous wedding couples on weekends.

Scenic Flights Adventures and Flight Training, +61 2 9791 0643 (contact@redbaron.com.au) [31]. A fantastic way to see Sydney Harbour is from the air. Red Baron Adventures do scenic flights over Sydney Harbour and the Northern Beaches most days of the year (weather permitting) in an open cockpit Pitts Special bi-plane. They also have heart stopping Aerobatic Flights available for the more adventurous (note: these are not done over Sydney Harbour). Flights range from $440 to $660 and go for between 45 min and 80 minutes.

Aboriginal Sydney

Far from being confined to the inland areas, Aboriginal people extensively occupied the Sydney area prior to the arrival of European settlers.

Rock Carvings, can be seen in the Royal National Park - catch the train and ferry to Cronulla and Bundeena. There are extensive carvings in Kuringai National Park, near West Head that are accessible only by car. Closer to the city, there are examples at Balls Head and Berry Island, near to Wollstonecraft station. There is an interpretive walk at Berry Island.

Meeting of Civilisations. Interpretive centre is at the site of the landing place of Captain Cook, at Kurnell.

Bangarra Dance Theatre, is a modern dance company, inspired by indigenous Australian themes.

Aboriginal Art. A wander through The Rocks and you will find many places exhibiting and selling contemporary Aboriginal art. The Art Gallery of New South Wales the City Centre has an Aboriginal and Torres Strait Islander Gallery, which is free to visit.

Canberra

Museums and other institutions

City area - North of Lake Burley Griffin

Australian War Memorial

Australian War Memorial, Treloar Crescent (top of ANZAC Parade, at the other end from Parliament House), ph +61 2 6243-4211 or +61 2 6243-4598 (for recorded information), fax +61 2 6243 4325, [18]. Daily 10AM-5PM. Not just a memorial, this is one of Australia's premier museums, covering Australian military history from Federation to the present day and including fascinating exhibits of equipment, memorabilia and battle dioramas. You could easily spend a full day here (it has a café, or bring a picnic lunch if the weather is nice and sit on the lawns at the front). Anzac Parade, leading up to the War Memorial has a number of memorials to different wars and those involved in wars. Free entry, allow 4-7 hours.

Canberra Museum and Gallery, Cnr London Circuit & Civic Square, Civic, [19]. Tue-Fri 10AM-5PM; Sat-Sun 12PM-5PM. A museum and art gallery featuring works and exhibits of the local region. Also features the Sydney Nolan Collection - the works of Sir Sydney Nolan, a famous Australian artist. Free. edit

Figure D.3: Sample task screenshot where users were shown the full articles before writing summaries. A live timer displayed current time left for task. Screenshot truncated for ease of presentation.

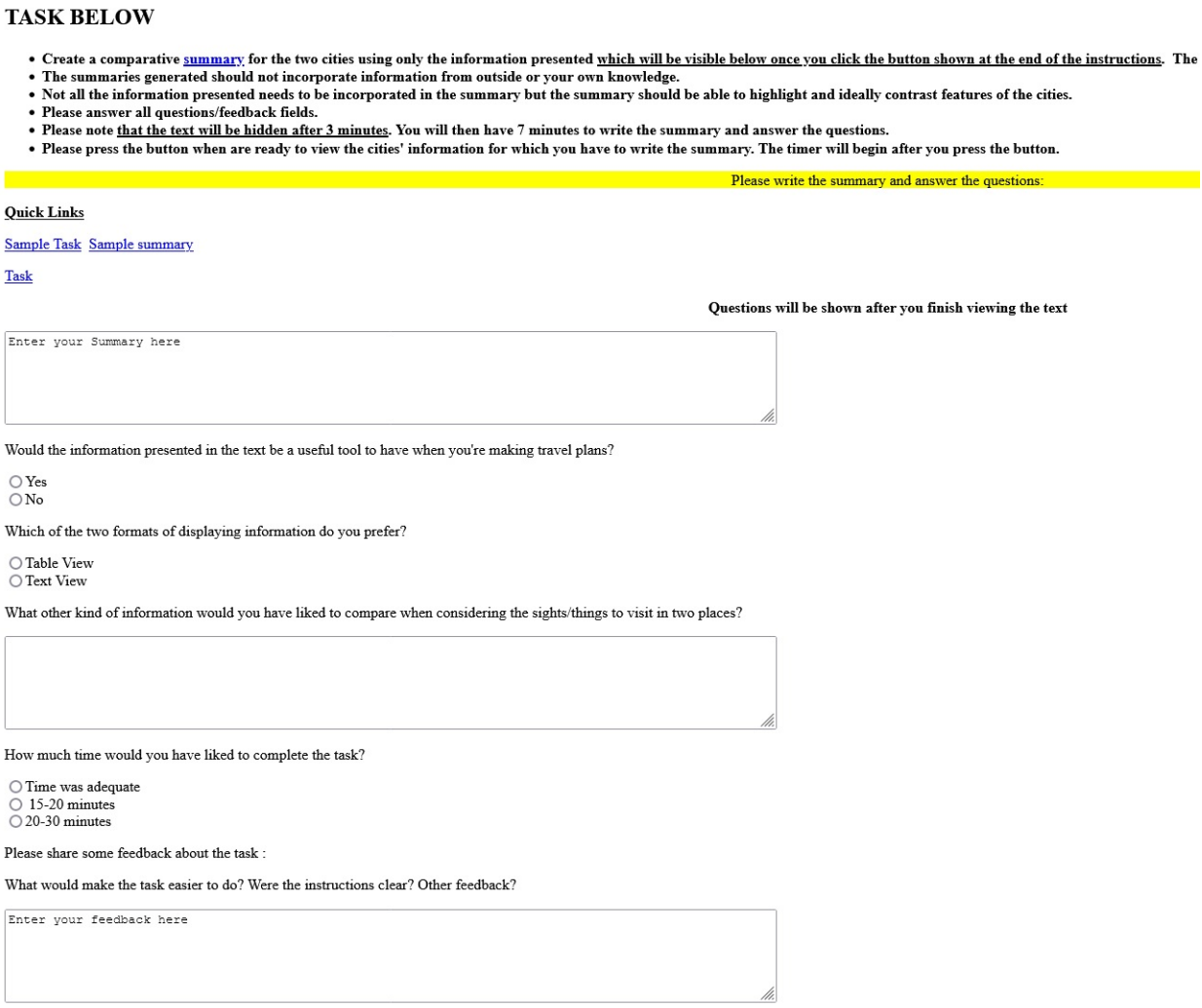
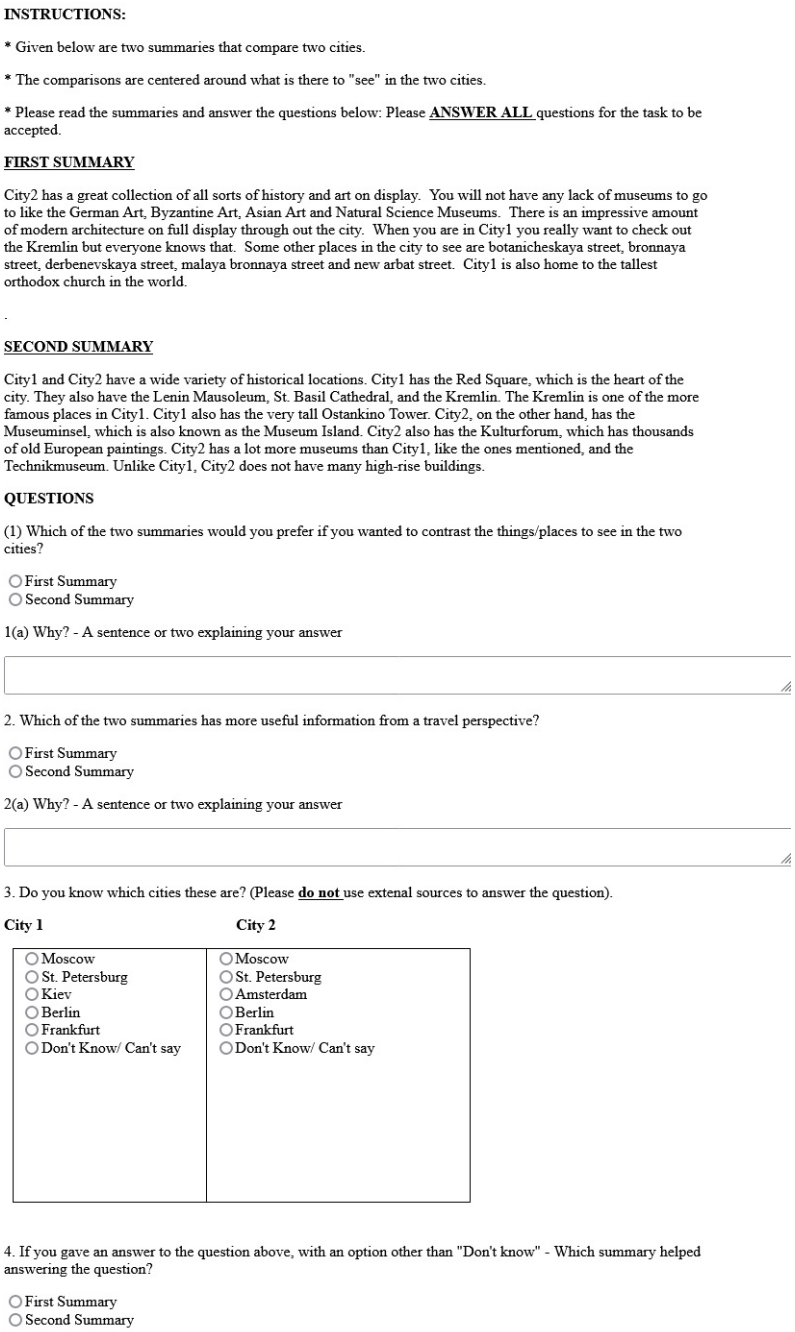


Figure D.4: Sample task screenshot where users were asked to write summaries after viewing the full articles. A live timer displayed current time left for task. Screenshot truncated for ease of presentation.



INSTRUCTIONS:

* Given below are two summaries that compare two cities.

* The comparisons are centered around what is there to "see" in the two cities.

* Please read the summaries and answer the questions below: Please **ANSWER ALL** questions for the task to be accepted.

FIRST SUMMARY

City2 has a great collection of all sorts of history and art on display. You will not have any lack of museums to go to like the German Art, Byzantine Art, Asian Art and Natural Science Museums. There is an impressive amount of modern architecture on full display through out the city. When you are in City1 you really want to check out the Kremlin but everyone knows that. Some other places in the city to see are botanicheskaya street, bronnaya street, derbenevskaya street, malaya bronnaya street and new arbat street. City1 is also home to the tallest orthodox church in the world.

.

SECOND SUMMARY

City1 and City2 have a wide variety of historical locations. City1 has the Red Square, which is the heart of the city. They also have the Lenin Mausoleum, St. Basil Cathedral, and the Kremlin. The Kremlin is one of the more famous places in City1. City1 also has the very tall Ostankino Tower. City2, on the other hand, has the Museuminsel, which is also known as the Museum Island. City2 also has the Kulturforum, which has thousands of old European paintings. City2 has a lot more museums than City1, like the ones mentioned, and the Technikmuseum. Unlike City1, City2 does not have many high-rise buildings.

QUESTIONS

(1) Which of the two summaries would you prefer if you wanted to contrast the things/places to see in the two cities?

○ First Summary
○ Second Summary

1(a) Why? - A sentence or two explaining your answer

2. Which of the two summaries has more useful information from a travel perspective?

○ First Summary
○ Second Summary

2(a) Why? - A sentence or two explaining your answer

3. Do you know which cities these are? (Please **do not** use extenal sources to answer the question).

City 1	City 2
○ Moscow	○ Moscow
○ St. Petersburg	○ St. Petersburg
○ Kiev	○ Amsterdam
○ Berlin	○ Berlin
○ Frankfurt	○ Frankfurt
○ Don't Know/ Can't say	○ Don't Know/ Can't say

4. If you gave an answer to the question above, with an option other than "Don't know" - Which summary helped answering the question?

○ First Summary
○ Second Summary

Figure D.5: Sample task screenshot where users were asked compare written summaries. Screenshot truncated for ease of presentation.

Appendix E

System Outputs

E.1 Recommendation Questions

Each example below shows a question, the answer entity returned by the spatio-textual CSRQA model as well as the representative review document used by the model. As can be seen these are still pretty long and do not have any structure within them.

Examples 1-3 below show how joint spatio-textual reasoning helps return better answers as compared to a baseline CsrQAwithout spatial reasoning. Examples 4 and 5 show how models can get answers right or wrong depending on the information available in the knowledge sources. Incorrect answers are shown in red, while correct answers are shown in green. Text from review documents, that may be helpful for answering, is shown in blue.

E.1.1 Example 1: Restaurant recommendation with location constraints (Correct answer returned)

Question: '*Any suggestions where to eat within walking distance of the Washington monument? Something moderate that will work for a group including teenagers - grandparents. Thanks.*"

City: Washington DC, USA

Entity Type: Restaurant

Answer returned by Spatio-Textual CsrQA: Old Ebbitt Grill, 675 15th St NW, Washington DC, DC 20005-5702

Distance from location in question: Approximately 0.7 miles.

Comment: As can be seen in the review document below the entity is well rated for its food, service and ambience. See text in blue. It is also just 0.7 miles from Washington

Monument.

Review Document of Entity: *This restaurant is fantastic for great food and great dining. Great food - would highly recommend this restaurant but be prepared for a long wait time. Went there this past thursday with friends and I must say that the service was fantastic and the food was really good. The food is good and the service excellent, but I think its the decor and location that bring us back. The food was excellent and reasonably priced for the quality. With the history, kind staff, and delicious food this restaurant is a must when you visit DC. This restaurant has it down pat; the service is excellent, the food is yummy and the prices are not bad at all for such a fine dining establishment. Very good food, great service and just a fun place to go! This place came highly recommended and although the food was really good, I just wasn't impressed with the service. Food was great, service was also excellent and it's a great location right near the White House. We had a great time in this restaurant. We had a nice dinner. We were in & out in an hour. We had a very nice evening. We went here for our anniversary. We had a great waiter. We went one day for breakfast! We all had a great experience. We were very pleased. It was late and we were so hungry. The food was good and the service was fast. The service was great too. Service was really slow. The food was great and the service was excellent. The food was great and the service was exceptional. The service was great also. Service was excellent for such a large and busy place. Service was very good and attentive. The service was friendly, timely, and the food was very good. Service was outstanding. We had oysters, chowder, crab cakes, pork hops and red snapper. The crab cakes are truly delicious and the ribeye was perfectly cooked and so tender! Crab artichoke appetizer was tasty although the big chunks of artichoke made it difficult to eat. Oysters were fresh and delicious, had a wonderful grilled chicken salad as an entree, and the pecan pie was amazing! We enjoyed the artichoke crab dip, the calamari, oyster stew and chowder. The jambalaya was delicious and full of seafood. We had a delicious burger, clam chowder and the best salmon cake I have tasted, washed down with a couple of the special ales. Calamari and Artichoke dip were our appetizers, and left no room for improvement. Many in the group started with the fried oyster chowder with bacon and kale in a rich cream sauce (yes, I said "fried"); one of the best oyster stews that I have had anywhere. I had the sauteed seafood salad - it was fabulous! A friendly bartender and a good selection of beer on tap made the hour wait for dinner go fast. The drinks are delicious and it's a lively spot with diverse people. From the pre dinner drinks to the excellent entrees, it's a great place to go. You can get everything on the menu at the bar, and the bartenders have great stories. The bar is always so much fun, and the Mojitos are fantastic. We ate at bar due to the crowd and the bartenders do an excellent job. The drinks are always well done. Fun atmosphere and great drink menu for the adults! I popped in here for a drink and some happy hour (3-6pm) half price oysters, but came away disappointed. Cool bar to have a drink while waiting for a table, great food (recommend the filet mignon) with excellent service The place was packed, but we had reservations and were*

taken to our table right away. We had no reservations, but it was a Thursday evening, and we were seated in 30 minutes. We were there for an early dinner about 6:00 and had to wait about 40 minutes to get a seat - the bar was FULL+. The place was packed when we arrived for a 6:30 reservation, but we were led to a nice booth in the oyster bar area almost immediately. The restaurant was very busy, but we had a reservation and were seated promptly. We were seated near reception desk which was less than ideal and very crowded but we were glad to have a table without reservation (we came at 4:30pm) on a busy long weekend. It was a very busy place but we had reservations and were seated very quickly. The restaurant was filling up quickly, but we were seated immediately on a Saturday morning with reservations through Open Table. It was very busy when we got there for lunch and we did not have a reservation, however there were two spots of the bar available so we sat there. We showed up without reservations on a Friday night and we had to wait 45 minutes (which was well worth it). Very crowded but worth the wait. The ambiance is awesome and the place is very busy. This place is always packed, and it's worth it. This place is a must if you are coming to D. C. Crowded at times, but worth the wait. Expect it to be very crowded as the place is very popular. Expect to wait, but with three different bars and the history of the place it is well worth it. This is usually a solid place to go if you're in the area. This place is a must see when in DC. It is a great place just to stop by for a drink as well. Not a good start. So much history! it's all good. Always a pleasure. always a good sign. full of tourists,like us. Now the rest. An excellent establishment. for this meal. Not so much! This is a neat place that is close to the White House area. This old place is a place of history and convenient to any hotel near the White House and the National Mall. This is such a historic landmark in DC and is in a great location near the White House. There is a lot of Presidential history to this place and only a couple blocks from the White House. Located a block from the White House, it is teeming with tourists and locals alike. It's in a great location right by the White House and close to the Mall. Oldest saloon in DC, 1856. Located 2 blocks from the White House and down the street from the W and The Willard hotels. This historic must see is right across from the White House. Across the street from Treasury, this classy (and classic) Washington experience features an amusing mix of tourists and bureaucrats. We had a great meal at this restaurant. We went for breakfast and the food was really good. My friends and I had dinner here at the food was amazing. We enjoyed a nice meal at lunch prices. We were seated before they said we would be, the food was delicious and it's charming. I enjoyed my meal completely here at Old Ebbitt Grill. Once seated we found the food was delicious and the best we had during our time in DC. We had an amazing meal at the Old Ebbitt Grill. The dinner entrees were good, according to my friends. We had a great brunch at the Old Ebbitt Grill.

Answer returned by baseline CsrQA: Jaleo 480 7th St NW Washington DC, DC 20004-2207

Distance from location in question: Approximately 1.2 miles.

Comment: While this entity is well rated for its food and services, it is twice as far as away as alternatives and is also more expensive.

Review Document: *The place is very nice and friendly. Fun place for drinks and tapas and appropriately casual. Just a 10 minute walk from the Mall, this place is great anytime you need a sightseeing break. Great venue for food and beverage but really noisy and throngs of people on the weekend. We went here during the week when it wasnt too busy. Went there for my birthday while in DC for the long weekend and was very happy with it. Its a great place to visit with family. Great place for lunch or dinner. I have been to this location 4-5 times. And it was a close walk from our hotel too. The food was very good, nice menu, the service was great, and the decor of the restaurant was wonderful. We visited Jaleo's during Restaurant Week, and we found that it continues to be one of the best restaurants in downtown Washington. Besides the incredible variety of the Menu, the quality of the food is amazing, just as if you were in Spain. Jaleo is an amazing restaurant with a swanky but cool decor. The food was delicious and very reasonable for the pre-theater menu option. Food was excellent and the portions were great for lunch. The food, the service and the ambience is totally worthwhile. The Sangria was ok, not the best I've had but the food was really good. Longevity in the restaurant biz isn't easy, but good food is good food, and this place delivers it. The food was delicious and like most restaurants in DC, the atmosphere was lively. The only thing that we did not like was the paella. It was all very tasty. All were very good. we enjoyed their paella. It was all very yummy. it was very very good! We didn't have the best start. It was a lot of fun. It was nice to not have to make any choices. Everything was just perfect. Service was very average!! Service was very good. Service was excellent. Service was professional and personable. Service was terrible. The service was perfect. The service was good and the food was GREAT! The service was excellent and do was the food! First: food is really good. The food was delicious (although pricey!) The food was fabulous. The food was even more amazing. Liked the food as it was very good. All of our food was exceptional. The food was excellent. We enjoyed this meal very much. The food is a good interpretation of neo-classic dishes from Madrid, Basque and Catalan. Excellent tapas and other Spanish food. This is the place if you are too and want to try Spanish Tapas and small plates. One of my all-time favorite restaurants serves small dish Spanish tapas in our nation's capitol. We have eaten Tapas at the best restaurants in Spain and it is much better in Barcelona but for the United States this is really good. Jaleo is a Spanish tapas restaurant with a smart, modern interior to admire while you're meals are prepared. If you like tapas and love Spain than this is the place to go. As a Spaniard, I can say that this restaurant offers the best tapas in Washington DC. The food is authentic Spanish; we love to get the tapas. The food is just wonderful but it misses the rustic, authentic Spanish taps. All in all, a nice meal. Always a good experience. We'll definitely return. Lots of fun Very trendy. All simply, delicious. Jaleo is indeed a very*

good restaurant. Jaleo is a lovely restaurant. Very good service. Definitely GO! The shrimp and garlic were perfectly cooked, and the flan for dessert was the best I have ever eaten anywhere. As our main we sampled perfectly cooked scallops, sautéed spinach, but the stars of our main dishes were the butifarra with fried white beans and gambas al ajillo - homemade pork sausage and garlic shrimp!!! We particularly liked the goat cheese, fried dates, potato wrapped chorizo, sauteed spinach and shrimp and the pan. We ordered three tapas; chorizo, patatas bravas, and one with lots of peas and vegetables. I had the roasted lamb sandwich with tomatoes and olives- it was incredibly flavorful! The Shrimp dish and the Chorizo sausage were my favorites. Started off with the tomato bread and a trio of the smoked meats, which I believe were the iberico, chorizo and serrano, then came the endive with goat cheese and citrus, a roasted onion with pine nuts and blue cheese. This time the Rossejat - toasted paella with shrimp and calamari was our favorite. the sangria was delicious, and the scallops were perfect. Liquid olives were amazing, chorizo and potato mash was tasty and salmon cone, is like sushi served in a fried roll. From my favorite drink, Sangria to the tapas selection, you just can't go wrong with anything that you choose. We also had a carafe of the sangria which, while tasty, was a bit steep at $38. The Sangria is good and they are known for their Gin and Tonics. Our drinks were great with a good variety of choices by the glass as well as the usual wine list for full bottles. The Sangría de vino tinto (house sangria) was red, full bodied and full of flavor and is now one of my favorites. We ordered a half carafe of Sangria the strawberries and mint werer a good mix and added to the Sangria flavors, this was very good. Terrific wines by the glass and try the sangria for a refreshing summer change. Though a bit pricey, this leading tapas bar has the best sangria in town, with a unique mix that combines flavor and a modest kick. I always enjoy the paired wines that allow new samplings of interesting offerings. Great cocktails and tapas here. We were able to walk right in and be seated, and the restaurant was quite busy. We did not have reservations but the staff was wonderful and seated us right away. We had dinner at Jaleo on a Saturday night and the restaurant was extremely busy, my husband had made a reservation thru Open table so we were quickly seated. This particular night we had a really outstanding experience, because both the food and service were superb. Our drinks were gone about half-way through the meal, but we didn't get to order more until the busser had completely cleared and cleaned our table, and by the time the drinks arrived we had been sitting there, with nothing to do, for over 15 minutes. We were seated near the rear of this restaurant so we able to enjoy our food and talk to each other at the same time - it's a pretty noisy place, so we lucked out. Our waitress was more than happy to have us there the entire time and did not try to get us to leave. I was a little perturbed at first as the choice of tables was not great, the restaurant was full and we had reserved. We had a reservation for three and were seated at a community table with a large noisy party at the other end. We didn't have reservations, so we went to the bar and ate there.

E.1.2 Example 2: Restaurant recommendation with location constraints (Correct answer returned)

Question: '*Hey FT, I'm arriving in San Francisco in August 2010 with my family, so we're starting to plan the trip now. We're quite stoked to visit SF. We would like to ask FT for their recommendations for the following things (need to be in Union Square/Chinatown if possible) - A good Chinese restaurant - Bubbletea shop/cafe - A Vietnamese/pho restaurant - Chinese bakeries - A restaurant that serves wonton noodle :) - Sushi place (can be in Japantown too) Thank you for all of your help!*'

City: San Francisco, USA

Entity Type: Restaurant

Answer returned by Spatio-Textual CsrQA: Slanted door, 1 Ferry building ste 3, San Francisco, CA 94111-4227

Distance from location in question: Approximately 1 mile.

Comment: As can be seen in the review document below is well rated for Vietnamese Food (See text in blue). However, the review document has no reference to its distance or location with respect of China Town, and that is presumably one reason the baseline CsrQAdid not return this answer entity.

Review Document of entity: *The Vietnamese cuisine here is good, not in anyway authentic but nicely adapted. The food is upscale Vietnamese and easy to share. Best Vietnamese food I've ever eaten. Very upscale Vietnamese food with fantastic flavors. I have eaten proper Vietnamese food in Vietnam (and proper Pho is Hanoi) but the stuff here is just ordinary. The Slanted Door serves up very tasty vietnamese cuisine in a vibrant and hip setting. A Vietnamese cuisine fusion menu, which we all found very good indeed. We love Vietnamese food having been introduced to it in the late 1960s. The food is very good and we have never had family style Vietnamese before which was fun. Cocktail selections were good and the drinks well made. Many excellent choices for wine by glass. Cocktails are fantastic as well. Makes the great food, cocktail and wine selections even better! They have interesting cocktails and wines by the glass selected especially to go with the food. The wine list has many expensive wines, but we enjoyed a great malbec for just $28. Bartenders were great the whiskey cocktail was one of the best I've had, and the spring rolls were a hit The cocktails were really good too. There is a nice selection of wines to compliment the menu offerings not just the usual ordinary California wines found in many restaurants. Great drink and wine list. The service was excellent as well. The service was good but not excellent (no complaints). The service was very good as well. The service was pleasant and prompt. Service was very good. Service is another story. The service was very pleasant and not intruding. Service was great. The food was excellent and the waiters/waitresses refreshingly unpretentious (considering the place) and sweet. The food is delicious, view great and service*

friendly. It 's everything you could want in a restaurant, great food, great atmosphere and great staff. The restaurant is fun, the service very good and the food is fantastic. The location is great right off the embarcadero and the food is good. The food is great, Service is wonderful given the fact that the tables and banquets are always full. Don't get me wrong the food is very good - However, with this said, it's still one of my favorite restaurants to visit when in SFO. Overall the food was good, but I wouldn't bother to go there again. I love everything about this place - the food, the location, the service and the vibe. Great food but hundreds of people inside and very noisy. Another visit to the Slanted Door and it was as excellent as ever. Slanted Door is my "go-to" place when visitors are in town. I loved the Slanted Door, it is highly recommended. It was a bit loud but a perfect place for a special dinner out. Slanted Door lived up to its hype. Ambience, though noisy was nice, we had a table overlooking the water. We had a table by the window with a lovely view of the bridge. It's crowded, full and very noisy as all everything is hard: floors, walls, tables, chairs. The Slanted Door was recommend by our local friends. This was my second visit to The Slanted Door and I was not disappointed. The restaurant was full but we were happy to have a table. All the food was delicious, fresh, and perfectly cooked! The food was delicious, very fresh flavours. To top it off, the food was delicious too! This was our second time to dine at the Slanted Door and the second meal was as good as the first. The food was very good, but a little pricey. The food was excellent and served promptly,although not by our waitress. Service and the food was faultless. Rarely saw our waiter after the food was ordered. We had a very wonderful dining experience at this restaurant. Delicious tapas, we got the scallops and they were great. Started with a platter of 1/2 doz oysters served with the sauces lime and horeradish, we had the vegetable rolls with peanut sauces and also the non veg roll. The entire meal was fresh, delicious, and beautifully done. The scallops were perfectly cooked and the flavor was boosted. Had their ceviche appetizer and it was very well balanced between so spiciness and tartness. Lemongrass chicken, shrimp with caramel sauce and the green papaya salad were a few of the great dishes we tried. We had 5 small plates- consisting of 3 types of oysters- really delicious with a nice dipping sauce, tuna tartare, yellowtail and uni- super fresh and prepared perfectly, bbq spareribs, tender, juicy, sauced lovely. Had a starter of an Asian style coleslaw with grapefruit as well as the imperial spring rolls, both of which were excellent, followed by glass noodles with crab and the shaking beef, both of which I'd recommend. We had: Raw oysters from Washington, the best yellowtail EVER with crispy shallots and Thai basil, clams with chilies and pork bellies, and cellophane noodles with dungeness crab. The menu was a little less "out there" than I expected but enjoyed very delicious - straightforward but packed with flavor - dishes such as ahi tuna spring rolls, garlic broccoli, and squid. Great spot to visit, nice location in the Ferry Building. This place is in a perfect setting in the ferry building and in the evening you have a great view of the lit up bridge. Great location on the bay, in the Ferry Building. It's a fun place and very hip, right on the waterfront opposite the Embarcadero

Centre. This has outstanding food and quite the ambience at the embarcadero ferry building. This is a big, bustling place right on the Bay, with a view of the SF-Oakland Bay Bridge. This is a don't miss place in San Francisco with a bonus being getting to see all the food places also located in the Ferry Building. Located in the Ferry Building overlooking the beautiful bay and the bridge. The atmosphere is nice, being in the Ferry Building with beautiful views of the Bay Bridge that now has 25000 twinkling lights. Good location in the ferry building - take a walk afterward for some nice views. All very delicious. well it was OK. This is a special restaurant. very unique. Certainly a place to go back. Thank goodness for that! very trendy Oh yeah, this is what it's all about. It was all excellent! Quite a surprise. We were seated right away, and the wait staff was fun and helpful. We had early (5:30) reservations and when we walked in, we were one of the first to arrive. We arrived to our reservations early on a Wednesday night and were seated promptly. We arrived in the late afternoon so it was busy but the large lunch crowd was over, so we were given a nice large table in the corner which wasn't too loud given the overall bustle of the restaurant. It was a Sunday and we did not have a reservation and we were a group of seven. Reservations were required - so we sat at the bar for an early Dinner. It was very busy but we got lucky and were seated right away. We arrived without reservations but had only a very brief wait in the lounge area before being seated when they opened at 5:30. We had reservations and were immediately seated at a table with a view of the bridge. The place was crowded, but luckily we got a table for six almost immediately.

Answer returned by baseline CsrQA: Ton Kiang 5821 Geary Blvd San Francisco, CA 94121-2004

Distance from location in question: Approximately 4 miles.

Comment: As can be seen in the review document below the entity is considered a good chinese restaurant. However it is not close to China Town and the review document mentions it is not but the reasoner is unable reason using this information. This is example highlights the important of joint-reasoning on different knowledge types.

Review Document of entity: *Everything was very fresh & very good. It was so good in our memories, we plan to try again. It's fresh and delicious. It was great then and it is still great today. I found the dim sum to be good but I know I have had better. It was fabulous, especially the Hakka claypot dishes. Would have loved to try them all, but we were just too full. Ton Kian didn't disappoint with plenty of non-shellfish options. Dim sum has gone downhill over the past few years I had high hopes for Ton Kiang, and while the dim sum was good, the selection was relatively narrow and dishes were pricy. This place is out of the way a bit from downtown or other tourist areas, but quite possibly has some of the best Dim Sum in town. Ton Kiang is one of the best Dim Sum places in San Francisco if not the best. Frommers ratesTon Kiang as the Dim Sum place to visit in San Francisco. We ate here twice during our week visiting San Francisco as it was located right across the street and we were keen to try some dim*

sum without necessarily having to travel into Chinatown. Amazing place with a super selection of food. A must go in San Francisco! We opted to try a place called Ton Kiang which was on the way from our hotel in SOMA to Muir Woods were we planned a hike. Dim sum in San Francisco is probably the best you will get in the United States. Ton Kiang is a great place to take out-of-town guests who want to try authentic dim sum, and you want to take them to a relatively nice looking restaurant. Come here for excellent and varied dim sum at half the price of the downtown options. All the food was very good. The staff was attentive and the food kept coming at a pace that kept us all happy. the food was fresh and tasty. Our party of six dined about 90 minutes here for lunch and ended up spending about $15/person which was a great deal for the variety and quality of food we we were served. This restaurant has an extensive menu and we were very pleased. It was my first visit to this restaurant, but a couple of others in our group had been there before and go there regularly. We were the first in the restaurant and by the time we left it was packed! We didnt wait long to be seated and the dim sum was fantastic. The restaurant was clean and service was polite and prompt. The price seemed reasonable and the food was filling and tasty. it was sad. Not the cheapest. And not delicious. Had to keep asking for stuff. Everything was just average. all delish. All are really terrific. We were delighted. This was NOT one of those times. a bit $$$ but worth it Parking can be difficult in Richmond area. Parking is tricky, but freshness of the food is well worth it. Parking on Geary was difficult in the area. Parking is kind of difficult as well as the line to be seated, but be patient, the food is worth it. Parking around the restaurant can be difficult. Parking however is extremely difficult and finding a parking place can be a very frustrating experience especially during dinner hours. You can't go wrong eating here, but arrive way before you're hungry; parking is difficult and there is a line many times! Parking is attainable in this section of the avenues, so another plus for this restaurant. There is ample parking within 5 blocks of the restaurant. Went on a weekday soon after they opened & got parking right in front of the restaurant. Simply fantastic - best dim sum I've had, and very reasonably priced. My go-to dim sum place in the city. Eating dim sum at Ton Kiang is the the BEST dim sum experience. been a few years since we ate here but it was & still is one of the BEST dim sum & Chinese food in SanFrancisco. Ton Kiang used to be one of the best for dim sum, but recently it has become too crowded and now the food is just average – almost factory-turned-out dim sum with above-average prices. The dim sum at Ton Kiang can be had all day long, but, as somebody who grew up eating dim sum, it seems a bit out of place to eat it beyond lunch time. Nothing fancy, just delicious dim sum with so many choices. I think this is the best dim sum restaurant in San Francisco. We had various dim sum, wonton soups and a couple of stir fries (chicken & cashew and prawns with glazed walnuts). Generally the steamed dumplings were fabulous. I wasn't too thrilled about the dumplings and didn't like the condiment options. Veggie hot and sour soup was OK, as we could not get the regular without ordering a large bowl. For vegetables, they had tender Chinese Broccoli with optional Oyster

Sauce, sautéed green beans and sautéed Pea Shoots, which were outstanding! some of the dim sum we had was: shrimp balls, scallop & shrimp dumplings, siu mai dumplings and shrimp dumplings. I especially liked their steamed roll with pork, but everything that I had was good. Everything we tried was delicious, specially the scallop shrimp dumplings and the shrimp ball. fave dim sum are the asparagus & duck & siu mai/pork dumplings & shrimp balls. Service was horrible. Service was quick and courteous. Service was excellent and fast. Service is fast and efficient. Great service too! And the service was friendly but proper. The prices are reasonable. We ordered a lot, and were satisfied by them. As the place got busier, so did the servers. The service is fast and efficient. Enjoy this bustling restaurant. One of the best Dim Sum!! What a great experience! Great food! Awesome Dim Sum. Great great dim sum! Excellent food. This was a great treat. First to the food. I will definitely go back! It is a typical Chinese dim sum restaurant, not a fancy one but the food is the best. Although there can be a wait at the door, and though it is pricier than other dim sum places, the quality of the food as well as the unusual selection more than makes up for that. A bit pricier and a whole lot cleaner than the dim sum places in chinatown. However, if you are looking for good food, good value, and a place that is frequented by tourists and locals, as well as asians and non-asians, Ton Kiang is the place for you. The restaurant is clean (unlike many other similar establishments) and has a very efficient, friendly service. Service is decent but not A+, but then very very few Chinese restaurants are. This is were normal locals wait in line on weekends to eat excellant although not fancy Dim Sum

E.1.3 Example 3: Hotel recommendation with location constraints and budgetary constraints (Partially correct answer returned)

Question: '*I've visited a few times previously and always stayed around the Midtown area. I'm thinking about another trip in June to celebrate turning 30! Last time I visited I spent a wonderful afternoon in Greenwich Village and would prefer to stay around this area. I would be staying around 4 or 5 nights (depending on price) and need to be in the city on the 17th June. Any recommendations on hotels in the area? Looking to pay around $200 per night including tax.*"

City: New York, USA

Entity Type: Hotel

Answer returned by Spatio-Textual CsrQA: Washington Square Hotel, 103 Waverly Place, Greenwich Village, New York city, NY 10011

Distance from location in question: Approximately 0.6 miles.

Comment: As can be seen in the review document below the entity is close to Greenwich and is generally rated well for its location and cleanliness (See text in blue). However, a

web-search reveals it is not generally available for $200 a night. Our reasoner does not have access to price information.

Review Document of Entity: *"We didn't feel welcomed at all. Had to upgrade There was a wonderful view. Not a great experience. I have stayed here on many occasions. All very good. The staff was very friendly and helpful The staff was very friendly and helpful. The staff were very friendly and helpful. It's a fine hotel - typical old NYC hotel with small guest rooms - but let's be honest - the main thing this hotel has going for it is location. The hotel is very nicely decorated. The hotels location is in a very nice area. Great location and the hotel was clean. Overall, this is a very comfortable hotel in a great location; close to NYU. Hotel is small, old, well-kept and very comfortable. Great location with easy access to the West 4th street station. Great location in Greenwich village right next to Washington Square Park and 3 blocks from Broadway. The Location was great - close to Washington Square Park Love the location with Washington Square and the Village only steps away. Located in Greenwich village just next to Soho is great and as we walk cities, it's nice being located next to a bit of a landmark like the Washington Square Park, as it makes the hotel easy to find. It is in a fantastic location right near Subways and all of Greenwich village yet only a short walk to meat-packing district or the wonderful New Whitney art Museum. Can't wait to come back! Great hotel Good location. Very clean. Excellent location. Price very good! Comfortable room, excellent beds. the prices are a bit high Great room, great location. location is great for my purposes Very good location, convenient to everything. Not what i expected in this price range — felt like a hostel. Great location in the Village! We checked in at 3am after notifying the hotel that that was our check in time. Front desk didn't have any toiletries, had to walk to convenience store nearby (I forgot my razor). All of the front desk staff was friendly and accommodating. front desk staff were incredibly helpful The staff was courteous and helpful and the room was ready a little before 3:00 which worked great for us to check in and drop off our bags. Hotel was very nice and the staff was friendly. Ventilation in the bathroom was very poor. The room was quite small and it was humid even though the air conditioner was on. The room was very drafty and the baseboard heaters did not work so well. The window did not open in the bathroom and it was like a sauna after each shower Bathroom was very clean and the shower was excellent. We had an ensuite shower and bath. There was an old fashioned radiator in the room that was turned all the way up, but the room was still pretty cold. The room was comfortable, cozy and clean. Room was cozy and clean. The room was small, but it was clean and the bed was comfortable. Bed was comfortable, and the room was very clean. Bed was very comfy and room was cozy. The beds were very comfortable and the room was clean. Room was a little small. Included breakfast was very small. Breakfast was very nice, great location Breakfast cafe and breakfast were disappointing. The complimentary breakfast was poor. The complimentary breakfast was also very good. Breakfast Pastries were good but not excellent Continental breakfast was included, but we opted for the hot breakfast at*

an extra charge.

Answer returned by baseline CsrQA: Radio City Apartments, 142 West 49th Street, New York City, NY 10019

Distance from location in question: Approximately 3 miles.

Comment: As can be seen in the review document below the entity is generally rated well for its location (See text in blue), but it does not fulfill a key location constraint about being close to Greenwich. The CsrQAsystem however selects a hotel in Mid-Town Manhattan presumably because of its mention in the question and because the review mentions it.

Review Document of entity: *Rooms clean, plenty of towels tea & coffee replaced every day. Kitchen had microwave, coffee pot, full Coker pots pans etc, washing up liquid, sponge, t thowel. The hotel doesn't provide food but the kitchenette had a kettle with brewing coffee/tea, a microwave and top hob for cooking. The bathroom was very small but adequate as was the kitchenette which was OK for making coffee and had a fridge freezer for drinks. It was spotless - washing up liquid, kitchen roll etc all provided too. The room we had was bare bones – every amenity was the cheapest any hotel could find from pots and pans, cutlery, towels and mattresses. very good location. Excellent location! Excellent location. Perfect location. Very conviniente! Location, location, location! Location - Location - Location! Location, location, location. LOcation, location, location. Location is everything! The location is just perfect! Location is excellent. Staff were very friendly. The staff were extremely friendly and helpful. Staff was very friendly. Staff was welcoming and helpful. Staff were extremely polite and helpful. Location can't be beat. The staff was also friendly and helpful. It was quite noisy. It was very small. It was horrible! It was WONDERFUL!! It looked much more updated. We stayed one night. Had everything we wanted. The location is really great - walking distance from most of the main tourist attractions. The location is excellent, close the best parts of the city, staff very gentil and helpful Note to potential visitors, there is a front desk that is manned 24/7 and a lovely Italian restaurant next door that is connected through the lobby. Very nice rooms and staff-not all the perks of the higher end hotels, but great space with kitchen included! Very spacious rooms, amazing location, staff was very friendly. Location was good, easy walk to the places we wanted to explore. The location was perfect and though not a glamorous hotel it was perfect for our stay and I would go there again. Hotel is very clean, brilliant location and the staff are very helpful. Front desk and housekeeping staff were really friendly and had no issues with extending my stay by a night, or storing my suitcase after check out until my train departure in the afternoon. We've been there 6 nights and it was absolutely O. K. After check out we left our luggage in the hotel and returned 4 hours later. I asked if it was possible to extend our stay in hotel for another night because our flight was cancelled and they've answered me there was a single room for 424$ without taxes. They booked a Limo without our permission although we*

asked them to book a yellow cab. Since we didn't drive, they allowed us to leave our luggage there until we were scheduled to catch our bus back home. I immediately told the driver that we ordered a taxi and not a private service and that I was upset that the STAFF AT RADIO CITY APARTMENTS ARE LYING TO THEIR CUSTOMERS. The limo company may have a contract with the hotel, but they should say so when we asked to book a "taxi. " The staff on the front desk were awful - Rude and unhelpful - so sad when a smile costs nothing. We got upgraded to the PentHouse room, very spacious and with a large terrace! Room was spacious and clean with good kitchen facilities. The rooms were clean and comfortable for my family The apartments are old and our room was quite small (but we are out all day and only sleep in room). The rooms had a musty smell 2 bedroom aprrtment - all rooms were small but comfortable and clean. Bed was comfortable and the apartment was clean and spacious. It is very convenient to all the Midtown attractions; subway close by as well. Great location - close to time square and subway station from airport. Location is excellent, next to Times Square, easy to subways stations, surrounding by broadway theatres. Brilliant location in the heart of Manhattan, close to Times Square and Top of the Rock. Great location but anything near Times Square is noisy at night. Close to Times Square and about a 15 minute walk to Central Park. Close to Times Square, Broadway and Subway. Our room had a small window a/c that was loud and obnoxious, and didn't cool well. No air conditioning so the room got very hot. The air conditioner was a bit loud but the room was cool. The noise of the A/C (window unit) was terrible. Rooms heated by radiators which were very noisy at night. We could hear all the noise from the street (being centrally located, that is a loooot of noise) so it was difficult to stay asleep throughout the night. We could open the window but it was noisy at night from the street below.

E.1.4 Example 4: Hotel Recommendation (Correct answer returned)

Question: '*Hi , am planning a trip to moscow & st pet and want to stay at the most centrally located hotels in both. I want to be right in the shopping and tourist sites and love hustle & bustle. Has anyone been to the new ritz in moscow? is it closer to the action than the marriott aurora ? and in st petes does anyone know best cetral hotel. It want a hotel where everything as at the doorstep... thansk sooo much...*'

City: Moscow, Russia

Entity Type: Hotel

Answer returned by Spatio-Textual CsrQA: Savoy, Rozhdestvenka Str.3/6 Bld 1, Meshchansky, 109012 Moscow

Comment: This is an example of a question that implicitly requires location reasoning but our spatio-textual model cannot do a multi-hop inference for location reasoning us-

ing"*shopping and tourist sites*"; i.e., a model would first need to find locations that are shopping and tourist sites and then find hotels close to them. However, the review mentions that it has a good location close to tourist sites and presumably why both reasoners return this correct answer entity.

Review Document of entity: *Staff were friendly and helpful staff are very friendly. The staff was extremely friendly and kind. Reception and restaurant staff were very kind and helpful. The staff was very friendly. The staff are extremely helpful and courteous. not all the concierge staff speak English Location is perfect. Location is perfect Location is very good. Location is very good Location is perfect, with many bars/restaurants within blocks. Location is excellent Excellent hotel in the very centre of Moscow! A very good location with access to the city centre and the underground. This is a lovely hotel in a perfect spot: it's close to the main attractions (walking distance to the Red Square and the Bolshoi Threatre) and in the middle of a lively neighbourhood with tons of shops, botiques and all sort of restaurants. Stylish Hotel - 3rd visit this year - great place in central location Lovely room and a great location for restaurants, shopping and sight-seeing. A stylish, well managed classic hotel, close to all sights in the centre of Moscow. Perfect location, nice refurbishment of historic surroundings, good value for money, larger than usual rooms, friendly staff who speak English. just a great hotel Very clean. Great staff. Great staff Perfect location. Good location. The beds were super comfortable and the rooms were clean. The hotel room was spacious, comfortable and bright. The rooms were very nice and cosy! Rooms felt luxorious. My room was quiet, spacious and comfortable Rooms were amazing - spacious and quiet- Junior suite with the balcony was just the best! I requested a quiet room but the room 410 was beween a service room and the stuff stairwell. Had to call room service (which was fast however). I cancelled the extra night last night (booked elsewhere with Booking. Tried to speak to reception to either change the room or cancel with one night charge - they charged me 100% for the whole stay even that I check out next morning.Tried to speak to booking. I booked a taxi with the hotel to come to pick me up at the train station. We stayed for 3 nights and had a booking for an extra night (2 nights later). The breakfast we ordered through room service was not the best and overpriced Swimming pool and sauna excellent too. Pool & sauna when not closed Sauna and pool pleasent. The pool and sauna were a pleasant bonus. Sauna, Pool and Gym inculuded in the room. Shower inside the tub I like gym and swimming pool very much. Super nice well sized bathroom with shower, huge bath and bidet. Bar prices. bar was a little inattentive. Excellent location in the Center. The bar is gorgeous too. Heritage building. Quite pricey for a small hotel. Beautiful luxury restaurant. Breakfast was served until 11am which was a great bonus. Breakfast is very good. Great buffet breakfast and dining choices. The breakfast buffet was extensive including a freshly made-to-order omelette if your choice. Great breakfast buffet 5. Breakfast was nice but a lot of the food was cold Breakfast place is gorgerous. Everything was great. Wouldn't out me off staying again though . Wonderful stay*

here. there is nothing not to like!!!!! Everything was perfect Everything was perfect! Nothing to mention

Answer returned by baseline CsrQA: Savoy, Rozhdestvenka Str.3/6 Bld 1, Meshchansky, 109012 Moscow

Review Document of entity: Same as shown previously.

E.1.5 Example 5: Restaurant Recommendation (Incorrect answer returned)

Question: '*Hi all,yes I know it's a bad habit... But are there still restaurants in Bangkok where smoking cigarettes is allowed? I think that inside all air-conditioned restaurants it00b4s forbidden, but could you maybe give some advice, which restaurants still have smoking sections, maybe cigar bars or outdoor dining where one could smoke? Thank you in advance for your answers!Best regards,sanook:)*'

City: Bangkok, Thailand

Entity Type: Restaurant

Answer returned by Spatio-Textual CsrQA: Supatra River House 266 Soi Wat Rakhang, Arunamarin Road Bangkok

Comment: As can be seen in the review document does not contain any information regarding smoking and yet it is returned as an answer by both systems. This is perhaps an example of inadequate reasoning in our models.

Review Document of entity: *Its easy to get there by BTS Chong nonsi station, exit 3 heading to big junction (Narathivas Junction) cross the street then turn left, looking for first sub-street (you'll see the Indian restaurant at the corner) turn right, go straight and there is the restaurant. don't expect much service because people go there for simple , tasty comfort food and good price. They offer Sichuan food which was a bit spicy. I enjoy coming here for Chinese food because they have quite a friendly price . One of my favourite Chinese restaurants in town. However if once in a while you want to have really good MALA aka szechuan peppercorn hotpot this is the place. The bathroom is a bit icky and service is Chinese style, they wont smile but they will serve you. A bit crowded on weekend. Its alway busy in evening and weekend. Highly recommend for Chinese food lovers. The food was really good and tasty that seems like we are eating in China!! Their stir fried veggies is a good mix of garlic and not oily. The favorite dishes are half chicken, fried garlic fox, spicy bean with pork. They offer goat, beef, pork and chicken, fish and a lot of veggies that you can't get at other hotpot restaurants like water cress. It's delicious. A bit hard to find but worth the visit. Great taste. It was very delicious! Best no nonsense hidden gem. I'm really enjoy all the dishes. I heard many good things about this place so I finally decided to experience it myself. Initially, we waited to be called after informing the*

staff. The food is good and fresh. Its a great family restaurant and still affordable. The food is delicious - If you are allergic to MSG don't come here. We tried 5-6 dishes including a whole boiled chicken (a huge portion, delicious), an excellent spicy fried tofu and fried kidney; the only dish that wasn't worth it was the vegetable soup, which was too salty. I recommend "Tearing Chicken", it's a boiled seasoned chicken, it's the most popular dish here. all great home cooking, a lot of deep fried, bit oily yet satisfying with rice! The fried tofu, eggplant dish and steamed eggs are comforting typical homemade food. You can ask for extra garlic and chilies to add to your soy sauce sesame oil sauce which they serve with coriander and spring onions. Steamed minced pork with salted fish, deep fried squid with chilli, deepfried shrimp with garlic, fried long beans with mince pork and chilli, steamed eggplant in claypot with salted fish, mapo tofu. They have the cold salted chicken here and stir fried long string beans. The price are reasonable and you will feel worthy. Great price. The staff then proceeded to issue queue number tags as well as handed out the menu for order-taking. The queue seemed disorganized as a group who arrived later than us was admitted before us. The restaurant was busy when we arrived for dinner. All dishes were good and considered cheap. Their fried squid is lightly battered, doused with peppers, salts, garlic and chilies. even the atmosphere is not really fantastic but it can compensate with the good taste of the food.

Answer returned by baseline CsrQA: Supatra River House 266 Soi Wat Rakhang, Arunamarin Road Bangkok

Review Document of entity: Same as shown previously.

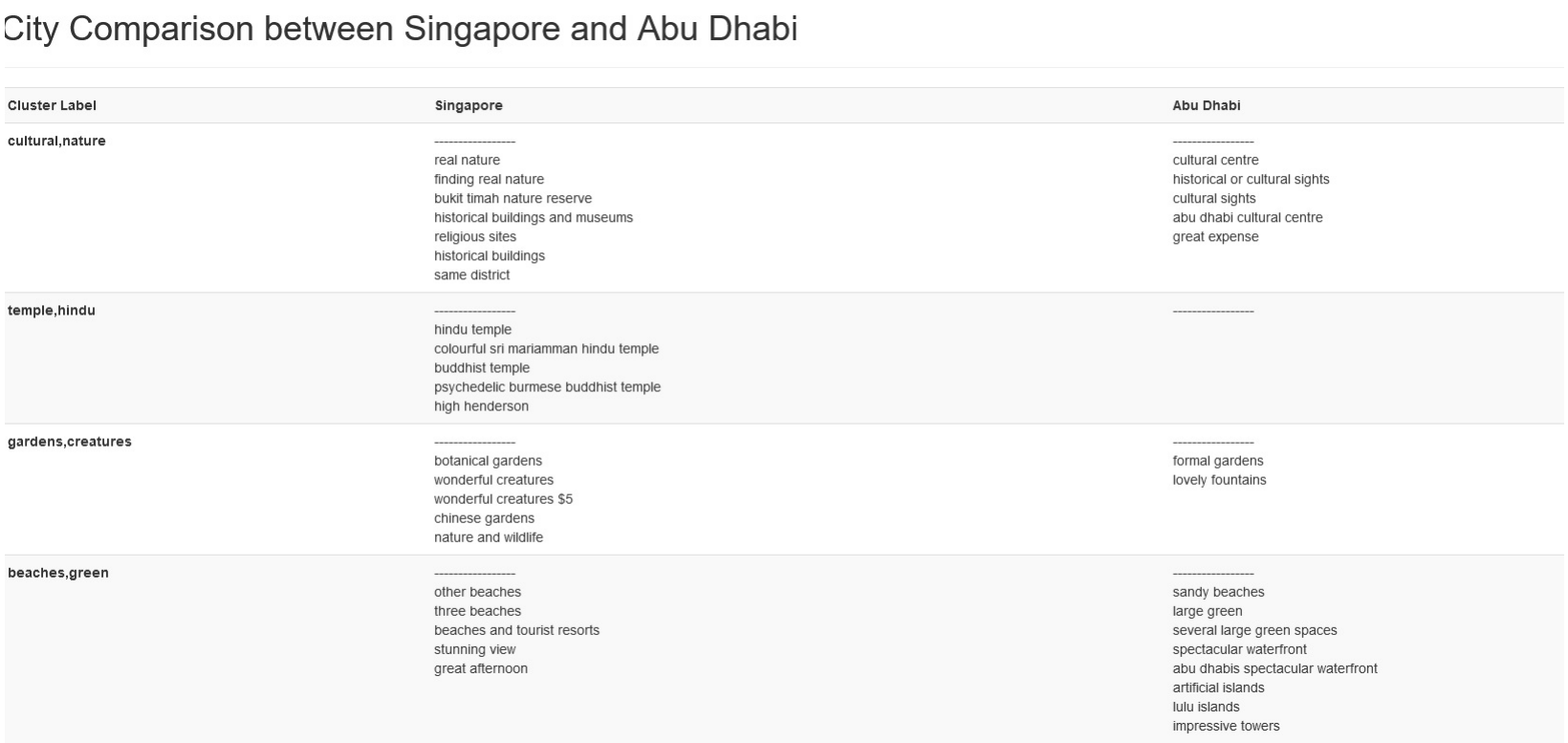
E.2 Comparison Questions

We now show examples from our system (based on EB G-pLSA) for comparing cities. Figures E.1, E.2, E.3 compare Singapore with three different cities – Philadelphia, Abu Dhabi and Kuala Lumpur respectively. Figures E.4, E.5, E.6 compare Rome with Goa, Jerusalem and Hyderabad respectively.

City Comparison between Singapore and Philadelphia

Cluster Label	Singapore	Philadelphia
park,historic	---------- beaches and tourist resorts three beaches	---------- historic park fairmount park fdr park pennypack park independence national historic park philadelphias signature historic site thriving park wissahickon valley park fairmount park technically skate park penn treaty park first zoo kimmel center
temple,hindu	---------- hindu temple colourful sri mariamman hindu temple buddhist temple psychedelic burmese buddhist temple chinese gardens	---------- jewish history japanese teahouse
art,public	---------- nature and wildlife real nature	---------- public art other public art great public art philadelphias art art museum area interpretive art public transportation please touch museum several sculptures free library thousand animals and wildlife various animals largest collection unique pieces breeding and housing various animals liberty place civil war

Figure E.1: Comparison generated between two cities - Singapore and Philadelphia. Truncated for ease of presentation.



City Comparison between Singapore and Abu Dhabi

Cluster Label	Singapore	Abu Dhabi
cultural,nature	---------- real nature finding real nature bukit timah nature reserve historical buildings and museums religious sites historical buildings same district	---------- cultural centre historical or cultural sights cultural sights abu dhabi cultural centre great expense
temple,hindu	---------- hindu temple colourful sri mariamman hindu temple buddhist temple psychedelic burmese buddhist temple high henderson	----------
gardens,creatures	---------- botanical gardens wonderful creatures wonderful creatures $5 chinese gardens nature and wildlife	---------- formal gardens lovely fountains
beaches,green	---------- other beaches three beaches beaches and tourist resorts stunning view great afternoon	---------- sandy beaches large green several large green spaces spectacular waterfront abu dhabis spectacular waterfront artificial islands lulu islands impressive towers

Figure E.2: Comparison generated between two cities - Singapore and Abu Dhabi. Truncated for ease of presentation. Notice the different topical organization; in contrast to the previous example, there is a finer cluster around beaches.

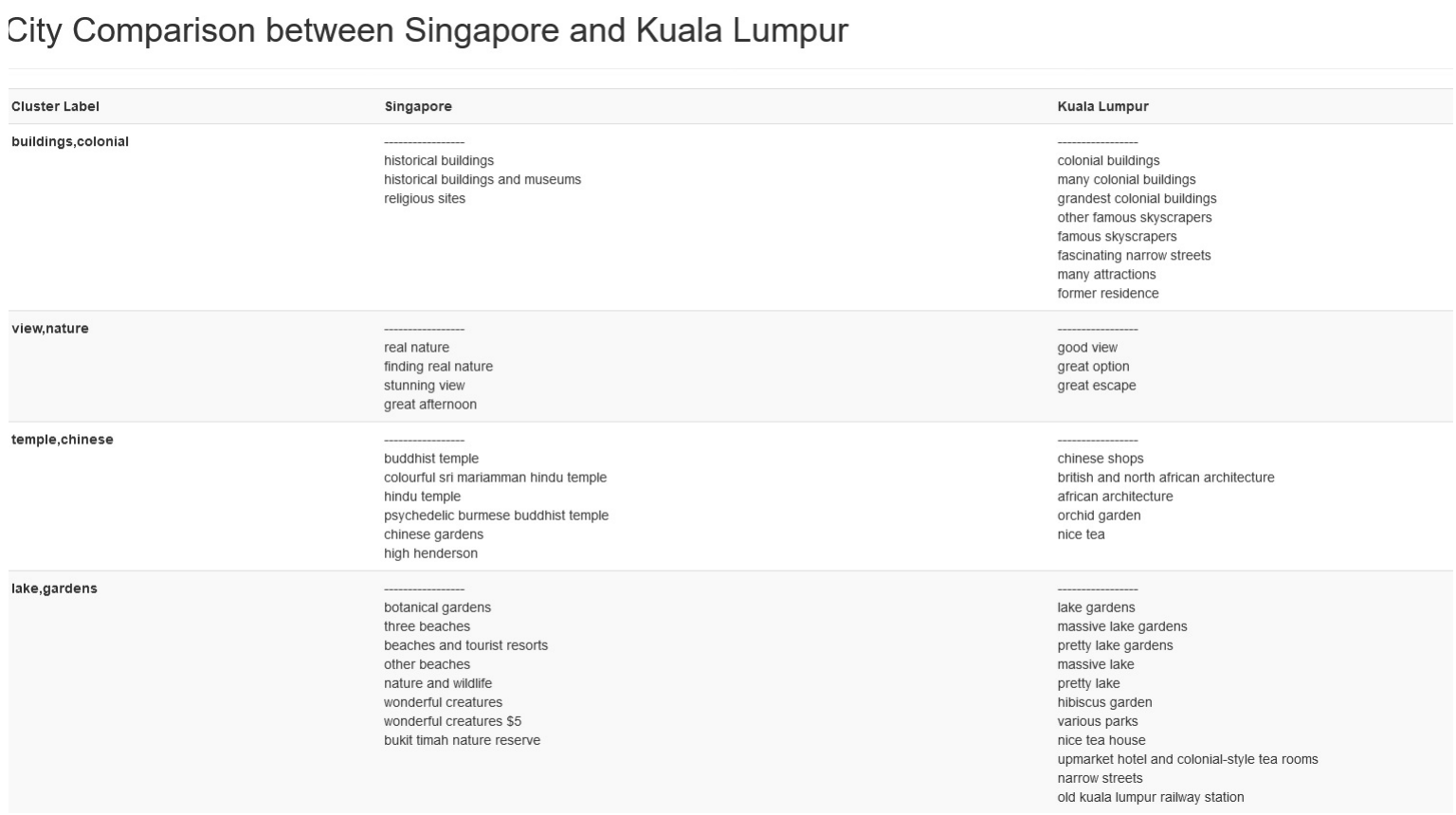


Figure E.3: Comparison generated between two cities - Singapore and Kuala Lumpur. Truncated for ease of presentation. Notice the different topical organization; in contrast to the previous example, there is a finer cluster around colonial buildings and there is a comparative cluster for beaches and parks.

City Comparison between Rome and Goa

Cluster Label	Rome	Goa
art,museum	----------	----------
	art museums	christian art museum
	etruscan art	art galleries and museums
	national museum	subodh kerkars art gallery
	romes national museum	modern art gallery
	romes wax museum	architectural museum
	childrens museum	art galleries and libraries
	excellent astronomy museum	modern art
	classical art	religious museum
	baroque art	goa state museum
	private art	goa chitra museum
	musical instrument museum	religious art
	antique roman and greek art and sculptures	secular art
	vatican museum	gerard da cunhas architectural museum houses
	museum opening hours	naval aviation museum
	fine classical art	christian art
	previously private art collection	christian religious art
	worlds largest etruscan art collection	many artists painters and architects
	greek art	many artists
	italian renaissance and baroque art	
	roman sculptures	
	renaissance and medieval buildings	
	renamed rome zoo	
	excellent astronomy	
	hands-on science	
collection,archaeological	----------	----------
	archaeological collection	vintage-cars collection
	vast archaeological collection	ethnographic artifacts
	archaeological site	historical research
	archaeological site and museum	largest collection
	archeological park	chapora fort
	egyptian collection	fort offers
	important collection	
	enormous collection	
	most important collection	
	stunning egyptian collection	
	ancient forum	
	ancient forums	
	small medieval village	
	unique elephant	
	web site	
	ivth century	
	small bits	
	sixth century	
	fourth century	
beach,beaches	----------	----------
	colle oppio park	candolim and sinquerim beaches
	quirinal hills	vagator and anjuna beaches
	underneath park	arambol beach
	other hills	colva beach
	seven hills	patnem beach
	exotic birds	velsao beach
	amazing roof gardens	southern beaches
	original seven hills	popular beaches
	sweeping views	beautiful beach
	amazing roof	other popular beaches
		turtle beach
		wide beach
		very wide beach
		salcettes beach stretch starts
		nearby islands
		concretised coast
		scenic beauty
		natural scenic beauty
		unique rain water
		magnificent albuquerque mansion
		unique rain

Figure E.4: Comparison generated between Rome and Goa. Truncated for ease of presentation. Amongst others, notice the cluster related to beaches and parks.

City Comparison between Rome and Jerusalem

Cluster Label	Rome	Jerusalem
jewish,islamic	----------	----------
	vatican area	temple mount jewish / muslim
	second world war rome / esquilino-san giovanni	jewish art
	rome or others	islamic building
	sixth century bc	islamic art
	fourth century bc	jewish community
	third century ad	jewish sites
	two most popular views	jewish temple
	most important collection	islamic architecture
	very popular spot	most holy jewish sites
		muslim sites
		most amazing islamic building
		christian spot
		most holy christian spot
		arab village
		abandoned arab village
		holies site
		ancient ritual
		1967 arab-israeli war
		arab-israeli war
		ritual purification
		non-recreational ritual
		popular tourist sites
		most important site
		third most important site
hills,gardens	----------	----------
	quirinal hills	jerusalem botanical gardens
	fountains and gardens	botanical gardens
	many hills	lush big garden
	seven hills	big garden
	separate hills	beautiful cluster
	amazing roof gardens	spacious multilevel half-ruined buildings
	original seven hills	spectacular night-show
	more than seven hills	amazing array
	underneath park	
	lovely bernini	
	exotic birds	
	amazing roof	
	major attractions	
	short walk	
	sweeping views	
	good views	
	good climb	
	excellent views	
	unique elephant	
	great place	
	excellent view	
	interesting place	
	few good reports	
	good reports	
church,century	----------	----------
	ivth century	15th century
	third century	domini flevit church
	fourth century	maronite church
	sixth century	noster church
	churches and houses	lutheran church
	medieval village	first church
	medieval buildings	armenian hospice syriac church
	renaissance and medieval buildings	armenian cathedral and museum murestan square
	small medieval village	armenian cathedral
	baroque art	garden tomb
	magnificent square	temple mount plaza
	imperial rome	temple mount
	roman forum	jerusalem city hall
	roman sculptures	magnificent dome
	tall buildings	kidron valley monuments
	stunning egyptian collection	half-ruined buildings
	few pieces	
	colle oppio park	
	trajans forum	
	egyptian collection	

Figure E.5: Comparison generated between two cities - Rome and Jerusalem. Truncated for ease of presentation. In contrast to the previous comparison, notice clusters related to Islamic and Jewish art emerge.

City Comparison between Rome and Hyderabad

Cluster Label	Rome	Hyderabad
fountain,pond	----------	----------
	trevi fountain	little pond
	central fountain	polluted little pond
	famous fountain	artificial lake
	fountain or two	bagh-e-aam garden
	fascinating fountain	botanical garden
	fountains and gardens	
	interesting fountains	
	amazing roof gardens	
tec,hi-tec	----------	----------
	vatican area	tec city
	rome municipality	hi tec city
	relevant district	hyderabad city
	pincio metro line	hyderabad public school
	main area	imposing state legislative assembly building
	small square	nighttime and large pedastrian space
	very well-preserved city wall	busy city
	large area	entire village
	larger area	walkable places
	capital city	hi-tec city area
	well-preserved city	few walkable places
	old city	hi-tec city
	city boundaries	historical region
		old city
		public school
streets,hills	----------	----------
	other hills	tiny streets
	many hills	main hall
	cobbled streets	high court
	side streets	
	narrow streets	
	many street	
	side street	
	more than seven hills	
	many street entertainers	
	red line	
	huge line	
houses,shops	----------	----------
	small shops	most shops
	churches and houses	beautiful scruffy old shops and houses
	small houses	tiny streets and shops
	old houses	old shops
	galleria nazionale darte moderna houses	beautiful buildings
	tall buildings	industrial house
	small medieval village	single-storeyed building
	many old palaces	crafts village
	old palaces	esteem house
	small settlements	nqqar khana place
	small bits	beautiful place
	separate hills	beautiful skyview
	quirinal hills	old grandmother
	original seven hills	
	seven hills	
	very popular spot	
	popular spot	
	old appian way	
	old appian	
museum,art	----------	----------
	art museums	modern art museum
	etruscan art	museum 10 30-20 00
	national museum	rock museum
	romes national museum	archaeological museum
	romes wax museum	modern art
	childrens museum	health museum
	excellent astronomy museum	art work
	capitoline museums	technological museum
	world-class museums	state archaeological museum

Figure E.6: Comparison generated between two cities - Rome and Hyderabad. Truncated for ease of presentation. Notice the first cluster related to water bodies and the cluster related to temples and roman architecture that emerges in the comparison.

Bibliography

A. Abujabal, M. Yahya, M. Riedewald, and G. Weikum. Automated template generation for question answering over knowledge graphs. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pages 1191–1200, 2017. doi: 10.1145/3038912.3052583. URL https://doi.org/10.1145/3038912.3052583.

A. Abujabal, R. Saha Roy, M. Yahya, and G. Weikum. ComQA: A community-sourced dataset for complex factoid question answering with paraphrase clusters. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 307–317, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1027. URL https://www.aclweb.org/anthology/N19-1027.

H. Akoglu. User's guide to correlation coefficients. *Turkish Journal of Emergency Medicine*, 18:91 – 93, 2018.

A. Amini, S. Gabriel, S. Lin, R. Koncel-Kedziorski, Y. Choi, and H. Hajishirzi. MathQA: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2357–2367, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1245. URL https://www.aclweb.org/anthology/N19-1245.

D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1409.0473.

C. F. Baker, C. J. Fillmore, and J. B. Lowe. The berkeley framenet project. In *Pro-*

ceedings of the 17th International Conference on Computational Linguistics - Volume 1, COLING '98, pages 86–90, Stroudsburg, PA, USA, 1998. Association for Computational Linguistics. doi: 10.3115/980451.980860. URL https://doi.org/10.3115/980451.980860.

A. Banerjee and J. Ghosh. Scalable clustering algorithms with balancing constraints. *Data Min. Knowl. Discov.*, 13(3):365–395, Nov. 2006. ISSN 1384-5810. doi: 10.1007/s10618-006-0040-z. URL http://dx.doi.org/10.1007/s10618-006-0040-z.

M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni. Open information extraction from the web. In *IN IJCAI*, pages 2670–2676, 2007.

A. Bapna, G. Tur, D. Hakkani-Tur, and L. Heck. Towards zero shot frame semantic parsing for domain scaling. In *Interspeech 2017*, 2017.

F. Basik, B. Hättasch, A. Ilkhechi, A. Usta, S. Ramaswamy, P. Utama, N. Weir, C. Binnig, and U. Cetintemel. Dbpal: A learned nl-interface for databases. In *Proceedings of the 2018 International Conference on Management of Data*, pages 1765–1768, 2018.

J. Berant and P. Liang. Semantic parsing via paraphrasing. In *Association for Computational Linguistics (ACL)*, 2014.

J. Berant, A. Chou, R. Frostig, and P. Liang. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1533–1544, 2013. URL http://aclweb.org/anthology/D/D13/D13-1160.pdf.

B. Bi, C. Wu, M. Yan, W. Wang, J. Xia, and C. Li. Incorporating external knowledge into machine reading for generative question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2521–2530, 2019. doi: 10.18653/v1/D19-1255. URL https://doi.org/10.18653/v1/D19-1255.

D. M. Blei, A. Y. Ng, M. I. Jordan, and J. Lafferty. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:2003, 2003.

D. Bogdanova and J. Foster. This is how we do it: Answer reranking for open-domain how questions with paragraph vectors and minimal feature engineering. In *NAACL HLT*

2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016, pages 1290–1295, 2016. URL http://aclweb.org/anthology/N/N16/N16-1154.pdf.

K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-102-6. doi: http://doi.acm.org/10.1145/1376616.1376746. URL http://portal.acm.org/citation.cfm?id=1376746#.

A. Bordes, S. Chopra, and J. Weston. Question answering with subgraph embeddings. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 615–620, 2014a. URL http://aclweb.org/anthology/D/D14/D14-1067.pdf.

A. Bordes, J. Weston, and N. Usunier. Open question answering with weakly supervised embedding models. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part I*, pages 165–180, 2014b. doi: 10.1007/978-3-662-44848-9_11. URL http://dx.doi.org/10.1007/978-3-662-44848-9_11.

A. Bordes, N. Usunier, S. Chopra, and J. Weston. Large-scale simple question answering with memory networks. *arXiv preprint arXiv:1506.02075*, 2015.

L. Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075. URL https://www.aclweb.org/anthology/D15-1075.

V. Castelli, R. Chakravarti, S. Dana, A. Ferritto, R. Florian, M. Franz, D. Garg, D. Khandelwal, J. S. McCarley, M. McCawley, M. Nasr, L. Pan, C. Pendus, J. F. Pitrelli, S. Pujar, S. Roukos, A. Sakrajda, A. Sil, R. Uceda-Sosa, T. Ward, and R. Zhang. The techqa dataset. *CoRR*, abs/1911.02984, 2019. URL http://arxiv.org/abs/1911.02984.

D. Cer, Y. Yang, S. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, Y. Sung, B. Strope, and R. Kurzweil. Universal sentence encoder. *CoRR*, abs/1803.11175, 2018. URL http://arxiv.org/abs/1803.11175.

M. Chang, L. Ratinov, and D. Roth. Guiding semi-supervision with constraint-driven learning. In *In Proc. of the Annual Meeting of the ACL*, 2007.

D. Chen, A. Fisch, J. Weston, and A. Bordes. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada, July 2017a. Association for Computational Linguistics. doi: 10.18653/v1/P17-1171. URL https://www.aclweb.org/anthology/P17-1171.

D. Chen, A. Fisch, J. Weston, and A. Bordes. Reading Wikipedia to answer open-domain questions. In *Association for Computational Linguistics (ACL)*, 2017b.

L. Chen, J. M. Jose, H. Yu, F. Yuan, and D. Zhang. A semantic graph based topic model for question retrieval in community question answering. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, WSDM '16, pages 287–296, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3716-8. doi: 10.1145/2835776.2835809. URL http://doi.acm.org/10.1145/2835776.2835809.

M. Chen, M. D'Arcy, A. Liu, J. Fernandez, and D. Downey. Codah: An adversarially-authored question answering dataset for common sense. In *Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for NLP*, pages 63–69, 2019.

J. Cheng, L. Dong, and M. Lapata. Long short-term memory-networks for machine reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561, Austin, Texas, Nov. 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1053. URL https://www.aclweb.org/anthology/D16-1053.

J. Cheng, S. Reddy, V. Saraswat, and M. Lapata. Learning structured natural language representations for semantic parsing. *arXiv preprint arXiv:1704.08387*, 2017.

K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1179. URL https://www.aclweb.org/anthology/D14-1179.

E. Choi, H. He, M. Iyyer, M. Yatskar, W. Yih, Y. Choi, P. Liang, and L. Zettlemoyer. Quac: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2174–2184, 2018. URL https://aclanthology.info/papers/D18-1241/d18-1241.

J. Christensen, S. Soderland, G. Bansal, and Mausam. Hierarchical summarization: Scaling up multi-document summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pages 902–912, 2014. URL http://aclweb.org/anthology/P/P14/P14-1085.pdf.

C. Clark, K. Lee, M.-W. Chang, T. Kwiatkowski, M. Collins, and K. Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *NAACL*, 2019.

P. Clark, O. Etzioni, T. Khot, A. Sabharwal, O. Tafjord, P. Turney, and D. Khashabi. Combining retrieval, statistics, and inference to answer elementary science questions. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, pages 2580–2586. AAAI Press, 2016. URL http://dl.acm.org/citation.cfm?id=3016100.3016262.

R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12: 2493–2537, 2011.

G. Cong, C. S. Jensen, and D. Wu. Efficient retrieval of the top-k most relevant spatial web objects. *Proc. VLDB Endow.*, 2(1):337–348, Aug. 2009. ISSN 2150-8097. doi: 10.14778/1687627.1687666. URL https://doi.org/10.14778/1687627.1687666.

D. Contractor, Mausam, and P. Singla. Entity-balanced gaussian plsa for automated comparison. In *Proceedings of NAACL-HLT*, pages 69–79, 2016.

D. Contractor, K. Shah, A. Partap, Mausam, and P. Singla. Large scale question answering using tourism data. *CoRR*, abs/1909.03527, 2019. URL http://arxiv.org/abs/1909.03527.

D. Contractor, B. Patra, Mausam, and P. Singla. Constrained bert bilstm crf for understanding multi-sentence entity-seeking questions. *Natural Language Engineering*, page 1–23, 2020. doi: 10.1017/S1351324920000017.

J. R. Curran. *From Distributional to Semantic Similarity*. PhD thesis, Institute for Communicating and Collaborative Systems School of Informatics University of Edinburgh, 2003. URL http://www.era.lib.ed.ac.uk/bitstream/1842/563/2/IP030023.pdf.

Z. Dai, X. Wang, P. Ni, Y. Li, G. Li, and X. Bai. Named entity recognition using bert bilstm crf for chinese electronic health records. In *2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–5, 2019. doi: 10.1109/CISP-BMEI48845.2019.8965823.

M. Daniluk, T. Rocktäschel, J. Welbl, and S. Riedel. Frustratingly short attention spans in neural language modeling. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017. URL https://openreview.net/forum?id=ByIAPUcee.

R. Das, M. Zaheer, and C. Dyer. Gaussian lda for topic models with word embeddings. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 795–804, Beijing, China, July 2015. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P15-1077.

R. Das, M. Zaheer, S. Reddy, and A. McCallum. Question answering on knowledge bases and text using universal schema and memory networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 358–365, 2017. doi: 10.18653/v1/P17-2057. URL https://doi.org/10.18653/v1/P17-2057.

G. De Rassenfosse, J. Kozak, and F. Seliger. Geocoding of worldwide patent data. *Scientific data*, 6(1):1–15, 2019.

K. Deb and K. Deb. *Multi-objective Optimization*, pages 403–449. Springer US, Boston, MA, 2014. ISBN 978-1-4614-6940-7. doi: 10.1007/978-1-4614-6940-7_15. URL https://doi.org/10.1007/978-1-4614-6940-7_15.

P. Deepak, D. Garg, and S. Shevade. Latent space embedding for retrieval in question-answer archives. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 855–865, 2017.

S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, 41(6):391–407, 1990.

J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019. URL https://aclweb.org/anthology/papers/N/N19/N19-1423/.

D. Dua, Y. Wang, P. Dasigi, G. Stanovsky, S. Singh, and M. Gardner. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proc. of NAACL*, 2019.

M. Dunn, L. Sagun, M. Higgins, V. U. Güney, V. Cirik, and K. Cho. Searchqa: A new q&a dataset augmented with context from a search engine. *CoRR*, abs/1704.05179, 2017. URL http://arxiv.org/abs/1704.05179.

J. Eisner. High-level explanation of variational inference. 2011. URL http://www.cs.jhu.edu/~jason/tutorials/variational.html.

L. El Asri, H. Schulz, S. Sharma, J. Zumer, J. Harris, E. Fine, R. Mehrotra, and K. Suleman. Frames: a corpus for adding memory to goal-oriented dialogue systems. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 207–219, Saarbrücken, Germany, Aug. 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-5526. URL https://www.aclweb.org/anthology/W17-5526.

M. Eric, R. Goel, S. Paul, A. Sethi, S. Agarwal, S. Gao, A. Kumar, A. Goyal, P. Ku, and D. Hakkani-Tur. Multiwoz 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 422–428, 2020.

O. Etzioni, A. Fader, J. Christensen, S. Soderland, and Mausam. Open Information Extraction: the Second Generation. In *International Joint Conference on Artificial Intelligence (IJCAI)*, Barcelona, Spain, July 2011.

A. Fader, L. S. Zettlemoyer, and O. Etzioni. Paraphrase-driven learning for open question answering. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*, pages 1608–1618, 2013. URL http://aclweb.org/anthology/P/P13/P13-1158.pdf.

A. Fader, L. Zettlemoyer, and O. Etzioni. Open question answering over curated and extracted knowledge bases. In *Proceedings of the 20th ACM SIGKDD International*

Conference on Knowledge Discovery and Data Mining, KDD '14, pages 1156–1165, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2956-9. doi: 10.1145/2623330.2623677. URL http://doi.acm.org/10.1145/2623330.2623677.

A. Fan, Y. Jernite, E. Perez, D. Grangier, J. Weston, and M. Auli. ELI5: long form question answering. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3558–3567, 2019. URL https://www.aclweb.org/anthology/P19-1346/.

D. Ferrés Domènech. Knowledge-based and data-driven approaches for geographical information access. 2017.

J. R. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 363–370, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. doi: 10.3115/1219840.1219885. URL https://www.aclweb.org/anthology/P05-1045.

A. Fisch, A. Talmor, R. Jia, M. Seo, E. Choi, and D. Chen. MRQA 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 1–13, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-5801. URL https://www.aclweb.org/anthology/D19-5801.

G. D. Forney. The Viterbi algorithm. *Proceedings of the IEEE*, 61:268–278, 1973.

N. Ganganath, C.-T. Cheng, and C. Tse. Data clustering with cluster size constraints using a modified k-means algorithm. In *Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2014 International Conference on*, pages 158–161, Oct 2014. doi: 10.1109/CyberC.2014.36.

R. Gangi Reddy, D. Contractor, D. Raghu, and S. Joshi. Multi-level memory for task oriented dialogs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3744–3754, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1375. URL https://www.aclweb.org/anthology/N19-1375.

M. Gardner, J. Grus, M. Neumann, O. Tafjord, P. Dasigi, N. F. Liu, M. Peters, M. Schmitz, and L. S. Zettlemoyer. Allennlp: A deep semantic natural language processing platform. 2017.

F. Gey, R. Larson, M. Sanderson, K. Bischoff, T. Mandl, C. Womser-Hacker, D. Santos, P. Rocha, G. M. Di Nunzio, and N. Ferro. Geoclef 2006: the clef 2006 cross-language geographic information retrieval track overview. In *Workshop of the Cross-Language Evaluation Forum for European Languages*, pages 852–876. Springer, 2006.

J. Goldstein, V. O. Mittal, J. G. Carbonell, and M. Kantrowitz. Multi-document summarization by sentence extraction. In *NAACL-ANLP 2000 Workshop: Automatic Summarization*, 2000.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

S. Guo, K. Liu, S. He, C. Liu, J. Zhao, and Z. Wei. Ijcnlp-2017 task 5: Multi-choice question answering in examinations. In *IJCNLP*, pages 34–40, 2017.

Y. Guo, Z. Cheng, L. Nie, X.-S. Xu, and M. Kankanhalli. Multi-modal preference modeling for product search. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1865–1873, 2018.

M. Gupta, N. Kulkarni, R. Chanda, A. Rayasam, and Z. C. Lipton. Amazonqa: a review-based question answering task. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 4996–5002. AAAI Press, 2019.

S. Harel, S. Albo, E. Agichtein, and K. Radinsky. Learning novelty-aware ranking of answers to complex questions. In *The World Wide Web Conference*, pages 2799–2805, 2019.

L. He, M. Lewis, and L. Zettlemoyer. Question-answer driven semantic role labeling: Using natural language to annotate natural language. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 643–653, 2015. URL http://aclweb.org/anthology/D/D15/D15-1076.pdf.

L. He, K. Lee, M. Lewis, and L. Zettlemoyer. Deep semantic role labeling: What works and what's next. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume*

1: Long Papers, pages 473–483, 2017. doi: 10.18653/v1/P17-1044. URL https://doi.org/10.18653/v1/P17-1044.

L. He, K. Lee, O. Levy, and L. Zettlemoyer. Jointly predicting predicates and arguments in neural semantic role labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers*, pages 364–369, 2018. URL https://aclanthology.info/papers/P18-2058/p18-2058.

M. Henderson, R. Al-Rfou, B. Strope, Y.-H. Sung, L. Lukács, R. Guo, S. Kumar, B. Miklos, and R. Kurzweil. Efficient natural language response suggestion for smart reply. *arXiv preprint arXiv:1705.00652*, 2017.

D. Hewlett, A. Lacoste, L. Jones, I. Polosukhin, A. Fandrianto, J. Han, M. Kelcey, and D. Berthelot. Wikireading: A novel large-scale language understanding task over wikipedia. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*, 2016. URL https://www.aclweb.org/anthology/P16-1145/.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

T. Hofmann. Probabilistic latent semantic analysis. In *In Proc. of Uncertainty in Artificial Intelligence, UAI'99*, pages 289–296, 1999.

S. Honda, S. Shi, and H. R. Ueda. Smiles transformer: Pre-trained molecular fingerprint for low data drug discovery. 2019.

D. Hoogeveen, A. Bennett, Y. Li, K. M. Verspoor, and T. Baldwin. Detecting misflagged duplicate questions in community question-answering archives. In *Twelfth international AAAI conference on web and social media*, 2018a.

D. Hoogeveen, L. Wang, T. Baldwin, and K. M. Verspoor. Web forum retrieval and text analytics: A survey. *Foundations and Trends in Information Retrieval*, 12(1):1–163, 2018b.

M. Hopkins, R. Le Bras, C. Petrescu-Prahova, G. Stanovsky, H. Hajishirzi, and R. Koncel-Kedziorski. SemEval-2019 task 10: Math question answering. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 893–899, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/S19-2153. URL https://www.aclweb.org/anthology/S19-2153.

R. Hu and A. Singh. Transformer is all you need: Multimodal multitask learning with a unified transformer. *CoRR*, abs/2102.10772, 2021. URL https://arxiv.org/abs/2102.10772.

B. Huang and K. Carley. A hierarchical location prediction neural network for twitter user geolocation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4732–4742, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1480. URL https://www.aclweb.org/anthology/D19-1480.

J. Huang, O. Etzioni, L. Zettlemoyer, K. Clark, and C. Lee. Revminer: An extractive interface for navigating reviews on a smartphone. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, pages 3–12, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1580-7. doi: 10.1145/2380116.2380120. URL http://doi.acm.org/10.1145/2380116.2380120.

L. Huang, R. Le Bras, C. Bhagavatula, and Y. Choi. Cosmos QA: Machine reading comprehension with contextual commonsense reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2391–2401, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1243. URL https://www.aclweb.org/anthology/D19-1243.

Z. Huang, W. Xu, and K. Yu. Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991, 2015. URL http://arxiv.org/abs/1508.01991.

M. Iyyer, J. Boyd-Graber, L. Claudino, R. Socher, and H. Daumé III. A neural network for factoid question answering over paragraphs. In *Empirical Methods in Natural Language Processing*, 2014.

D. Jiaoman, L. Lei, and L. Xiang. Travel planning problem considering site selection and itinerary making. In *Proceedings of the 2018 Conference on Research in Adaptive and Convergent Systems*, RACS '18, page 29–36, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450358859. doi: 10.1145/3264746.3264781. URL https://doi.org/10.1145/3264746.3264781.

A. Jitta and A. Klami. On controlling the size of clusters in probabilistic clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1601–1611, 2017. doi: 10.18653/v1/P17-1147. URL https://doi.org/10.18653/v1/P17-1147.

V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, 2020.

T. Kew, A. Shaitarova, I. Meraner, J. Goldzycher, S. Clematide, and M. Volk. Geotagging a diachronic corpus of alpine texts: Comparing distinct approaches to toponym recognition. In *Proceedings of the Workshop on Language Technology for Digital Historical Archives*, pages 11–18, 2019.

D. Khashabi, S. Chaturvedi, M. Roth, S. Upadhyay, and D. Roth. Looking beyond the surface:a challenge set for reading comprehension over multiple sentences. In *Proceedings of North American Chapter of the Association for Computational Linguistics (NAACL)*, 2018.

T. Khot, A. Sabharwal, and P. Clark. Answering complex questions using open information extraction. *CoRR*, abs/1704.05572, 2017. URL http://arxiv.org/abs/1704.05572.

H. D. Kim, K. Ganesan, P. Sondhi, and C. Zhai. Comprehensive review of opinion summarization. 2011.

R. Kiros, Y. Zhu, R. Salakhutdinov, R. S. Zemel, R. Urtasun, A. Torralba, and S. Fidler. Skip-thought vectors. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *NIPS*, pages 3294–3302, 2015. URL http://dblp.uni-trier.de/db/conf/nips/nips2015.html#KirosZSZUTF15.

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press, 2009. ISBN 0262013193, 9780262013192.

S. Kundu and H. T. Ng. A question-focused multi-factor attention network for question answering. In *AAAI*, pages 5828–5835. AAAI Press, 2018.

T. Kwiatkowski, E. Choi, Y. Artzi, and L. S. Zettlemoyer. Scaling semantic parsers with on-the-fly ontology matching. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1545–1556, 2013. URL http://aclweb.org/anthology/D/D13/D13-1161.pdf.

T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, M. Kelcey, J. Devlin, K. Lee, K. N. Toutanova, L. Jones, M.-W. Chang, A. Dai, J. Uszkoreit, Q. Le, and S. Petrov. Natural questions: a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*, 2019. URL https://tomkwiat.users.x20web.corp.google.com/papers/natural-questions/main-1455-kwiatkowski.pdf.

J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. ISBN 1-55860-778-1. URL http://dl.acm.org/citation.cfm?id=645530.655813.

G. Lai, Q. Xie, H. Liu, Y. Yang, and E. Hovy. Race: Large-scale reading comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794. Association for Computational Linguistics, 2017. doi: 10.18653/v1/D17-1082. URL http://aclweb.org/anthology/D17-1082.

T. M. Lai, T. Bui, and S. Li. A review on deep learning techniques applied to answer selection. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 2132–2144, 2018. URL https://aclanthology.info/papers/C18-1181/c18-1181.

Y. Lan and J. Jiang. Query graph generation for answering multi-hop complex questions from knowledge bases. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 969–974, 2020.

Q. Le and T. Mikolov. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pages II–1188–II–1196. JMLR.org, 2014. URL http://dl.acm.org/citation.cfm?id=3044805.3045025.

J. Lee, M. Seo, H. Hajishirzi, and J. Kang. Contextualized sparse representations for real-time open-domain question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 912–919, 2020.

J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer, and C. Bizer. DBpedia - a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal*, 6(2):167–195, 2015. URL http://jens-lehmann.org/files/2015/swj_dbpedia.pdf.

J. L. Leidner, B. Martins, K. McDonough, and R. S. Purves. Text meets space: Geographic content extraction, resolution and information retrieval. In J. M. Jose, E. Yilmaz, J. Magalhães, P. Castells, N. Ferro, M. J. Silva, and F. Martins, editors, *Advances in Information Retrieval*, pages 669–673, Cham, 2020. Springer International Publishing. ISBN 978-3-030-45442-5.

K. Lerman and R. McDonald. Contrastive summarization: An experiment with consumer reviews. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, NAACL-Short '09, pages 113–116, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1620853.1620886.

J. Leskovec and A. Krevl. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

M. Li, L. Chen, G. Cong, Y. Gu, and G. Yu. Efficient processing of location-aware group preference queries. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, page 559–568, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450340731. doi: 10.1145/2983323.2983757. URL https://doi.org/10.1145/2983323.2983757.

C. Liang, K. Hsu, C. Huang, C. Li, S. Miao, and K. Su. A tag-based statistical english math word problem solver with understanding, reasoning and explanation. In *IJCAI*, pages 4254–4255. IJCAI/AAAI Press, 2016.

P. S. Liang. *Learning Dependency-Based Compositional Semantics*. PhD thesis, University of California, Berkeley, 2011.

K. H. Lim, S. Karunasekera, A. Harwood, and Y. George. Geotagging tweets to landmarks using convolutional neural networks with text and posting time. In *Proceedings of the*

24th International Conference on Intelligent User Interfaces: Companion, IUI '19, page 61–62, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450366731. doi: 10.1145/3308557.3308691. URL https://doi.org/10.1145/3308557.3308691.

W. Lin, Z. He, and M. Xiao. Balanced clustering: A uniform model and fast algorithm. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 2987–2993, 2019. doi: 10.24963/ijcai.2019/414. URL https://doi.org/10.24963/ijcai.2019/414.

X. V. Lin, R. Socher, and C. Xiong. Multi-hop knowledge graph reasoning with reward shaping. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3243–3253, 2018.

B. Liu and L. Zhang. A survey of opinion mining and sentiment analysis. pages 415–463, 2012. doi: 10.1007/978-1-4614-3223-4_13. URL http://dx.doi.org/10.1007/978-1-4614-3223-4_13.

B. Liu et al. Sentiment analysis and subjectivity. *Handbook of natural language processing*, 2(2010):627–666, 2010.

Y. Liu and M. Lapata. Hierarchical transformers for multi-document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5070–5081, 2019.

C. Louizos, M. Welling, and D. P. Kingma. Learning sparse neural networks through l_0 regularization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018. URL https://openreview.net/forum?id=H1Y8hhg0b.

D. Lukovnikov, A. Fischer, J. Lehmann, and S. Auer. Neural network-based question answering over knowledge graphs on word and character level. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, pages 1211–1220, Republic and Canton of Geneva, Switzerland, 2017. International World Wide Web Conferences Steering Committee. ISBN 978-1-4503-4913-0. doi: 10.1145/3038912.3052675. URL https://doi.org/10.1145/3038912.3052675.

T. Luong, H. Pham, and C. D. Manning. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1412–1421, 2015. URL http://aclweb.org/anthology/D/D15/D15-1166.pdf.

T. Mandl, P. Carvalho, G. M. Di Nunzio, F. Gey, R. R. Larson, D. Santos, and C. Womser-Hacker. Geoclef 2008: the clef 2008 cross-language geographic information retrieval track overview. In *Workshop of the Cross-Language Evaluation Forum for European Languages*, pages 808–821. Springer, 2008.

K. V. Mardia, J. T. Kent, and J. M. Bibby. *Multivariate Analysis (Probability and Mathematical Statistics)*. Academic Press, 1 edition, 1980. URL http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/0124712525.

R. McDonald, G. Brokos, and I. Androutsopoulos. Deep relevance ranking using enhanced document-query interactions. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1849–1860, 2018. URL https://aclanthology.info/papers/D18-1211/d18-1211.

T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, pages 3111–3119, USA, 2013. Curran Associates Inc. URL http://dl.acm.org/citation.cfm?id=2999792.2999959.

G. A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, Nov. 1995. ISSN 0001-0782. doi: 10.1145/219717.219748. URL http://doi.acm.org/10.1145/219717.219748.

B. Mitra and N. Craswell. An introduction to neural information retrieval. *Foundations and Trends® in Information Retrieval*, 13(1):1–126, December 2018. URL https://www.microsoft.com/en-us/research/publication/introduction-neural-information-retrieval/.

B. Mitra and N. Craswell. An updated duet model for passage re-ranking. *arXiv preprint arXiv:1903.07666*, 2019.

B. Mitra, F. Diaz, and N. Craswell. Learning to match using local and distributed representations of text for web search. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1291–1299. International World Wide Web Conferences Steering Committee, 2017.

R. J. Moore, R. Arar, G.-J. Ren, and M. H. Szymanski. Conversational ux design. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 492–497, 2017.

N. Mrksic, D. Ó. Séaghdha, B. Thomson, M. Gasic, L. M. Rojas-Barahona, P. Su, D. Vandyke, T. Wen, and S. J. Young. Counter-fitting word vectors to linguistic constraints. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 142–148, 2016. URL http://aclweb.org/anthology/N/N16/N16-1018.pdf.

A. P. Natsev, A. Haubold, J. Tešić, L. Xie, and R. Yan. Semantic concept-based query expansion and re-ranking for multimedia retrieval. In *Proceedings of the 15th International Conference on Multimedia*, MULTIMEDIA '07, pages 991–1000, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-702-5. doi: 10.1145/1291233.1291448. URL http://doi.acm.org/10.1145/1291233.1291448.

T. Nguyen, M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, and L. Deng. MS MARCO: A human generated machine reading comprehension dataset. In *CoCo@NIPS*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2016.

R. Nogueira and K. Cho. Task-oriented query reformulation with reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 574–583, 2017. URL https://aclanthology.info/papers/D17-1061/d17-1061.

P. Ochieng. Parot: Translating natural language to sparql. *Expert Systems with Applications: X*, 5:100024, 2020.

F. Őzcan, A. Quamar, J. Sen, C. Lei, and V. Efthymiou. State of the art and open challenges in natural language interfaces to data. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, pages 2629–2636, 2020.

P. Padia, B. Singhal, and K. H. Lim. User-relative personalized tour recommendation. In *IUI Workshops*, 2019.

M. Palmer, D. Gildea, and P. Kingsbury. The proposition bank: An annotated corpus of semantic roles. *Comput. Linguist.*, 31(1):71–106, Mar. 2005. ISSN 0891-2017. doi: 10.1162/0891201053630264. URL http://dx.doi.org/10.1162/0891201053630264.

P. Pantel, E. Crestan, A. Borkovsky, A.-M. Popescu, and V. Vyas. Web-scale distributional similarity and entity set expansion. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2*, EMNLP '09,

pages 938–947, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. ISBN 978-1-932432-62-6. URL http://dl.acm.org/citation.cfm?id=1699571.1699635.

M. J. Paul, C. Zhai, and R. Girju. Summarizing contrastive viewpoints in opinionated text. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 66–76, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1870658.1870665.

R. A. Pazos R., J. J. González B., M. A. Aguirre L., J. A. Martínez F., and H. J. Fraire H. *Natural Language Interfaces to Databases: An Analysis of the State of the Art*, pages 463–480. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-33021-6. doi: 10.1007/978-3-642-33021-6_36. URL http://dx.doi.org/10.1007/978-3-642-33021-6_36.

J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *In EMNLP*, 2014.

C. Pithyaachariyakul and A. Kulkarni. Automated question answering system for community-based questions. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

R. S. Purves, P. Clough, C. B. Jones, M. H. Hall, and V. Murdock. *Geographic Information Retrieval: Progress and Challenges in Spatial Search of Text*. 2018.

X. Qiu and X. Huang. Convolutional neural tensor network architecture for community-based question answering. In *IJCAI*, pages 1305–1311, 2015.

D. Raghu, N. Gupta, and Mausam. Unsupervised learning of kb queries in task-oriented dialogs. *Transactions of the Association for Computational Linguistics*, 9, 2021.

P. Rajpurkar, R. Jia, and P. Liang. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2124. URL https://www.aclweb.org/anthology/P18-2124.

Q. Ran, Y. Lin, P. Li, J. Zhou, and Z. Liu. NumNet: Machine reading comprehension with numerical reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural*

Language Processing (EMNLP-IJCNLP), pages 2474–2484, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1251. URL https://www.aclweb.org/anthology/D19-1251.

J. Rao, H. He, and J. Lin. Noise-contrastive estimation for answer selection with deep neural networks. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, pages 1913–1916, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4073-1. doi: 10.1145/2983323.2983872. URL http://doi.acm.org/10.1145/2983323.2983872.

S. Reddy, M. Lapata, and M. Steedman. Large-scale semantic parsing without question-answer pairs. *Transactions of the Association for Computational Linguistics*, 2:377–392, 2014. URL http://aclweb.org/anthology/Q14-1030.

S. Reddy, O. Täckström, M. Collins, T. Kwiatkowski, D. Das, M. Steedman, and M. Lapata. Transforming dependency structures to logical forms for semantic parsing. *Transactions of the Association for Computational Linguistics*, 4:127–140, 2016.

S. Reddy, D. Chen, and C. D. Manning. Coqa: A conversational question answering challenge. *CoRR*, abs/1808.07042, 2018. URL http://arxiv.org/abs/1808.07042.

R. Riegel, A. G. Gray, F. P. S. Luus, N. Khan, N. Makondo, I. Y. Akhalwaya, H. Qian, R. Fagin, F. Barahona, U. Sharma, S. Ikbal, H. Karanam, S. Neelam, A. Likhyani, and S. K. Srivastava. Logical neural networks. *CoRR*, abs/2006.13155, 2020. URL https://arxiv.org/abs/2006.13155.

S. Robertson and H. Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, Apr. 2009. ISSN 1554-0669. doi: 10.1561/1500000019. URL http://dx.doi.org/10.1561/1500000019.

S. Romeo, G. Da San Martino, A. Barrón-Cedeno, A. Moschitti, Y. Belinkov, W.-N. Hsu, Y. Zhang, M. Mohtarami, and J. Glass. Neural attention for learning to rank questions in community question answering. In *Proceedings of the 26th International Conference on Computational Linguistics, Osaka, Japan*, 2016.

D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning Representations by Back-propagating Errors. *Nature*, 323(6088):533–536, 1986. doi: 10.1038/323533a0. URL http://www.nature.com/articles/323533a0.

M. Sachan, T. Faruquie, L. Subramaniam, and M. Mohania. Using text reviews for product entity completion. In *Proceedings of 5th International Joint Conference on*

Natural Language Processing, pages 983–991, Chiang Mai, Thailand, November 2011. Asian Federation of Natural Language Processing. URL http://www.aclweb.org/anthology/I11-1110.

M. Sachan, K. Dubey, and E. P. Xing. Science question answering using instructional materials. In *ACL (2)*. The Association for Computer Linguistics, 2016.

M. Saeidi, M. Bartolo, P. Lewis, S. Singh, T. Rocktäschel, M. Sheldon, G. Bouchard, and S. Riedel. Interpretation of natural language rules in conversational machine reading. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2087–2097, 2018. URL https://aclanthology.info/papers/D18-1233/d18-1233.

D. Saha, A. Floratou, K. Sankaranarayanan, U. F. Minhas, A. R. Mittal, and F. Özcan. Athena: an ontology-driven system for natural language querying over relational data stores. *Proceedings of the VLDB Endowment*, 9(12):1209–1220, 2016.

D. Santos and L. M. Cabral. Gikiclef: Expectations and lessons learned. In *Workshop of the Cross-Language Evaluation Forum for European Languages*, pages 212–222. Springer, 2009.

S. Scheider, E. Nyamsuren, H. Kruiger, and H. Xu. Geo-analytical question-answering with gis. *International Journal of Digital Earth*, 0(0):1–14, 2020. doi: 10.1080/17538947.2020.1738568. URL https://doi.org/10.1080/17538947.2020.1738568.

M. Seo, J. Lee, T. Kwiatkowski, A. Parikh, A. Farhadi, and H. Hajishirzi. Real-time open-domain question answering with dense-sparse phrase index. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4430–4441, 2019.

M. J. Seo, H. Hajishirzi, A. Farhadi, O. Etzioni, and C. Malcolm. Solving geometry problems: Combining text and diagram interpretation. In *EMNLP*, pages 1466–1476. The Association for Computational Linguistics, 2015. ISBN 978-1-941643-32-7.

M. J. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi. Bidirectional attention flow for machine comprehension. *CoRR*, abs/1611.01603, 2016. URL http://arxiv.org/abs/1611.01603.

D. Shahaf, C. Guestrin, and E. Horvitz. Trains of thought: Generating information maps. In *International World Wide Web Conference (WWW)*, 2012.

Y. Shen, W. Rong, N. Jiang, B. Peng, J. Tang, and Z. Xiong. Word embedding based correlation model for question/answer matching. *arXiv preprint arXiv:1511.04646*, 2015.

P. Singh and E. Simperl. Using semantics to search answers for unanswered questions in q&a forums. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 699–706. International World Wide Web Conferences Steering Committee, 2016.

R. Sipos and T. Joachims. Generating comparative summaries from reviews. In *22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013*, pages 1853–1856, 2013.

F. Souza, R. Nogueira, and R. Lotufo. Portuguese named entity recognition using bert-crf. *arXiv preprint arXiv:1909.10649*, 2019.

I. Srba and M. Bielikova. A comprehensive survey and classification of approaches for community question answering. *ACM Trans. Web*, 10(3):18:1–18:63, Aug. 2016. ISSN 1559-1131. doi: 10.1145/2934687. URL http://doi.acm.org/10.1145/2934687.

R. L. Stratonovich. Conditional markov processes. In *Non-linear transformations of stochastic processes*, pages 427–453. Elsevier, 1965.

H. Sun, H. Ma, W.-t. Yih, C.-T. Tsai, J. Liu, and M.-W. Chang. Open domain question answering via semantic enrichment. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, pages 1045–1055, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3469-3. doi: 10.1145/2736277.2741651. URL http://doi.acm.org/10.1145/2736277.2741651.

H. Sun, B. Dhingra, M. Zaheer, K. Mazaitis, R. Salakhutdinov, and W. Cohen. Open domain question answering using early fusion of knowledge bases and text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4231–4242, 2018.

I. Sutskever, O. Vinyals, and Q. Le. Sequence to sequence learning with neural networks. *Advances in NIPS*, 2014.

O. Tafjord, M. Gardner, K. Lin, and P. Clark. Quartz: An open-domain dataset of qualitative relationship questions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November*

3-7, 2019, pages 5940–5945, 2019. doi: 10.18653/v1/D19-1608. URL https://doi.org/10.18653/v1/D19-1608.

M. Tan, B. Xiang, and B. Zhou. Lstm-based deep learning models for non-factoid answer selection. *CoRR*, abs/1511.04108, 2015. URL http://arxiv.org/abs/1511.04108.

A. Trischler, T. Wang, X. Yuan, J. Harris, A. Sordoni, P. Bachman, and K. Suleman. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada, Aug. 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-2623. URL https://www.aclweb.org/anthology/W17-2623.

P. Trivedi, G. Maheshwari, M. Dubey, and J. Lehmann. A corpus for complex question answering over knowledge graphs. In *Proceedings of 16th International Semantic Web Conference - Resources Track (ISWC'2017)*, 2017. URL http://jens-lehmann.org/files/2017/iswc_lcquad.pdf.

Y.-H. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L.-P. Morency, and R. Salakhutdinov. Multimodal transformer for unaligned multimodal language sequences. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6558–6569, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1656. URL https://aclanthology.org/P19-1656.

G. Tsatsanifos and A. Vlachou. On processing top-k spatio-textual preference queries. In *Proceedings of the 18th International Conference on Extending Database Technology, EDBT 2015, Brussels, Belgium, March 23-27, 2015*, pages 433–444, 2015. doi: 10.5441/002/edbt.2015.38. URL https://doi.org/10.5441/002/edbt.2015.38.

S. Vakulenko, J. D. Fernandez Garcia, A. Polleres, M. de Rijke, and M. Cochez. Message passing for complex question answering over knowledge graphs. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, page 1431–1440, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450369763. doi: 10.1145/3357384.3358026. URL https://doi.org/10.1145/3357384.3358026.

D. Valcarce, A. Bellogín, J. Parapar, and P. Castells. Assessing ranking metrics in top-n recommendation. *Inf. Retr. J.*, 23(4):411–448, 2020. doi: 10.1007/s10791-020-09377-x. URL https://doi.org/10.1007/s10791-020-09377-x.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, u. Kaiser, and I. Polosukhin. Attention is all you need. In *Proceedings of the 31st International*

Conference on Neural Information Processing Systems, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

N. Vo, L. Jiang, C. Sun, K. Murphy, L.-J. Li, L. Fei-Fei, and J. Hays. Composing text and image for image retrieval-an empirical odyssey. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6439–6448, 2019.

E. Voorhees. Query expansion using lexical-semantic relations. In B. Croft and C. Rijsbergen, editors, *SIGIR ?94*, pages 61–69. Springer London, 1994. ISBN 978-3-540-19889-5. doi: 10.1007/978-1-4471-2099-5_7. URL http://dx.doi.org/10.1007/978-1-4471-2099-5_7.

D. Vorona, A. Kipf, T. Neumann, and A. Kemper. Deepspace: Approximate geospatial query processing with deep learning. In *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 500–503, 2019.

A. Vtyurina and C. L. Clarke. Complex questions:let me google it for you. In *Proceedings of the second Web QA Workshop WEBQA 2016*, 2016. URL http://plg2.cs.uwaterloo.ca/~avtyurin/WebQA2016/papers/paper4.pdf.

D. Wang and E. Nyberg. CMU OAQA at TREC 2015 liveqa: Discovering the right answer with clues. In *Proceedings of The Twenty-Fourth Text REtrieval Conference, TREC 2015, Gaithersburg, Maryland, USA, November 17-20, 2015*, 2015. URL http://trec.nist.gov/pubs/trec24/papers/oaqa-QA.pdf.

D. Wang and E. Nyberg. Mu oaqa at trec 2016 liveqa: An attentional neural encoder-decoder approach for answer rankin. In *Proceedings of The Twenty-Fifth Text REtrieval Conference, TREC 2016*, 2016. URL http://trec.nist.gov/pubs/trec25/papers/CMU-OAQA-QA.pdf.

Y. Wang. Distributed gibbs sampling of the latent topic models : The gritty details. 2008. URL http://cxwangyi.files.wordpress.com/2012/01/llt.pdf.

J. Welbl, P. Stenetorp, and S. Riedel. Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association for Computational Linguistics*, 6:287–302, 2018. doi: 10.1162/tacl_a_00021. URL https://www.aclweb.org/anthology/Q18-1021.

T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, and J. Brew. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771, 2019.

J. Xia, C. Wu, and M. Yan. Incorporating relation knowledge into commonsense reading comprehension with multi-task learning. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 2393–2396, New York, NY, USA, 2019a. Association for Computing Machinery. ISBN 9781450369763. doi: 10.1145/3357384.3358165. URL https://doi.org/10.1145/3357384.3358165.

Q. Xia, Z. Li, M. Zhang, M. Zhang, G. Fu, R. Wang, and L. Si. Syntax-aware neural semantic role labeling. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):7305–7313, Jul. 2019b. doi: 10.1609/aaai.v33i01.33017305. URL https://ojs.aaai.org/index.php/AAAI/article/view/4717.

H. Xu, B. Liu, L. Shu, and P. S. Yu. Bert post-training for review reading comprehension and aspect-based sentiment analysis. In *Proceedings of NAACL-HLT*, pages 2324–2335, 2019.

K. Xu, S. Reddy, Y. Feng, S. Huang, and D. Zhao. Question Answering on Freebase via Relation Extraction and Textual Evidence. In *Proceedings of the Association for Computational Linguistics (ACL 2016)*, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://sivareddy.in/papers/kun2016question.pdf.

Z. Xu, H.-T. Zheng, S. Zhai, and D. Wang. Knowledge and cross-pair pattern guided semantic matching for question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9370–9377, 2020.

B. Yang and T. M. Mitchell. A joint sequential and relational model for frame-semantic parsing. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 1247–1256, 2017. URL https://aclanthology.info/papers/D17-1128/d17-1128.

Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. W. Cohen, R. Salakhutdinov, and C. D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

K. Yatani, M. Novati, A. Trusty, and K. N. Truong. Review spotlight: A user interface for summarizing user-generated reviews using adjective-noun word pairs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages

1541–1550, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0228-9. doi: 10.1145/1978942.1979167. URL http://doi.acm.org/10.1145/1978942.1979167.

W. Yih, M. Chang, X. He, and J. Gao. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *ACL (1)*, pages 1321–1331. The Association for Computer Linguistics, 2015.

W. Yih, M. Richardson, C. Meek, M. Chang, and J. Suh. The value of semantic parse labeling for knowledge base question answering. In *ACL (2)*. The Association for Computer Linguistics, 2016.

P. Yin, N. Duan, B. Kao, J. Bao, and M. Zhou. Answering questions with complex semantic constraints on open knowledge bases. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, pages 1301–1310, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3794-6. doi: 10.1145/2806416.2806542. URL http://doi.acm.org/10.1145/2806416.2806542.

M. L. Yiu, X. Dai, N. Mamoulis, and M. Vaitis. Top-k spatial preference queries. In *2007 IEEE 23rd International Conference on Data Engineering*, pages 1076–1085, 2007.

A. W. Yu, H. Lee, and Q. Le. Learning to skim text. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1880–1890, 2017.

S. Yuepeng, L. Min, and W. Cheng. A modified k-means algorithm for clustering problem with balancing constraints. In *Proceedings of the 2011 Third International Conference on Measuring Technology and Mechatronics Automation - Volume 01*, ICMTMA '11, pages 127–130, Washington, DC, USA, 2011. IEEE Computer Society. ISBN 978-0-7695-4296-6. doi: 10.1109/ICMTMA.2011.37. URL http://dx.doi.org/10.1109/ICMTMA.2011.37.

L. S. Zettlemoyer. *Learning to map sentences to logical form*. PhD thesis, Massachusetts Institute of Technology, 2009.

C. Zhai, A. Velivelli, and B. Yu. A cross-collection mixture model for comparative text mining. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 743–748, New York, NY, USA, 2004. ACM. ISBN 1-58113-888-1. doi: 10.1145/1014052.1014150. URL http://doi.acm.org/10.1145/1014052.1014150.

C. Zhang, Y. Zhang, W. Zhang, and X. Lin. Inverted linear quadtree: Efficient top k spatial keyword search. *IEEE Transactions on Knowledge and Data Engineering*, 28 (7):1706–1721, 2016.

K. Zhang, W. Wu, F. Wang, M. Zhou, and Z. Li. Learning distributed representations of data in community question answering for question retrieval. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, WSDM '16, pages 533–542, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3716-8. doi: 10.1145/2835776.2835786. URL http://doi.acm.org/10.1145/2835776.2835786.

X. Zhang, J. Wu, Z. He, X. Liu, and Y. Su. Medical exam question answering with large-scale reading comprehension. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5706–5713, 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16582.

J. Zhao, M. Yu, H. Chen, B. Li, L. Zhang, Q. Song, L. Ma, H. Chai, and J. Ye. POI semantic model with a deep convolutional structure. *CoRR*, abs/1903.07279, 2019a. URL http://arxiv.org/abs/1903.07279.

W. Zhao, T. Chung, A. K. Goyal, and A. Metallinou. Simple question answering with subgraph ranking and joint-scoring. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 324–334, 2019b. URL https://aclweb.org/anthology/papers/N/N19/N19-1029/.

W. Zheng, J. X. Yu, L. Zou, and H. Cheng. Question answering over knowledge graphs: Question understanding via template decomposition. *Proc. VLDB Endow.*, 11(11): 1373–1386, July 2018. ISSN 2150-8097. doi: 10.14778/3236187.3236192. URL https://doi.org/10.14778/3236187.3236192.

J. Zhou and W. Xu. End-to-end learning of semantic role labeling using recurrent neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 1127–1137, 2015.

J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun. Graph neural networks: A review of methods and applications. *arXiv preprint arXiv:1812.08434*, 2018.

S. Zhu, D. Wang, and T. Li. Data clustering with size constraints. *Knowledge-Based Systems*, 23(8):883 – 889, 2010. ISSN 0950-7051. doi: http://dx.doi.org/10.1016/j.knosys.2010.06.003. URL http://www.sciencedirect.com/science/article/pii/S095070511000095X.

List of Publications

Peer-reviewed archival publications

- Danish Contractor, Parag Singla, Mausam: Entity-balanced Gaussian pLSA for Automated Comparison. *In proceedings of HLT-NAACL* (2016): 69-79
- Danish Contractor, Barun Patra, Mausam, Parag Singla: Constrained BERT BiLSTM CRF for understanding multi-sentence entity-seeking questions. *Natural Language Engineering Journal* 27(1): 65-87 (2021)
- Danish Contractor, Krunal Shah, Aditi Partap, Parag Singla, Mausam: Answering POI-recommendation Questions *In Proceedings of the ACM International Conference on Information and Knowledge Management 2021 (CIKM '21)*
- Danish Contractor, Shashank Goel, Mausam, and Parag Singla: Joint Spatio-Textual Reasoning for Answering Tourism Questions. *In Proceedings of the Web Conference 2021 (WWW' 21)*

Peer-reviewed non-archival publications

- Danish Contractor, Barun Patra, Parag Singla, Mausam: Understanding and Answering Multi-sentence Entity Seeking Questions *Workshop on Reasoning for Complex QA (RCQA) at AAAI* (2019)

Biography

Danish Contractor pursued his Ph.D at the Amarnath and Shashi Khosla School of IT, Department of Computer Science and Engineering, Indian Institute of Technology Delhi. He currently works as a Senior Research Scientist at IBM Research AI, Gurgaon and has over thirteen years of experience in various aspects of software development and research. His primary area of interest is Question Answering & Conversational AI and in the last nine years he has developed innovative research solutions for many real-world applications in the areas of Education, Information Management and Social Media Mining. Danish's research has been incorporated into multiple IBM products including IBM Watson Assistant and IBM eDiscovery Analyzer and has led to the creation of new offerings such as – IBM Watson Enlight for Educators and IBM Cognitive Content Collator. In 2018, Danish was named one of the top *Innovators Under 35* in India by MIT Technology Review Magazine and in 2020 he was awarded an outstanding UK Alumni Award for 'Professional achievement' by British Council India. Danish holds a Masters degree in Advanced Computer Science from the University of Cambridge and a Bachelors degree in Computer Engineering from the Netaji Subhas Institute of Technology (NSIT), University of Delhi.