Understanding Comparative Questions and Retrieving Argumentative Answers

Dissertation
zur Erlangung des akademischen Grades
doctor rerum naturalium (Dr. rer. nat.)

vorgelegt dem Rat der Fakultät für Mathematik und Informatik
der Friedrich-Schiller-Universität Jena

von M.Sc. Alexander Bondarenko
geboren am 16.04.1982 in Elista

Gutachter:

1. Prof. Dr. Matthias Hagen

2. Prof. Dr. Carsten Eickhoff

3. Prof. Dr. Nicola Ferro

Tag der öffentlichen Verteidigung: 29.06.2023

Acknowledgments

I am deeply grateful to my supervisor Matthias Hagen for providing much-needed support, guidance, and feedback throughout all these years. My immense gratitude also goes to my colleagues Maik Fröbe, Sebastian Günther, Ferdinand Schlatt, and the Webis group—I learned a lot from you. I would also like to thank my long-time collaborator Pavel Braslavski and the students who worked with me side-by-side and supported me on this journey: Ekaterina Shirshakova, Jonas Hirsch, and Jan Heinrich Reimer.

My parents and family have provided me with much-needed love and support, for which I am grateful. I would also like to thank my friends: Lyosha—that you doubtlessly believed in me, Nikolay and Masoud—for your huge support in the early years. Markus: thank you for your patience and encouragement.

Last but not least, I would like to thank Carsten Eickhoff and Nicola Ferro who were so kind to agree to be reviewers of my thesis.

Abstract

Understanding Comparative Questions and Retrieving Argumentative Answers

Making decisions is an integral part of everyday life, yet it can be a difficult and complex process. While peoples' wants and needs are unlimited, resources are often scarce, making it necessary to research the possible alternatives and weigh the pros and cons before making a decision. Nowadays, the Internet has become the main source of information when it comes to comparing alternatives, making search engines the primary means for collecting new information. However, relying only on term matching is not sufficient to adequately address requests for comparisons. Therefore, search systems should go beyond this approach to effectively address comparative information needs.

In this dissertation, I explore from different perspectives how search systems can respond to comparative questions. First, I examine approaches to identifying comparative questions and study their underlying information needs. Second, I investigate a methodology to identify important constituents of comparative questions like the to-be-compared options and to detect the stance of answers towards these comparison options. Then, I address ambiguous comparative search queries by studying an interactive clarification search interface. And finally, addressing answering comparative questions, I investigate retrieval approaches that consider not only the topical relevance of potential answers but also account for the presence of arguments towards the comparison options mentioned in the questions. By addressing these facets, I aim to provide a comprehensive understanding of how to effectively satisfy the information needs of searchers seeking to compare different alternatives.

Abstract (in German)

Understanding Comparative Questions and Retrieving Argumentative Answers

Entscheidungen zu treffen ist ein wesentlicher Bestandteil des täglichen Lebens, kann aber ein schwieriger und komplexer Prozess sein. Während die Wünsche und Bedürfnisse der Menschen unbegrenzt sind, sind die Ressourcen oft knapp, sodass es notwendig ist, die möglichen Alternativen zu erforschen und die Vor- und Nachteile abzuwägen, bevor man eine Entscheidung trifft. Heutzutage ist das Internet zur wichtigsten Informationsquelle für den Vergleich von Alternativen geworden, sodass Suchmaschinen das wichtigste Mittel zur Beschaffung neuer Informationen sind. Allerdings reicht es nicht aus, sich nur auf das Term-Matching zu verlassen, um Vergleichsanfragen angemessen zu beantworten. Daher sollten die Suchsysteme über diesen Ansatz hinausgehen, um den Bedarf an vergleichenden Informationen effektiv zu decken.

In dieser Dissertation untersuche ich aus verschiedenen Perspektiven, wie Suchsysteme auf vergleichende Fragen reagieren können. Erstens untersuche ich Ansätze zur Identifizierung vergleichender Fragen und untersuche den ihnen zugrunde liegenden Informationsbedürfnisse. Zweitens untersuche ich eine Methodik zur Identifizierung wichtiger Bestandteile von Vergleichsfragen, wie z. B. die zu vergleichenden Optionen, und zur Erkennung der Haltung von Antworten gegenüber diesen Vergleichsoptionen. Dann befasse ich mich mit mehrdeutigen vergleichenden Suchanfragen, indem ich ein interaktives Interface zur Klärung von Suchanfragen untersuche. Und schließlich untersuche ich zur Beantwortung vergleichender Fragen Retrieval-Ansätze, die nicht nur die thematische Relevanz potenzieller Antworten berücksichtigen, sondern auch das Vorhandensein von Argumenten gegenüber den in den Fragen genannten Vergleichsoptionen

berücksichtigen. Durch die Behandlung dieser Aspekte will ich ein umfassendes Verständnis dafür schaffen, wie man den Informationsbedarf von Suchenden, die Alternativen vergleichen wollen, effektiv befriedigen kann.

Ehrenwörtliche Erklärung

Hiermit erkläre ich,

– dass mir die Promotionsordnung der Fakultät bekannt ist,

– dass ich die Dissertation selbst angefertigt habe, keine Textabschnitte oder Ergebnisse eines Dritten oder eigenen Prüfungsarbeiten ohne Kennzeichnung übernommen und alle von mir benutzten Hilfsmittel, persönliche Mitteilungen und Quellen in meiner Arbeit angegeben habe,

– dass ich die Hilfe eines Promotionsberaters nicht in Anspruch genommen habe und dass Dritte weder unmittelbar noch mittelbar geldwerte Leistungen von mir für Arbeiten erhalten haben, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen,

– dass ich die Dissertation nicht als Prüfungsarbeit für eine staatliche oder andere wissenschaftliche Prüfung eingereicht habe.

Bei der Auswahl und Auswertung des Materials sowie bei der Herstellung des Manuskripts haben mich folgende Personen unterstützt:

....

....

Ich habe die gleiche, eine in wesentlichen Teilen ähnliche bzw. andere Abhandlung bereits bei einer anderen Hochschule als Dissertation eingereicht: Ja / [✓ Nein].

Wenn Ja, Name der Hochschule: ....

Ergebnis: ....

Jena, den __.__.2023

Alexander Bondarenko

Contents

1 Introduction

The work of economists has suggested that one of the main drivers of decision making or choice tasks is peoples' unlimited wants given limited resources [162]. Consequently, making informed decisions often involves the collection of new knowledge and additional information about the alternatives and weighing pro and con arguments towards different options [7, 133]. Nowadays, everybody has the chance to acquire knowledge and find any kind of information on the Web on almost any topic. Thus, search engines are often a person's choice for these tasks.

The origins of modern information retrieval and web search can be traced back to libraries, where the need arose to effectively search through large collections of books [164]. While information search and retrieval methods are still widely used for search in collections of digital libraries, today's retrieval approaches are applied to many kinds of data modalities, including web documents, images, videos, music, and emails. Ubiquitous access to the Internet has made web search engines the first resort for many people to look for all kinds of information. Even for big, often life-changing decisions, more than 80 % of American adults prefer to inform themselves online rather than asking friends or family members [196]. And even for vital, health-related matters, about 70 % of American adults start with an online search [65, 66]. Thus, the Web has become one of the major sources of information, and web search engines are often the main tools for finding information. Search and retrieval applications have to correctly interpret an information need encoded in a query, to retrieve and rank data instances relevant to the query (text documents, images, etc.) from usually large collections, and to present the results to the searcher.

This dissertation deals with a specific type of information needs: comparisons. Such information needs are often formulated as comparative questions when submitted to search engines, i.e., questions requesting to compare several options like “Is it better to move abroad or stay?”, which people may ask when seeking solutions to choice tasks. I study comparative information needs from four perspectives. First, I investigate effective ways of recognizing whether questions have an intent for comparison, i.e., identifying comparative questions. Second, I study comparative questions submitted to a search engine to better understand how frequently and what types of such questions are often asked on the Web. Third, to address the ambiguity of comparative questions, I propose to use clarifying questions and clarification options that search systems can proactively suggest to the users. And finally, I analyze retrieval approaches that rank documents based on their topical relevance and account for the presence of arguments and opinions in documents towards the comparison options mentioned in the questions. These approaches were submitted to the shared tasks on argument retrieval that we organized, forming the basis for retrieving argumentative answers to comparative questions.

1.1 Comparative Information Needs

Addressing comparative information needs probably requires from search systems tailored approaches to searching for relevant information, analyzing and processing this information, and presenting it to the searcher. The result presentation commonly used in modern search engines includes a list of retrieved web documents, often a featured snippet or a direct answer, a knowledge panel, a “people also ask” panel, and paid results or ads. However, search results for comparative questions can be shown differently, e.g., as an aggregation of pro and con arguments towards the to-be-compared options, which would require some analysis of arguments like stance detection. An early-proposed alternative interface for comparisons presented search results as "ten blue links" but in two different columns for each of two comparison options typed in separate search boxes [132, 191]. Extending this idea of the result presentation for comparisons, Schildwächter et al. [169] proposed to extract from web documents sentences that contain the two to-be-compared options (also typed by the searcher in separate input fields) and that are classified as comparative. These comparative sentences are then shown to the searcher in two columns, aggregated by the "winning" option: e.g., the left column contains comparative sentences, where the left-hand typed comparison option "wins" over the right-hand typed one.

In this dissertation, I take one step further and address comparative information needs formulated as natural language questions in which the specific information need like the comparison options are yet to be automatically determined by a search system.

Analyzing Comparative Questions

To gain first insights about comparative information needs in web search, I analyze Russian comparative questions from a year-long Yandex search engine log described in Chapter 3. In the definition of *comparative* questions, I will follow the concept of Lehnert [110] presented in their computational model for question answering, who exemplary introduced questions asking for comparison, e.g., "Which is bigger, a basenji or a komondor?" which requests for information on a size comparison between two dog breeds. Note that Lehnert's question taxonomy does not include the respective category (the question falls into the *quantification* category); more details on the existing question taxonomies and the allegedly first introduction of the *comparative question* category are given in Section 2.2.

I begin the analysis of comparative questions in a data-driven fashion by randomly sampling questions from the archived Yandex log which we then labeled as comparative or not. By manual inspection of the labeled comparative questions, I observed that besides conventional question categories like *factual* (asking for facts, e.g., "Which is higher, Chimborazo or Kilimanjaro?") or *subjective* (asking for opinions and arguments, e.g., "Is it better to move abroad or stay?"), comparative questions can be additionally grouped into several categories that share some common characteristics. During grouping, I focus on two aspects: (1) the questions themselves, i.e., their form, structure, and possible intent, and (2) the answering system perspective, i.e., whether different approaches are potentially needed to process the questions and to search for relevant information. Given two question examples: "Which is better for studying computer science, Leipzig University or Jena University?" and "At which university should I study computer science?", the notable difference is that the former question contains explicit to-be-compared objects: 'Leipzig University' and 'Jena University', while the latter one does not and is thus less specific. I then subdivide the questions into *direct*, where the comparison objects are explicitly mentioned, and *indirect* comparisons. On the system's side, retrieving relevant information for indirect, less specific questions may pose additional challenges, like the need to decide what universities to compare, in what country, etc. Overall, I derive ten fine-grained categories and develop a taxonomy of comparative questions (see Section 3.1) that represents various aspects of comparative information needs such as direct or indirect comparisons, asking for facts or opinions and arguments, etc.

Based on the proposed taxonomy, we create datasets of Russian and English questions annotated as comparative or not; comparative questions are additionally labeled with fine-grained categories. Further, I develop high-precision classifiers that identify comparative questions and their fine-grained types. By applying the classifier for comparative questions on the archived year-long Yandex log, I find that about 3 % of the questions are comparative. Moreover, I find that more than 65 % of the comparative questions are non-factual (i.e., asking for opinions or arguments). Thus, answering non-factual questions may require additional steps to analyze opinions and arguments in potential answers like detecting the pro or con stance towards the to-be-compared objects.

Parsing Comparative Questions

The first step in actually answering comparative questions is question processing such as, for instance, identifying their important constituents like the to-be-compared objects. Given the example "Which is better for studying computer science, Leipzig University or Jena University?", besides the two comparison objects, another two elements may be useful for retrieving relevant information that can be used for presenting answers to the searcher. The comparison *aspect* 'studying computer science' indicates a particular facet over which a comparison of the objects should be performed, and the *predicate* 'better' specifies the direction of the comparison (i.e., the better not the worse option should be the answer).

To develop approaches for parsing comparative questions (i.e., identifying their important terms), we first create a dataset with comparative questions manually labeled with the comparison objects, aspects, and predicates (cf. Chapter 4). Further, I train and evaluate transformer-based token-level classifiers that tag each token in a question as a comparison 'object', 'aspect', 'predicate', or 'none'. The RoBERTa-based [118] token classifier is the most effective in identifying the predicates (F1 of 0.98) followed by the 'none'-token classification (F1 of 0.94) and the object classification (F1 of 0.93); the aspect classification is the hardest task (F1 of 0.80).

Answer Stance Detection

The result of a choice or a decision making task that underlies *subjective* (opposite to *factual*) comparative questions like choosing between universities is usually a selection of one of the alternatives under consideration [179]. In the process, decision making requires the overview of opinions and arguments in favor of one or the other alternative (or comparison object in our terminology). It has been suggested that users who search online for opinions and arguments on generic debated topics benefit from the results presented by separating and explicitly indicating the pro or con stance of retrieved arguments [4]. Similar positive effects on the user experience have been observed for identifying the "winning" option in comparative searches [169]. Expanding on these ideas, I hypothesize that searchers may benefit from an indication of what stances answers to comparative questions overall express towards the comparison objects. Thus, in Section 4.3, I propose transformer-based classifiers that identify subjective comparative questions and that detect the stance of potential answers (represented as text passages) towards the comparison objects as 'pro first object', 'pro second object', 'neutral', or 'no stance'. The most effective RoBERTa-based stance detector that uses sentiment prompting and masking of the comparison objects with special tokens achieves an accuracy of 0.63 on four stance labels, leaving room for future improvement.

Clarifying Comparative Questions

Ambiguous comparative questions, for instance, those without explicit comparison objects (indirect questions) or without comparison aspects, represent unclear information needs that may have varied interpretations. Common techniques that search engines usually use to tackle query ambiguity include result diversification in the sense of presenting results for different potential intents [168], query suggestions to let the user select a better query formulation [119], or a "people also ask" panel feature. Another approach that has recently been studied in information retrieval and web search is the idea that a system would engage in a conversation with the user and ask clarifying questions to refine the initial information need [218, 220]. Although numerous research efforts have been done to study clarification in web search, it has yet to be fully implemented in practical search applications.[1] To determine whether clarification interactions can help users to find more satisfactory results for searches in comparative scenarios, in Chapter 5, I present a user study on clarifying comparative information needs. The study results indicate that asking clarifying questions and proactively suggesting clarification options are beneficial for users: In particular, at least 70 % of the study participants indicated that they found clarifications useful to retrieve more relevant results for questions with unclear comparison aspects like "Which is better, Leipzig Uni-

[1] For instance, Bing was testing clarification in 2020: `https://www.seroundtable.com/generating-clarifying-questions-in-bing-search-29000.html`; however, currently it is not available to users from Germany.

versity or Jena University?" and without explicit comparison objects and aspects like "Which university should I study at?".

1.2 Argument Retrieval for Comparisons

Argument retrieval systems (often also called argument search) are developed to provide an overview of pro and con arguments for (usually) socially-debated topics like climate change. Generally, argument retrieval follows two different paradigms that employ argument mining and document retrieval in different orders: First retrieve, then mine, and vice versa [5]. For instance, the args.me search engine [202] operates on a collection of arguments crawled from online debate portals. The arguments were acquired and processed in an offline pre-processing (e.g., dividing sentences into premises and claims or assigning the stance to arguments). Retrieval is then performed in the online query phase (after argument mining). Differently, the ArgumenText search engine [58, 186] and the argument retrieval component of TARGER [50] operate on the Common Crawl[2] web crawls and first retrieve web documents and then use argument mining approaches in an online manner to extract arguments (e.g., by tagging premises and claims) and detect the stance towards the query topic.

My analysis of comparative questions from the Yandex log showed that the majority of them are non-factual (i.e., asking for opinions and arguments). Thus, answering non-factual comparative questions by retrieving documents that contain pro and con arguments towards the to-be-compared objects can benefit from the established argument retrieval methodology. In particular, argument retrieval for comparative questions should account not only for a general topical relevance but also ideally should perform more analysis of argumentative texts like, for instance, stance detection, argument mining by identifying argumentative structures (claims and premises), and estimating argument quality [201].

To foster the development of argument retrieval and argument analysis approaches, we have organized Touché,[3] a series of shared tasks and work-

[2] https://commoncrawl.org/

[3] https://touche.webis.de

shops on argument retrieval that were organized from 2020 trough 2022 in conjunction with the Conference and Labs of the Evaluation Forum (CLEF).[4] In Chapter 6, I summarize and analyze the results of the task on argument retrieval for comparative questions whose goal is to develop approaches that support users facing some choice problem from everyday life. The most successful participants' approaches use re-ranking based on important terms such as comparison objects and aspects or argument units in documents (premises and claims) and estimate argument quality. While in the first task iteration, none of the participants' approaches could outperform argumentation-agnostic BM25 [156] baseline, in the third task edition, the majority of approaches were more effective, achieving the highest nDCG@5 of 0.76 (BM25 achieved 0.47 in terms of nDCG@5).

1.3 Main Contributions

In the following sections, I describe the contributions of this dissertation. The publications on which the dissertation chapters are based are specified in Table 1.1: Findings of two publications build the basis of Chapters 3 and 4, while Chapter 5 is based on one peer-reviewed publication. Findings of three publications contribute to Chapter 6, whereas further two publications listed at the bottom of the table do not directly contribute to this dissertation and are used as related work to motivate the focus of the dissertation. Figure 1.1 illustrates the main contributions of the chapters to the potential overall pipeline for answering comparative questions.

1.3.1 Identifying and Analyzing Comparative Questions (Chapter 3)

Chapter 3 focuses on studying comparative information needs formulated as questions. We first created crowdsourced datasets containing 50,000 Russian and 31,000 English questions manually labeled as comparative or not; comparative questions were additionally labeled with fine-

[4] https://www.clef-initiative.eu/

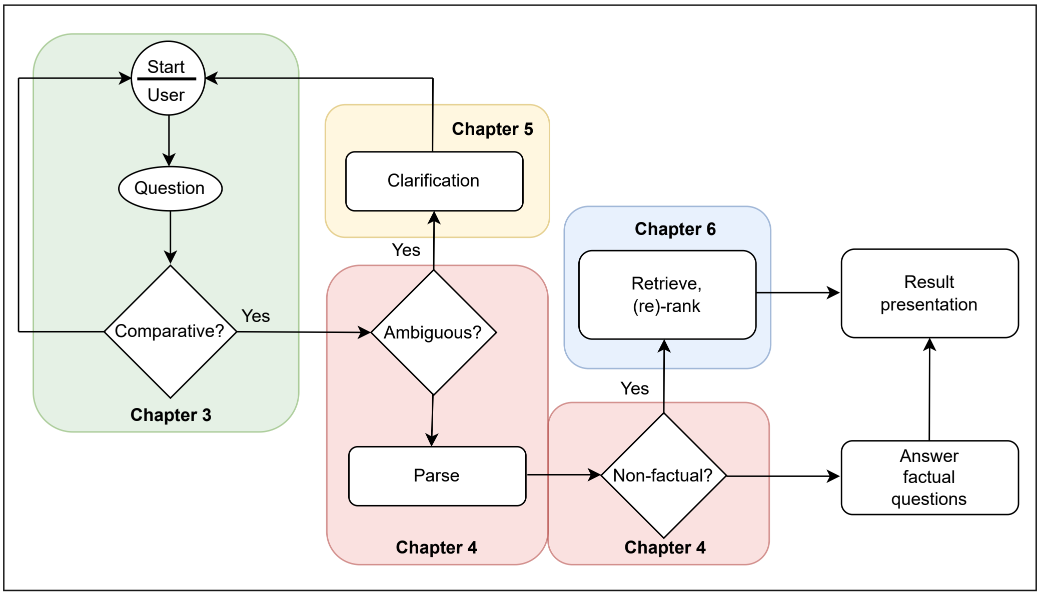


Figure 1.1: An overview of the contributions of the main chapters of this dissertation to the potential workflow for answering comparative questions.

grained categories. When deciding whether to change a (web) search result presentation for some queries, e.g., comparative questions, such queries should be reliably identified. Thus, I consider identifying comparative questions to be a high-precision classification task. To distinguish comparative questions from others, I experiment with different classifiers including hand-crafted lexico-syntactic rules, feature-based classifiers, and neural classifiers. Each classifier is optimized to always achieve a precision of 1.0 at predicting the class of comparative questions. While handcrafting the rules, I test each rule on the 80 % training set to ensure the perfect precision; for the feature-based and neural classifiers, I first select hyper-parameters to maximize the precision of predicting the comparative question class and refine the predictions afterwards by selecting operating points based on the classifiers' confidence represented by a prediction probability. When combined, a cascading ensemble of classifiers for Russian questions achieves a recall of 0.6 at a perfect precision of 1.0 for predicting comparative questions (cf. Section 3.2). In Section 3.5, I describe a similar ensemble of classifiers for English questions that recalls 71 % of the

Table 1.1: A selection of peer-reviewed publications by the author and their usage within this dissertation.

Used in	Venue	Type	Pages	Year	Ref.
Chap. 3	WSDM	Conference	9	2020	[31]
	Alexander Bondarenko*, Pavel Braslavski, Michael Völske, Rami Aly, Maik Fröbe, Alexander Panchenko, Chris Biemann, Benno Stein, and Matthias Hagen. Comparative Web Search Questions.*				
Chap. 3, 4	WSDM	Conference	9	2022	[34]
	Alexander Bondarenko*, Yamen Ajjour, Valentin Dittmar, Niklas Homann, Pavel Braslavski, and Matthias Hagen. Towards Understanding and Answering Comparative Questions.*				
Chap. 5	CHIIR	Conference	5	2022	[36]
	Alexander Bondarenko*, Ekaterina Shirshakova, and Matthias Hagen. A User Study on Clarifying Comparative Questions.*				
Chap. 6	CLEF	Conference	12	2020	[32]
	Alexander Bondarenko*, Maik Fröbe, Meriem Beloucif, Lukas Gienapp, Yamen Ajjour, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, and Matthias Hagen. Overview of Touché 2020: Argument Retrieval.*				
Chap. 6	CLEF	Conference	18	2021	[33]
	Alexander Bondarenko*, Lukas Gienapp, Maik Fröbe, Meriem Beloucif, Yamen Ajjour, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, and Matthias Hagen. Overview of Touché 2021: Argument Retrieval.*				
Chap. 6	CLEF	Conference	29	2022	[35]
	Alexander Bondarenko*, Maik Fröbe, Johannes Kiesel, Shahbaz Syed, Timon Gurcke, Meriem Beloucif, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, and Matthias Hagen. Overview of Touché 2022: Argument Retrieval.*				
–	CHIIR	Conference	5	2019	[169]
	Matthias Schildwächter, ***Alexander Bondarenko****, Julian Zenker, Matthias Hagen, Chris Biemann, and Alexander Panchenko. Answering Comparative Questions: Better than Ten-Blue-Links?.*				
–	EACL	Conference	10	2021	[48]
	Viktoriia Chekalina, ***Alexander Bondarenko****, Chris Biemann, Meriem Beloucif, Varvara Logacheva, and Alexander Panchenko. Which is Better for Deep Learning: Python or MATLAB? Answering Comparative Questions in Natural Language.*				

comparative questions in the labeled dataset. These results indicate that comparative questions can be effectively identified with high precision by combining predictions of different classifiers.

Better understanding search queries provides additional support for developing systems that retrieve information useful to the searcher. By analyzing comparative questions in the Yandex log and question and answer forum archive, I attempt to link the "what is asked" to "how it can be answered". To explore what kinds of comparative questions (and how frequently) are asked online, I perform the analysis of comparative questions and their fine-grained categories from 1.5 billion archived Yandex question-like queries and 11 million questions posted in the community question and answer forum Otvety (Russian counterpart of Quora). For instance, the analysis shows that the overwhelming majority of comparative questions in the forum (94 %) ask for opinions and arguments like which university to study at, and also 65 % of the comparative questions submitted to the search engine are non-factual. This means that for the majority of the comparative questions, knowledge bases probably cannot be used for finding the information useful to satisfy the underlying information needs. Hence, more sophisticated approaches that account for the argumentative nature of such queries are needed. I also find that comparative questions are often formulated without explicitly mentioning the to-be-compared options, making the questions ambiguous. These findings form the basis for the follow-up chapters that focus on the ambiguity of comparative questions and their argumentative nature.

1.3.2 Parsing Comparative Questions and Answer Stance Detection (Chapter 4)

Chapter 4 contributes to the steps towards answering subjective comparative questions (i.e., questions requiring opinions and arguments in answers) and focuses on the following two steps: (1) Identifying the important question constituents like the objects that the searcher intends to compare, the aspects of comparison, and the predicates indicating the direction of comparison, and (2) detecting the stance of potential answers, i.e., whether an answer expresses a preference in favor of one or another comparison object (as previously mentioned, the majority of comparative questions on the Web ask for opinions and arguments).

To identify constituents of comparative questions, for 3,500 English comparative questions (from the dataset of 31,000 English questions labeled as comparative or not), we crowdsourced token-level annotations with the comparison 'objects', 'aspects', 'predicates', or 'none' classes. Using the annotated data, I experiment with different transformer architectures and use them as token-level classifiers that tag question tokens by predicting the respective token classes (cf. Section 4.2). The evaluation results show that the easiest task is to classify the predicates (F1 of 0.98), and the hardest is aspect classification (F1 of 0.80); overall, RoBERTa model [118] was the most effective classifier. I further show that pre-classifying comparative questions as 'with an aspect' before the actual aspect tagging could improve the aspect classification by 0.1 in terms of an F1 score.

To tackle the stance detection task, for about 1,000 comparative questions, we first fetch human-written answers from Stack Exchange and Yahoo ! Answers (one "best" or "accepted" answer per question) that our annotators labeled with four stance classes: 'pro first object', 'pro second object', 'neutral', or 'no stance'. On this labeled data, I train and evaluate several configurations of stance detectors using feature-based and neural classifiers (cf. Section 4.3). The most accurate stance detector that does not require object identification is RoBERTa fine-tuned only on the answers that achieves an overall accuracy of 0.46 for four stance classes. However, identifying the first comparison object in a question and extending it with a sentiment prompt "is better" (input to the model: `O1 is better [SEP] answer`) improves the accuracy to 0.59. Whereas the overall most effective approach (accuracy of 0.63) is to identify and mask the comparison objects in questions and answers and to use sentiment prompts. In a post hoc evaluation, I prompt GPT-3 [41] to predict the stance, which achieves a slightly higher accuracy of 0.65. Since different configurations of classifiers are most effective for different stance classes, combining individual stance detectors in an ensemble can be an interesting avenue for future work.

1.3.3 Clarifying Comparative Questions (Chapter 5)

When a search system receives a question like “At which university should I study computer science?”, several details are needed to be clarified, for instance, whether the searcher is interested in specific universities or their specific properties like location or rank. Such a question can be considered as asking for ambiguous comparisons (compare to “Which is better for studying computer science, Leipzig University or Jena University?”). One possibility to tackle ambiguous information needs that has been recently studied is to clarify the initial user query. In the case of comparative questions, indirect comparative questions and questions without comparison aspects are ambiguous requests for comparisons, and, thus, can be clarified. In such cases, the system can engage in a conversation with the searcher, ask clarifying questions, and proactively suggest clarification options. Hence, in Chapter 5, I investigate whether clarification interactions are beneficial for searchers in comparative search scenarios.

To investigate to what extent clarification of ambiguous comparative questions helps searchers in finding more satisfactory answers, we conducted a user study in which study participants interact with a simulated search system to find answers to comparative questions. Our prototypical system reflects a search engine interface extended with a clarification feedback component to clarify indirect comparative questions and questions without comparison aspects. In Section 5.2, I describe the study setup, study participants, and data collection, and report the results in Sections 5.2.3 and 5.2.4. The study results showed that at least 70 % of the study participants found clarifying questions and suggested clarification options to be helpful for finding satisfactory answers to their initial ambiguous comparative questions. Additionally, the majority of participants enjoyed interacting with the system. These results indicate that clarifying questions may be a useful tool for search systems in comparative scenarios.

1.3.4 Argument Retrieval for Comparative Questions (Chapter 6)

Chapter 6 focuses on comparative questions requesting opinions and arguments in answers. In particular, the chapter analyzes the results of the shared tasks on argument retrieval for comparative searches. We organized the tasks to investigate the methodologies for retrieving and ranking documents relevant to comparative information needs that use argument analysis like argument quality estimation and answer stance detection.

With the goal of understanding what methods can enhance the effectiveness of argument retrieval for comparative questions, I overview and summarize the results of three years of organizing shared tasks on argument retrieval. In Chapter 6, I first describe the motivation, setup, and evaluation methodology and then focus on the results of the Argument Retrieval for Comparative Questions task of Touché. The main findings indicate that re-ranking first-stage retrieval results based on the assessment of argumentative facets of documents like their "argumentativeness" and argument quality almost always improves the overall retrieval effectiveness. Also, re-ranking based on important comparative terms such as comparison objects and aspects or argument units in documents (premises and claims) has been successful, improving over the BM25-based baseline in terms of topical relevance and argument quality.

1.3.5 Dissertation Structure

This dissertation is organized as follows: After the motivation in Chapter 1, Chapter 2 reviews related work providing background knowledge that the subsequent chapters are based on. Chapter 3 then describes the analysis of comparative questions that people ask online, introduces a taxonomy of such questions, and describes approaches to identify comparative questions and to classify them into fine-grained categories. Chapter 4 follows up with the methodology for question parsing by tagging the terms that are important for answering comparative questions. The chapter also contributes to the task of detecting the stance of potential answers. In Chapter 5, I address the ambiguity of comparative information needs and describe the types of comparative questions that can be ambiguous. Furthermore, I report the results of a user study on a clarification search interface for ambiguous comparative questions. Then, Chapter 6 reviews the approaches to ranking documents that contain arguments that were submitted by participants to the shared task on argument retrieval for comparative questions. Finally, Chapter 7 concludes the dissertation, summarizes its main findings, and discusses open questions and future work.

2 Background and Related Work

The following chapter introduces background research and related work that form the basis and motivate this dissertation. In Section 2.1, we overview the foundations of comparison structures in language from the linguistic perspective. This helps us better understand the arrangement of constituents in comparatives, which, in turn, supports the development of a methodology for identifying comparative questions. Further, Section 2.2 provides an overview of existing approaches to automatically classify sentences as comparative affirmatives and comparative questions that form the basis of the methodology described in Chapter 3. The section also reviews previous work on identifying the comparison objects, aspects, and predicates in sentences providing the basis for Chapter 4. This is followed by the review of the work dedicated to clarifying ambiguous information needs in web search outlined in Section 2.3. The findings of the work under review motivate our user study on the clarification interface for ambiguous comparative questions, which is presented in Chapter 5. Later, Section 2.4 overviews the argument mining and argument retrieval methodology that lays out the motivation for Chapter 6. And finally, in Section 2.5, we review task-specific applications that use comparative interfaces as a means to interact with users—this links the methodology proposed in this dissertation to some practical use cases.

2.1 Linguistic Perspective on Comparisons

Traditionally, *comparatives* have been considered in linguistic studies as a limited set of lexical structures like comparative adjectives and adverbs or comparative operators (e.g., same–as or different–than) [21, 22, 39, 185, 189, 197] that form the basis of *comparison* structures in language, or simply *comparisons*. Often, a comparison is also seen as a means of measurement [16, 197] and can serve, for instance, as a linguistic tool to correlate the degrees of some shared properties of two or more objects [221]. In the following simple affirmative example, "Mount A is higher than Mount B", the two 'mountains' (often called comparands) are compared over the 'height' (their shared property). Subsequently, the interrogative request for comparison can be expressed as a question "Is Mount A higher than Mount B?" preserving the constituents of the comparison, i.e., the comparands and the shared property. Although linguistics distinguishes between the object (i.e., what is compared), and the subject (i.e., against what it is compared) of a comparison, for simplicity and for the sake of transferring theoretical foundations into practical applications, in this dissertation, we will call both to-be-compared entities *comparison objects*.

Identification of comparison structures is related to recognizing respective specific textual signals and patterns. For instance, Moltmann [131] analyzed comparatives in English and introduced a list of comparative operators (e.g., *-er* ending of adjectives and adverbs in a comparative form) paired with comparative clause introducers (e.g., *than*). Hence, the operator–introducer pair like *-er–than* signals the presence of a comparison in text, be it an affirmative sentence or a question. Furthermore, Berezovsakaya and Hohaus [21] analyzed comparatives in English, Russian, German, Greek, and several other languages and concluded that there exist cross-language universal comparative operators like, for instance, adjectives and adverbs in a comparative form that also indicate the presence of comparisons. We will use the findings described in the aforementioned works and exploit textual signals to develop rule-based classifiers for identifying comparative questions in both the English and Russian languages. For instance, a rule `COMP` $\wedge$ `[than]` should classify a question as comparative if it contains an adjective or an adverb in a comparative form and a conjunction 'than', e.g., "Is Mount A higher than Mount B?".

2.2 Questions Asking for Comparison

In 1977, Lehnert [110] introduced a computational model of question answering that combined a question understanding with a story (context) understanding allowing to extract the answer from the story (stored in a system's memory), similar to how people would do it. Although Lehnert exemplary introduced an example of a comparative question "Which is bigger, a basenji or a komondor?" that was defined as "asking for a size comparison between two things", they did not explicitly propose a respective category for a question taxonomy. Instead, a request for comparison was seen as *relative scale questions* that could fall into different proposed categories, e.g., *quantification*. A *comparative question* category was explicitly introduced by Lauer and Peacock [108], who analyzed questions asked by expert auditors during enterprise audits, e.g., "What is the quality of your products compared with others in the industry?". Since then, current question taxonomies usually include comparative questions that are defined as asking to compare two or more things [29, 42, 74, 141].

Recently, comparative questions have drawn more attention in the question answering research community. A few works deliberately included comparative questions in datasets. For instance, Yang et al. [213] included questions asking to compare two objects in their HotpotQA question answering dataset like "Which city is larger, Pingxiang or Shijiazhuang?" (in total, 128 questions out of about 113,000 marked as comparative; all comparative questions are factual). Later, Sen et al. [172] published a multilingual dataset that contains 20,000 question–answer pairs (English examples were translated into 8 additional languages): About 10 % of the questions are factual comparisons like "Is Mont Blanc taller than Mount Rainier?".

2.2.1 Identifying Comparative Questions

Classification of question (or query) types is part of a general text classification task that exploits a wide range of different approaches [127]. A rule-based classification often uses a set of predefined handcrafted rules that map a question to a category within some taxonomy, e.g., questions that start with 'where' are assigned a 'place' category [147, 180]. Other types of classifiers include feature-based classifiers (e.g., naïve Bayes, Support Vector Machines, etc.) [123, 174, 224] and neural models (e.g., CNN, LSTM networks, transformer, etc.) [6, 29, 175, 215, 223].

Prior work on identifying comparative questions and comparative queries is rather limited. An early approach to identify comparative questions used a set of rules—sequential patterns over words, part-of-speech tags, placeholders for comparison objects, and beginning/end-of-question markers [113]. Later, Chang et al. [44] proposed a rule-based query type classification into eight categories based on keywords and their synonyms. The query type 'versus' assumed requests for comparison. The simplistic rule defined the presence of the tokens like 'vs' or 'difference' in queries to be sufficient for assigning the class 'versus'. However, the main focus of identifying comparison structures in the text has been on classifying comparative affirmative sentences in the field of opinion mining. Proposed approaches used rather a typical set of classifiers including handcrafted and class sequential rules, naïve Bayes, SVM, LSTM, CNN, and BERT [71, 85, 86, 117, 139, 194, 204, 211]. An interesting observation is that previous work on identifying comparative questions or sentences often considered *explicit* comparisons (i.e., only the cases where the comparison objects were explicitly mentioned). However, our analysis of comparative questions that people ask online shows that many of such questions do not contain explicit to-be-compared options (cf. Section 3.1).

In this dissertation, we also exploit fairly common approaches for text classification such as handcrafted rules, logistic regression, CNN [97], or transformer architectures like BERT [59]. However, our approach to identifying comparative questions is different from the previous work in two ways. First, we define the task as a high-precision classification and develop a cascading combination of classifiers that classifies comparative questions with a precision of 1.0 at each step. Second, we do not "ignore" questions without explicit to-be-compared options and address all the variations of comparative questions (direct, indirect, factual, subjective, etc.).

2.2.2 Parsing Comparative Questions

So far, only few works have been published on identifying the comparison objects, aspects, and predicates in comparative questions. An approach proposed by Li et al. [113] focused only on object identification and used class sequential rules and semantic role labeling to identify the comparison objects. Recently, a question answering system for comparative questions has been proposed that is able to identify the comparison objects, aspects, and predicates in questions [48]. The RoBERTa-based classifier [118] was however fine-tuned and evaluated on 3,000 comparative sentences (not questions). Similarly, several studies in sentiment analysis and opinion mining also proposed approaches to identify the objects, aspects, and predicates in affirmative sentences from the camera and car reviews. They exploited class sequential rules and semantic role labeling combined with SVM and naïve Bayes classifiers [85, 86, 91, 92]. Later, Arora et al. [14] experimented on 27,000 comparative sentences from camera reviews with uni- and bidirectional LSTMs [75, 81] with one and two hidden layers, and 100- and 300-dimensional GloVe embeddings [140]. They also showed that semantic role labeling applied on a larger dataset is less effective for the task than the one-layer BiLSTM classifier.

Different to most of the prior work, in this dissertation, we train and evaluate our approaches on comparative questions (not sentences). We tested several transformer models like BERT [59], ALBERT [107], or ELECTRA [54] and found that RoBERTa [118] is the most effective among these. Moreover, by pre-classifying questions as direct or indirect and as with or without the aspects, we can further improve the classification effectiveness.

2.3 Clarifying Information Needs

Vague or ambiguous search queries can make search systems "misinterpret" the correct underlying information needs. To address this issue, several solutions have been proposed like query reformulation in a conversational context or clarification, where the system proactively engages in the interaction with the searcher, be it a web search, a product search, or a voice assistant [9, 37, 89, 94, 95, 96, 101, 104, 218, 219, 220, 225].

2.3.1 Human-to-Human Interaction

Several works analyzed clarification interactions between humans (askers and answerers) to better understand what makes clarification helpful for satisfying askers' information needs. For instance, Kato et al. [89] studied the role of clarifying questions in an enterprise social question answering system. The study found that most often clarifications addressed checking the answerer's assumption about the task or the problem and requesting more information and details about the initial question. Whereas clarifying requests about the experience (e.g., "Did you try . . . ?") were most seldom. In another work, Braslavski et al. [37] analyzed clarifying responses in Stack Exchange posts and also found that the 'check' and 'more information' clarifications were the most common and that clarifying questions were overall ubiquitously present on the platform. By also analyzing Stack Exchange posts, Tavakoli et al. [192] attempted to understand what characteristics distinguish useful from non-useful clarifying questions (i.e., those that are answered by the asker and are valuable for the post and those that are left unanswered and are not valuable for the post). They found that useful clarifications often target the ambiguity and incompleteness in the initial post or attempt to confirm the responder's correct understanding of the request. Whereas the least useful clarifications targeted the assumed incorrectness in the initial post (e.g., "Are you sure . . . ?"). Thus, the three main conclusions about the human-to-human clarification interactions from the prior work are: (1) (Useful) clarifications often aim for resolving the ambiguity, (2) askers may need more support in how to clarify, and (3) clarifications target not only short (underspecified) questions.

This understanding of human communication sheds light on how clarification interaction between humans and search systems can be designed. In fact, we take into account the aforementioned findings when designing a user interface that simulates an interactive search system to study clarifications in comparative scenarios (cf. Chapter 5). In particular, our system targets the clarification of ambiguous comparative questions and supports searchers by proactively suggesting clarification options.

2.3.2 Human-to-System Interaction

While many related works focused on ranking and generating clarifying questions and creating respective corpora [9, 104, 105, 171, 219, 225], several studies analyzed user interaction with clarification interfaces. For instance, Kiesel et al. [94, 95] studied user interactions with the Amazon Alexa voice assistant and found that most of the users liked the clarification feature. They even favored unsuccessful correction attempts for false memories over no such attempts at all. An interesting conclusion was proposed that voice assistants should always ask clarifying questions for ambiguous queries because users are often open to responding to such requests.

Since in this dissertation we study comparative questions from the search or retrieval perspective, below we review in detail the works by Zamani et al. [218, 220] who studied clarifying questions in web search. Moreover, in our user study on clarifying comparative questions, we follow the ideas from these works like designing a search interface that asks clarifying questions and suggests clarification options and comparing user satisfaction with the search results before and after clarification.

In their work, Zamani et al. [218] besides proposing approaches to generate clarifying questions, also conducted two user studies. In the first study, five participants were asked to use the Bing search engine complemented with a *clarification pane* that asks clarifying questions and suggests clarification options. The study showed that all the participants were enthusiastic about the clarification pane and also believed that the search quality after clarification was improved. In the second study, 24 participants were interviewed after the interaction with the clarification pane. Most of the study participants reported that clarifications provided them with functional benefits (e.g., guided in the right direction) and emotional benefits (e.g., increased confidence in the search results).

In the follow-up, larger study Zamani et al. [220] analyzed the click-through data of Bing users in about 75 million cases when the clarification pane was shown to the users. The results revealed that (a) the user average engagement rate increased along with the query length (and also that users who submitted natural language questions were more likely to interact with the clarification pane), (b) for faceted queries the clarification pane was two times more likely to receive a click compared to the ambiguous queries, and (c) the user dissatisfaction [15] was about 17 % lower when they interacted with the clarification pane compared to the overall dissatisfaction with the search engine. In the subsequent experiment, three annotators were asked to provide an overall label (i.e., whether the whole clarification pane is useful, comprehensive, understandable, diverse, etc.) and to evaluate the landing page quality (i.e., the search quality of the secondary SERP after clarification) for 2,000 initial query–clarification pairs with the three labels: 'good', 'fair', and 'bad'. In more than 86 % of the cases, the annotators rated the clarification pane as 'fair', and in 89 % of the cases rated the results after clarification as 'good'. The results of our study align with the aforementioned findings and indicate that clarification is helpful for finding satisfactory results in comparative search scenarios.

2.4 Argument Retrieval

The goal of argument retrieval (often also called argument search) is to retrieve documents that contain arguments from (usually) large collections of documents that help to make a decision, to form an opinion, or to convince (or persuade) someone of a specific point of view. An argument is usually modeled as a conclusion (an opinion on a topic) with one or more supporting or attacking premises [202]. While a conclusion is a statement that can be accepted or rejected, a premise is a more grounded statement (e.g., statistical evidence, an anecdotal example, a referenced quote, etc.).

Argument retrieval often deals with the three following tasks: (1) Identifying argumentative queries [6], (2) mining arguments from texts [190], and (3) assessing an argument's topical relevance and quality [202]. Two different paradigms for argument retrieval have been proposed that perform argument mining and ranking in different order [5]. For instance, Wachsmuth et al. [202] extract and index arguments from online debate portals in a pre-processing step. Their argument search engine args.me[1] uses BM25F [157] to rank the extracted arguments at a query time afterwards, giving more weight to conclusions than premises. Also Levy et al. [111] first mine arguments from Wikipedia in an offline pre-processing before ranking. Following a different paradigm, Stab et al. [187] retrieve documents from the Common Crawl[2] at a query time (no prior offline argument mining) and use a topic-dependent neural network to then extract arguments from the retrieved documents. Similarly, the argument retrieval component of TARGER [50] operates on the Common Crawl web crawls and first retrieves web documents and then extracts arguments (e.g., by tagging premises and claims) and detects the stance towards the query topic.

Argument quality estimation addresses the understanding of what makes a good argument, which has been studied since the time of Aristotle [13]. In the overview study, Wachsmuth et al. [200] categorized different aspects of argument quality into a taxonomy that covers three dimensions: logic, rhetoric, and dialectic. The logic dimension concerns the strength of the internal structure of an argument (i.e., the relation between a conclusion and premises), while the rhetoric dimension covers the effectiveness of an argument in persuading an audience with its conclusion. Lastly, the dialectic dimension addresses the relation of an argument to other arguments on the topic. For example, an argument attacked by many others may be rather vulnerable in a debate. Argument relevance to a query is also categorized under the dialectical quality dimension [200].

[1] `https://www.args.me/`

[2] `http://commoncrawl.org`

Argument relevance has been typically estimated as an argument's semantic similarity to a given topic. For instance, Potthast et al. [144] evaluated four standard retrieval models for ranking arguments with regard to their topical relevance. They found that DirichletLM [222] was more effective at ranking arguments than BM25 [156], DPH [11], and tf-idf [87]. Other existing argument retrieval approaches additionally exploited argument relations. For instance, Wachsmuth et al. [203] connected two arguments in a graph when one used the other's conclusion as a premise and then computed an argument's PageRank [136] on this graph. This approach improved over a baseline that only used an argument's content and its internal structure [203]. Later, Dumani et al. [61] used the support and attack relations between clusters of premises and claims as well as between clusters of claims and a query. In an extended version, Dumani and Schenkel [60] also included the quality of a premise as a probability (fraction of premises that are worse with regard to cogency, reasonableness, and effectiveness). Using a pairwise quality estimator, the approach with the argument quality component was more effective than the one without taking argument quality into account.

2.4.1 Retrieval for Comparisons

Comparative information needs in web search were first addressed using basic interfaces for comparing two products entered separately in two search boxes [132, 191]. The search results were presented to the searcher as side-by-side two standard "ten blue links" lists for each product. Recently, identifying a comparison preference in a sentence (i.e., the "winning" option) has also been tackled more broadly (not just for product reviews) [120, 137] and forms the basis of the comparative argumentative machine CAM [169]. Similar to the early comparison interfaces, CAM takes user-specified two comparison objects and some comparison aspect(s) as input, retrieves relevant sentences using BM25, and then classifies sentence preferences (in favor of one or the other option) for a final tabular result presentation. A proper argument retrieval including argument tagging like recognizing premises and claims or argument quality estimation, however, was not included in CAM (in Section 2.5, we provide more details about the CAM interface).

In this dissertation, we are specifically interested in retrieval and ranking of documents that can be used as answers to non-factual (subjective) comparative questions—a task that has been largely overlooked—that not only account for a document topical relevance but also analyze documents' argumentative facets like the presence of arguments, argument quality estimation, and stance detection. By organizing the Touché shared tasks on argument retrieval for comparative questions, we aimed to foster the research in this direction that should ideally use the best practices from general information retrieval, argument search, and propose novel ideas.

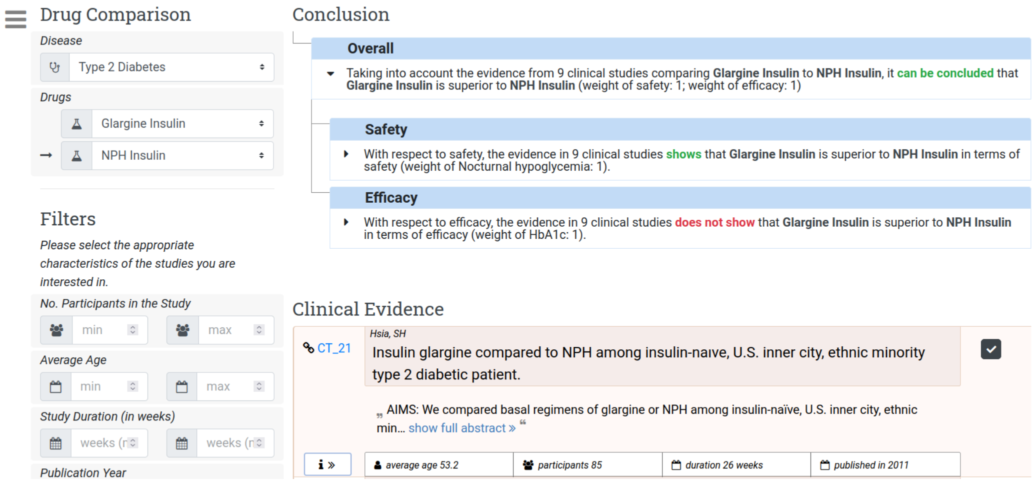
2.4.2 Stance Detection for Comparisons

Stance detection deals with the task of identifying whether some text expresses an attitude in favor, against, or neutral to a given target, usually some argumentative topic [17, 62, 64, 129, 182, 187]. The input target can be a proposition or a short phrase (e.g., a debated topic like climate change). Some researchers modify the label set for stance detection by adding further labels or by omitting the 'neutral' one. For example, in fake news detection [77], a label was added to describe texts as irrelevant for a given target. The 'neutral' label is usually omitted in domains where the texts are always polarized; for example, arguments on controversial topics are usually classified only as 'pro' or 'con' [17].

Studies that aim for detecting a "winning" object in comparative sentences [120, 137, 169] or a "preferred" object [78] are closest to our task of stance detection in comparative answers. Different from our goal of detecting the stance in answers to comparative questions that ask for opinions and arguments, these studies also classified "winning" options in factual comparisons like "gold is more expensive than silver". To address the task, Panchenko et al. [137] trained and evaluated an XGBoost classifier [49] on 7,000 sentences (labels: first object wins, second object wins, or no comparison). Later, on the same dataset, Ma et al. [120] trained and evaluated a dependency-based deep graph attention network that slightly outperformed XGBoost achieving a micro-averaged F1 of 0.87. We also tested our RoBERTa-based classifiers on the same dataset. Our classifier with unmasked objects achieves a micro-averaged F1 of 0.84, but when we mask the objects, the classifier outperforms the previous models achieving a micro-averaged F1 of 0.91 (cf. Section 4.3 for more details about our stance detector). Finally, Haque et al. [78] experimented with various transformer architectures on 9,000 sentences from mobile app reviews (labels: the mentioned app is preferred, the reference app preferred, or no preference). In our case, the targets are the comparison objects that are usually short phrases covering single concepts (e.g., 'move abroad' vs. 'stay'). In our label set, we include four labels: 'pro first comparison object', 'pro second object', 'neutral', or 'no stance' label to account for answers that avoid taking the stance towards any of the comparison objects. In contrast to most existing stance detection approaches that focus on single targets, comparative questions and answers contain multiple targets. Multi-target stance classification is a relatively new variant proposed by Sobhani et al. [181] who classified the stance of tweets towards two targets (e.g., Trump vs. Clinton). Also, we detect the overall stance of text passages (not sentences) that can contain different attitudes towards the comparison objects in different sentences making the task more challenging.

2.5 Task-Specific Applications

In this section, we provide an overview of several works that illustrate use cases for using comparison in the result presentation and aim to support choice and decision making tasks. Thus, the reviewed applications can potentially benefit from the contributions of this dissertation, e.g., by extending the systems with the component that allows typing a natural language comparative question, in which the comparison terms (e.g., objects, aspects, and predicates) can be automatically identified.



Please select the appropriate characteristics of the studies you are interested in.

Taking into account the evidence from 9 clinical studies comparing **Glargine Insulin** to **NPH Insulin**, it **can be concluded** that **Glargine Insulin** is superior to **NPH Insulin** (weight of safety: 1; weight of efficacy: 1)

With respect to safety, the evidence in 9 clinical studies **shows** that **Glargine Insulin** is superior to **NPH Insulin** in terms of safety (weight of Nocturnal hypoglycemia: 1).

With respect to efficacy, the evidence in 9 clinical studies **does not show** that **Glargine Insulin** is superior to **NPH Insulin** in terms of efficacy (weight of HbA1c: 1).

Insulin glargine compared to NPH among insulin-naıve, U.S. inner city, ethnic minority type 2 diabetic patient.

„ AIMS: We compared basal regimens of glargine or NPH among insulin-naïve, U.S. inner city, ethnic min... show full abstract » “

Figure 2.1: DIAeT web interface to compare different treatments for a given disease based on evidence from clinical trials.

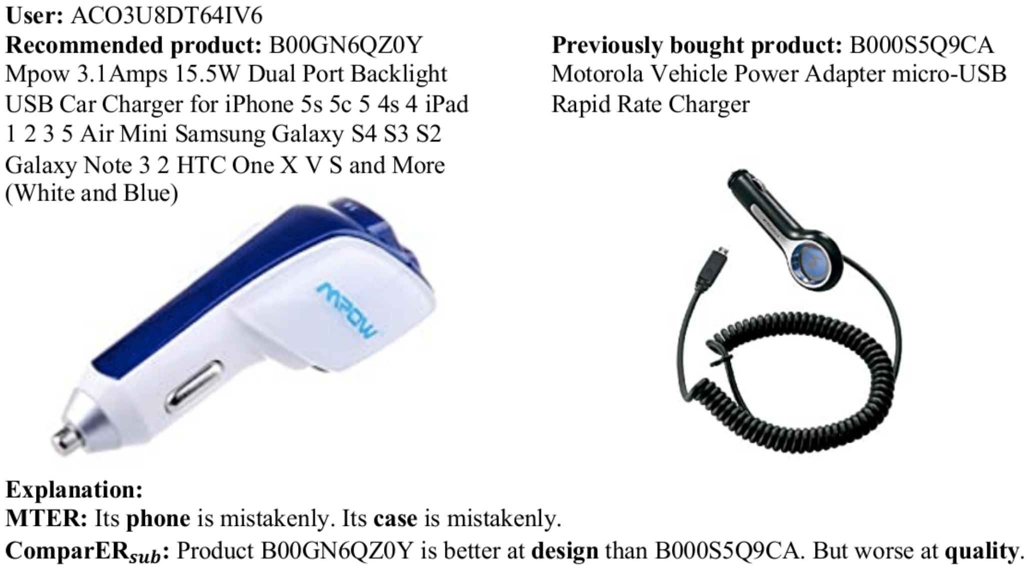
Medical Decision Making

Research in the field of evidence-based medical decision making suggests that aggregating the evidence available in multiple clinical trials is necessary to make informed decisions. Hence, Sanchez-Graillet et al. [163] proposed a framework called DIAeT: Dynamic Interactive Argumentation Trees to compare different treatments for a given disease. DIAeT automatically generates a conclusion from the existing evidence expressing the superiority of a given treatment in comparison to another treatment along several comparison dimensions extracted from clinical trials like efficacy, safety, etc. The conclusion has the form of a tree and consists of a general conclusion at the root level and several children levels representing the interim conclusions for specific comparison dimensions. The conclusion is generated using comparative templates with gaps filled by the data extracted from clinical trials. Figure 2.1 shows the DIAeT web interface.[3]

Product Recommendation

Le and Lauw [109] proposed a framework called ComparER: Comparative Explainable Recommendation (see Figure 2.2) that uses aspect-level com-

[3] `https://webtentacle1.techfak.uni-bielefeld.de/ratio-argviz/`



User: ACO3U8DT64IV6
Recommended product: B00GN6QZ0Y
Mpow 3.1Amps 15.5W Dual Port Backlight USB Car Charger for iPhone 5s 5c 5 4s 4 iPad 1 2 3 5 Air Mini Samsung Galaxy S4 S3 S2 Galaxy Note 3 2 HTC One X V S and More (White and Blue)

Previously bought product: B000S5Q9CA
Motorola Vehicle Power Adapter micro-USB Rapid Rate Charger

Explanation:
MTER: Its **phone** is mistakenly. Its **case** is mistakenly.
ComparER$_{sub}$: Product B00GN6QZ0Y is better at **design** than B000S5Q9CA. But worse at **quality**.

Figure 2.2: ComparER product recommendation interface with a comparative explanation as illustrated in [109].

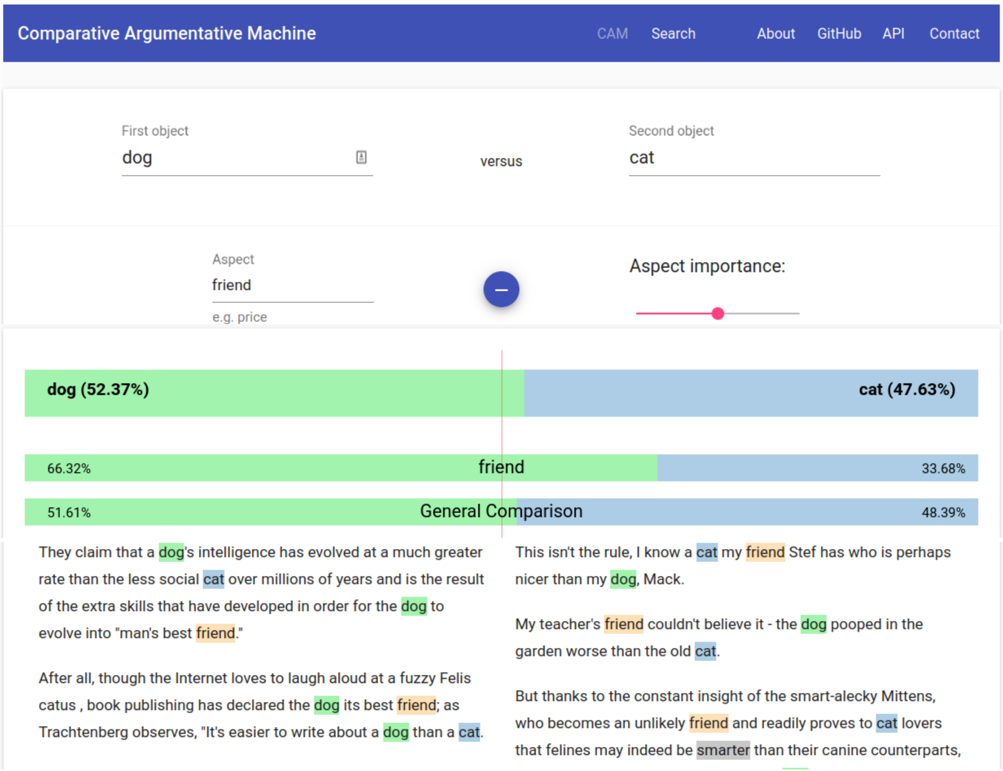
parisons between a target item and a reference item and aims for extending a product recommendation interface with an explanation component that follows the template below:

[recommended item] is better at [an aspect] than [reference item], but worse at [another aspect].

In a user study, the participants were asked whether such an explanation helped them to learn more about the recommended product. They were tasked to rate the recommendation interfaces with and without an explanation. The study results showed that the explanation-enhanced recommendation interface received significantly higher rating scores [109].

General-Purpose Comparison

An open-domain IR system to compare (any) objects, a comparative argumentative machine, was developed by Schildwächter et al. [169]. Its web interface takes two objects and some comparison aspect(s) from a user as



First object: dog

Second object: cat

Aspect: friend

Aspect importance:

They claim that a dog's intelligence has evolved at a much greater rate than the less social cat over millions of years and is the result of the extra skills that have developed in order for the dog to evolve into "man's best friend."

After all, though the Internet loves to laugh aloud at a fuzzy Felis catus , book publishing has declared the dog its best friend; as Trachtenberg observes, "It's easier to write about a dog than a cat.

This isn't the rule, I know a cat my friend Stef has who is perhaps nicer than my dog, Mack.

My teacher's friend couldn't believe it - the dog pooped in the garden worse than the old cat.

But thanks to the constant insight of the smart-alecky Mittens, who becomes an unlikely friend and readily proves to cat lovers that felines may indeed be smarter than their canine counterparts,

Figure 2.3: Web interface of the Comparative Argumentative Machine.

input, retrieves comparative sentences in favor of one or the other option, and estimates an overall "winning" option based on the number of found evidence sentences weighted by their relevance scores. Figure 2.3 shows the system's result presentation for a 'dog vs. cat' comparison over the comparison aspect of 'being a friend' (dogs win comparison).[4]

2.6 Summary

This chapter has introduced prior work on the research topics that comprise the main contributions of this dissertation. It has provided the background and given insights into the current state of research concerning the subsequent main chapters. In the first section, we focused on the linguistic basics of comparison structures in language and discussed their atomic

[4] http://ltdemos.informatik.uni-hamburg.de/cam/

constituents. Then, we reviewed text classification strategies to categorize different types of search queries and questions, including classifying comparative questions. The second section reviewed works that studied clarification approaches for ambiguous information needs. Then, we discussed argument retrieval methodology and reviewed prior work that focused on retrieval for comparative information needs. And finally, we reviewed applications that use interfaces based on a comparison result presentation.

3
Identifying and Analyzing Comparative Questions

In modern society, individuals are confronted with choice tasks on a daily basis, which are often grounded in comparing different available options. Psychologists have been studying decision making processes for decades and described different schemes and frameworks for these processes. They, however, seem to agree on a few common pieces that characterize decision making: in particular, (a) that even simple decisions like buying groceries may trigger a complex thinking process, (b) that decisions are often grounded in previous personal experience, and (c) that the process involves collecting new information and gathering knowledge about alternatives under consideration [7, 133].

The current state of the technological development of society has changed the way how and where people acquire information, including situations when they confront choice tasks. A recent study showed that for big decisions (e.g., rent vs. buy a house), about 80 % of Americans prefer to do online research rather than asking friends [196]. Hence, people turn to the Web and use search engines and fora like Quora to satisfy *comparative* information needs. Since more and more search engine queries are also formulated as actual natural language questions—a trend that is evoked by the recent advances in speech recognition and the spread of voice interfaces, which encourage users to shift from the telegram-style keyword queries to natural language questions [76, 138, 207]—in this dissertation, we focus on a special case of comparative queries: comparative questions. We hypothesize that search engine responses to such questions might be different from conventional search engine result pages (SERP) like "ten blue links", featured snippets, and direct answers. These kinds of search results miss, for instance, the opportunity to switch the output (for comparative information needs) straight to the overview that aggregates the pros and cons of the different options—similar to the decades-old idea of structured representations in e-commerce [184].

Web search usually starts with a query processing step that among others includes identifying a user intent or an information need. One of the well-known early-proposed taxonomies of web searches includes three query types: navigational, informational, and transactional [40]. Thus, before the actual search, a query type can be first classified. Accordingly, this chapter outlines a series of experiments and studies that aim for a better understanding of what users ask about when they formulate and post comparative questions on the Web, how these questions are formulated, and elaborates potential implications of the findings on search systems. In this chapter, we describe the analysis of comparative questions, i.e., questions asking to compare different options like "Should I buy or rent a house?" that were submitted to the search engine Yandex and posted on the question and answer forum Otvety@Mail.ru in 2012. Both corpora that we have at hand contain archived queries and posts that represent real users' requests and, hence, potentially reflect genuine information needs.

We start with data collection and labeling: In Section 3.1, we propose a taxonomy of comparative questions and describe the annotation procedure that follows the proposed taxonomy. Four native Russian-speaking annotators labeled 62,500 questions as comparative or not that we randomly sampled from the Yandex and Otvety archives and assigned ten fine-grained categories to the 3,000 comparative questions (e.g., whether a question asks for facts or arguments etc.). This labeled data provides initial insights into the distribution of comparative questions and their different types that search engines may receive as queries. For instance, we found that more than 65 % of the comparative questions request arguments and opinions such that reliable answers to such questions might require more than just some facts from a search engine's knowledge graph.

To classify questions as comparative, in Section 3.2 we propose a precision-oriented classifier that accurately identifies Russian comparative questions by combining carefully handcrafted lexico-syntactic rules with feature-based and neural approaches. At each step, we select the classifier's operating points such that they always predict a class of comparative questions with a precision of 1.0. The final cascading combination of classifiers recalls 60 % of the labeled comparative questions with perfect precision. In Section 3.5, we focus on English questions and, following the same idea, we develop a high-precision combination of classifiers to distinguish comparative questions from other questions. When individual steps are combined in a cascade, the classifier recalls 71 % of the comparative questions with a perfect precision of 1.0. A classifier of that quality is actually applicable in production systems since there are hardly any false positives to be expected (i.e., almost no wrong switch to a pro/con answer presentation for a question that is not comparative). Further, in Section 3.3 we describe BERT-based and CNN-based classifiers that categorize comparative questions into fine-grained types following our proposed taxonomy.

Further, Section 3.4 presents an analysis of the comparative questions identified by our classifier in the entire archive of 1.5 billion questions from the year-long Yandex log. We identify, that at least 3 % of the questions in the log are comparative (on average, there is at least one comparative question per second). Many comparative questions fall in the category of *consumer electronics* (e.g., "Which camera is better, Canon or Nikon?") followed by *cars and transportation* (e.g., "Which tires are best for the winter?"). A substantial portion of the frequently asked comparative questions does not specify concrete objects to be compared, and no comparison aspect is provided (e.g., "Which tablet is the best to buy?"). Such queries require more explanatory answers in the form of opinions or pro and con arguments that typically cannot be found in a search engine's fact-oriented knowledge graph. We, thus, also conduct a pilot study to analyze whether answers to similar questions from the Russian question and answer forum Otvety can help. For about 50 % of the comparative Yandex questions we find a fitting answer from Otvety; in particular, answers for the more frequent comparative Yandex questions are usually “mineable” from the web fora. This potential of mining answers, together with our proposed high-precision classification of comparative questions, indicates a very promising first step towards handling the result presentation for comparative questions differently than showing “ten blue links” only.

Finally, Section 3.6 concludes this chapter, discusses the implications of the contributions, and elaborates on open questions and future work.

3.1 Data Annotation and Question Taxonomy

To study real-life comparative questions, we mine them from two sources: (1) a year-long log of questions submitted to the Russian search engine Yandex in 2012, and (2) all the questions posted on the Russian question and answer platform Otvety in 2012. Following Völske et al. [198] from the Yandex query log, we extracted about 2 billion question-like entries that match any of 58 lexical question indicators (e.g., ‘how’, ‘what’, ‘where’, ‘should’), similar to the method proposed by Bendersky and Croft [20]—but adapted to the Russian language. We then clean the initial set of entries following the steps of Völske et al. [198]: removing spam and bot entries and removing consecutive duplicate entries from the same user, as well as entries not representing “genuine” user questions (e.g., crossword questions, questions from the TV game show Family Feud, or questions matching Wikipedia titles). These cleansing steps removed about 500 million of the 2 billion question-like entries, resulting in a cleaned set of 1.5 billion entries that we consider to be genuine questions (752 million unique questions from 183 million unique user IDs). Interestingly, even though the questions are in Russian, quite many of them contain Latin-spelled tokens (e.g., brands or asking for the correct spelling of some English word).

Following Völske et al. [198] again, we extracted about 6.6 million questions from the about 11 million questions posted on the Russian community question answering platform Otvety in 2012, for which a “best answer” was selected and that were asked by the users who posted at least three questions in 2012. Otvety ("answers") is the Russian counterpart of Yahoo ! Answers, with similar rules and incentives (points for good answers, etc.). Before being posted, each question is manually assigned to one of the 28 top-level topical categories by the asker. In our extraction, we omitted ambiguous categories (humor, miscellaneous, etc.) and merged closely related ones into 14 top-level categories (cf. Table 3.7).

To ensure a natural distribution of comparative questions, from the cleaned Yandex log, we randomly sampled 50,000 questions that at least three different users submitted (these questions are probably less privacy-sensitive), as well as 12,500 questions from the Otvety archive. Four native Russian-speaking annotators were instructed to label as comparative those questions that express a comparison intent through an examination of similarities or differences of two or more options, two or more groups of options, all options inside one group, or a single option against a group of options. The compared items may either be explicitly mentioned (e.g., "Which is better for studying computer science, Leipzig University or Jena University?") or may be given as a generic "set" (e.g., "Which university should I study computer science?"). In an initial annotation training phase on 200 questions, the four annotators reached an inter-annotator agreement of Fleiss' κ=0.88 (almost perfect agreement) after a round of instructions. Due to the high agreement, the annotators then labeled individual shares of the data independently (i.e., just one vote per question).

Despite extensive automatic pre-filtering aimed to remove non-genuine questions, our annotators still marked about 2,000 of the 50,000 Yandex questions and about 2,500 of the 12,500 Otvety questions as being incomplete, as parts of song lyrics that our filters had missed, or containing profanity. We replaced such questions with additional randomly-sampled questions to maintain the desired totals. Overall, the annotators labeled 1,405 Yandex questions (about 2.8 %) and 1,571 Otvety questions (about 12.6 %) as comparative (cf. Table 3.1).

In the second round of annotations, the same annotators then labeled the comparative questions from the first round with further ten fine-grained categories (annotators achieved a Fleiss' κ of 0.51 that corresponds to a

Table 3.1: Absolute and relative frequencies of the categories of comparative questions in our labeled dataset (percentages for categories are relative to the number of comparative questions). Our newly proposed categories specific to comparative questions are marked with (⋆).

	Yandex		Otvety	
Comparative	**1,405**	**(3% of all)**	**1,571**	**(13% of all)**
Opinion	916	(65%)	1,469	(94%)
Argumentative	676	(48%)	586	(37%)
Factual	378	(27%)	101	(6%)
Method	106	(8%)	41	(3%)
Reason	83	(6%)	10	(<1%)
Preference⋆ (requested)	985	(70%)	1,281	(82%)
(stated)	18	(1%)	77	(5%)
Direct⋆	603	(43%)	893	(57%)
Aspect⋆	302	(22%)	546	(35%)
Context⋆	238	(17%)	405	(26%)
Superlative⋆	180	(13%)	287	(18%)

moderate agreement; analogous training phase and the procedure of one vote per question were used) that are not mutually exclusive (i.e., a question can fall in more than one of the respective classes, and the annotators were instructed to select any that applied).

The first five categories are general question types from existing question taxonomies found in the literature. *Opinion* questions ask for a personal experience or opinion in the answer without the need of a shared settled knowledge (e.g., "Which to choose for vacation, Goa or UAE?") [99, 183, 216]. *Argumentative* questions request a solid argumentation in the answer in the form of pro and con arguments (e.g., "Who will win a presidential election, Trump or Clinton, and why?") [83]. *Factual* questions can be answered with a simple (often short) fact, where the answer is rather "static" (not changeable) over a sufficient period of time and independent of the answerer's opinion or experience (e.g., "Which contain more vitamin C, kiwis or lemons?") [3, 130]. *Method* questions request some *how to*-style instructions in the answer (e.g., "How to distinguish faux fur from real?") [128]. *Reason* questions seek an explanation or reasons in the answer that are based on scientific insights and knowledge (e.g., "What is common between proteins and amino acids?") [128].

In addition to the five general question categories from existing question taxonomies, we also asked the annotators to assign five further newly proposed categories of comparative questions. We derived these new categories, which are specific to comparative questions, in a data-driven fashion based on the manual inspection of the labeled set of comparative questions. The idea was to group questions based on their form, the underlying information need, and the potential methodology for question processing and answering. Comparative questions fall in the *preference* category if their intent to choose or select one option from several by either *requesting* a preference (e.g., "Which is more reliable, an iPhone or a Samsung?") or by explicitly *stating* a preference (e.g., "Why is an iPhone better than a Samsung?"). A comparative question is *direct* if it explicitly includes the to-be-compared objects (e.g., "Which is more reliable, an iPhone or a Samsung?") instead of implicitly determining a range of the possible items to compare (e.g., "Which mobile phone is it better to buy?"). A comparative question includes a comparison *aspect* when a particular shared property over which the objects can be compared or contrasted is mentioned. Such aspects can be stated in ascending or descending direction (e.g., asking whether a product is 'more expensive' or 'cheaper'), and they can be expressed through a simple comparative adjective or adverb (e.g., "Which is cheaper, an iPhone or a Samsung?") or through the combination of several lexical units (e.g., "Which is better for web development, PHP or Python?"). Comparative questions may also include additional *context* for the comparison (e.g., a target of a 4-year-old in "Which is better to buy for a 4-year-old, a remote control car or a toy transformer?"). Finally, a comparative question is *superlative* if it asks for the "best" item in a class and contains an adjective or an adverb in a superlative form (e.g., "Who is the best soccer player?"), rather than explicitly comparing two or more options (e.g., "Who is a better soccer player, Messi or Ronaldo?").

The labeling results (cf. Table 3.1) reveal a few interesting observations. Unsurprisingly, users on Otvety (the community question and answer platform) ask for relatively way more opinion comparisons and fewer factual comparisons than in Yandex. Still, more than 65 % of the comparative questions submitted to Yandex are also non-factual (i.e., directly answering them may be a difficult task), which motivates our further investigation of argument retrieval and analysis research like argument quality estimation and stance detection of potential argumentative answers (cf. Chapters 4 and 6). Also, about 60 % of the Yandex questions are indirect, and about 80 % miss comparison aspects, which motivates investigation of clarification approaches to refine initial information needs (cf. Chapter 5). In Section 3.3, we elaborate on the impact of different types of comparative questions on the methodology for their processing and answering.

3.2 Identifying Russian Comparative Questions

With the scenario of changing a search engine's result presentation for comparative questions in mind, we focus on the precision of identifying comparative questions (about 2.8 % of the Yandex questions). For that, we combine predictions of three different types of classifiers: (1) handcrafted lexico-syntactic rules, (2) traditional feature-based classifiers, and (3) neural networks. To develop the rules and to train and evaluate the classifiers, we split the annotated data into train (80 %) and test sets (20 %).

3.2.1 Rule-Based Classification

Inspired by the previous linguistic studies that describe comparison structures in language and their constituents [21, 22, 39, 185, 189, 197], by the opinion mining studies that identify comparative statements in reviews [71, 85, 86, 139, 194, 204, 211] and classify comparative questions [113], we use lexical and syntactic rules as a first step of our classifier aiming for perfect precision at recall as high as possible. We translated promising patterns from the literature into Russian, merged and "tuned" the rules on the training set to not end up with a too large number. Our potential 15 rules (cf. Figure 3.1) consist of regular expressions over question tokens, comparative (COMP) and superlative (SUPER) grammemes

(R1) `[better]` ∧ ¬`[how]`[1]

(R2) `COMP` ∧ `[or|vs|versus]`[2] ∧ `posn(COMP)` < `posn[or|vs|versus]` ∧ ¬`[more or less]`

(R3) `[how correct(ly)? (spell|write)]` ∧ `[or]`

(R4) `[what common|similar]` ∧ `[and|from|or|between|vs|versus]`

(R5) `[choose|buy|take]` ∧ `[or|between|vs|versus]`

(R6) `[in comparison]`

(R7) `[advantage|disadvantage|flaws]` ∧ `[of|over|compared to]`

(R8) `[difference(s)?|differentiate|distinguish]` ∧ `[and|from|or|between|vs|versus]`

(R9) `[better]`

(R10) `COMP` ∧ `[which]` ∧ ¬`[or|vs|versus]` ∧ ¬`[how]`

(R11) `[or]`

(R12) `COMP`

(R13) `COMP` ∧ `[which]` ∧ ¬`[or|vs|versus]`

(R14) `SUPER`

(R15) `[plus(es)?]` ∧ `[minus(es)?]`

Figure 3.1: Fifteen lexico-syntactic rules to classify Russian questions as comparative; presented are English translations.[1]

(identified with the MyStem POS tagger [170]), token positions (`posn`), and logical operators, and are ordered by descending precision (the ones with equal precision are ordered by descending recall).

Using the rules for classification, a given question will be classified as comparative if any of these rules matches (ignoring punctuation and capitalization). To determine a subset of rules that reach perfect precision, we

[1] Expressions in `[]` are in a regular expression syntax: so, a question matching (R1) must contain the token *better* but not the token *how* (here, tokens are approximate translations from Russian).

[2] Even though *vs* and *versus* are not Russian comparison words, they occasionally occur as such in questions; still, we do not consider them as standalone comparison indicators since only very few questions contain them ($< 0.005\,\%$), and since a significant number of *vs*-questions are non-comparative (e.g., “What is vs/versus?”).

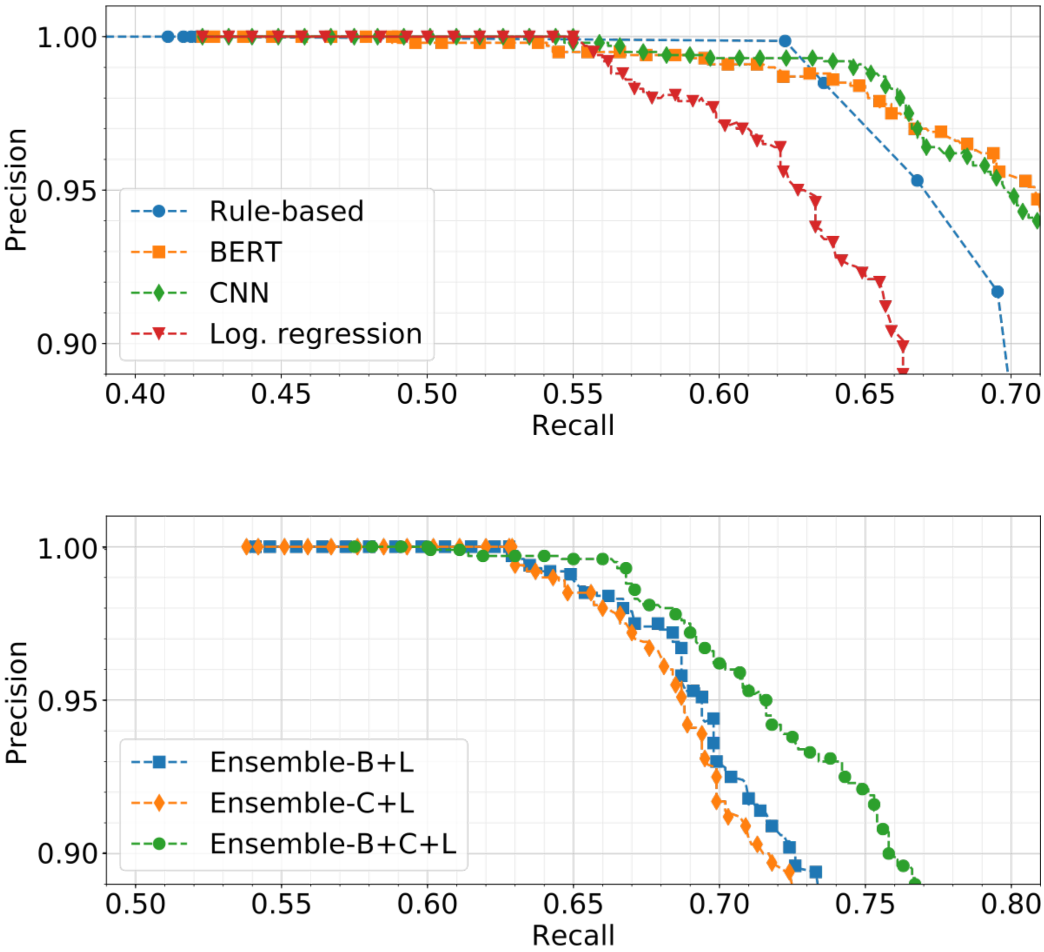


Figure 3.2: Precision-recall curves for the comparative question class on the Yandex training set.

examine their effectiveness on the training set. The blue line with circles in Figure 3.2 (top) shows the precision-recall curve resulting from successively combining the rules in descending precision order; rules (r1)–(r7) have a perfect precision of 1.0, and together achieve a recall of 0.42. Adding rule (r8) increases recall to 0.62 but slightly reduces precision to 0.9986 (a single misclassified example: "How to teach a dog to distinguish between friends and foes?"). The next rules then provide additional recall but at a much higher cost of precision (e.g., adding the rules (r9)–(r12) increases recall to above 0.70 while dropping precision to 0.74). Identifying more "perfect-precision" rules might be an interesting direction for future work.

3.2.2 Combining Rules with Feature-based and Neural Classifiers

To supplement the handcrafted rules in the pursuit of gaining more recall, we test adding several classifiers with manual feature engineering (SVM, logistic regression, naïve Bayes), as well as neural models (CNN [97], LSTM with recurrent dropout [70], capsule networks with dynamic routing [161] similar to the CapsNet-1 model [210], and BERT with a linear layer as a decoder on top [59]), which have become prevalent for text classification tasks. For all these models, we optimize the parameters in a grid search of commonly used value ranges[3] and evaluate their effectiveness in pilot experiments to identify promising combinations. Since we consider the classification of comparative questions as a high-precision task, we aim to further increase the recall of the rule-based approach at the smallest possible cost of precision. We thus train and test the feature-based and neural classifiers only on the "more difficult" questions that are not already identified as being comparative by the perfect-precision rule set (R1)–(R7). From the Yandex questions, this leaves 39,524 questions (650 comparative) as the reduced training and 9,876 questions (159 comparative) as the reduced test set. In a pre-processing step, each question is tokenized, lowercased, POS-tagged, and punctuation is removed. For the feature-based classifiers, we derive unigram bag-of-words representations (SVM, naïve Bayes) or uni- to four-gram bag-of-words representations (logistic regression), as these were the most effective setups in our pilot experiments. For CNN, LSTM, and capsule networks, fastText embeddings trained on the Russian Wikipedia are used [28]. For BERT, we fine-tune the pre-trained 'bert-based-multilingual-uncased' model with WordPiece embeddings [59].

The BERT, CNN, and logistic regression models vastly outperformed the other classifiers in our pilot experiments (higher recall at perfect preci-

[3] SVM: kernel: ["rbf", "linear"], gamma: [0.001, 0.0001], C: [1, 10, 20, 30, 40, 100, 1000]; logistic regression: penalty: ["l1", "l2"], C: [0.0001, 0.001, 0.01, 1, 100]; naïve Bayes: alpha: [0, 0.0001, 0.001, 0.01, 1]; CNN: number of filters: [25, 50, 100, 200, 500], learning rate: [0.005, 0.003, 0.001, 0.0005]; LSTM: number of units: [20, 50, 100, 200, 500], learning rate: [0.001, 0.0005, 0.0001, 0.0025], CapsNet: final layer dimensions: [8, 16, 32, 64], learning rate: [0.0005, 0.0001]; BERT: learning rate: [0.000001, 0.000002, 0.000003]; all neural classifiers: epochs: [3, 4, 5]

Pseudocode 1 Pseudo code of our Ensemble-$\mathcal{C}$ classifier.

```
Input: question $q$, classifiers $\mathcal{C} \subseteq \{\mathrm{CNN}, \mathrm{BERT}, \mathrm{Logistic}\}$
Output: 1 if $q$ is comparative, 0 otherwise
begin
    // Step 1: "perfect precision" rule set
    if ruleDecision((R1)–(R7), $q$) = 1 then return 1;
    // Step 2: "perfect precision" classifiers
    foreach $c \in \mathcal{C}$ do
        if classifierDecision($c, q$, perfectPrecisionThreshold($c$)) = 1 then return 1;
    end
    // Step 3: consensus of "non-perfect" classifiers
    $D_{\mathcal{C}} :=$ [classifierDecision($c, q$, decisionThreshold($c$)) : $c \in \mathcal{C}$]
    if unanimous($D_{\mathcal{C}}$) then return $D_{\mathcal{C}}[0]$;
    // Step 4: almost "perfect precision" rule R8
    return ruleDecision((R8), $q$)
end
```

sion). We, thus, only consider these classifiers as potential add-ons to the handcrafted rules. The models' hyperparameters are optimized to achieve the highest precision for the comparative question class using grid search and ten-fold cross-validation on the training set.[4]

Our proposed ensemble of rules and feature-based and neural classifiers is a four-step decision process (cf. algorithm in Pseudocode 1). Given a question q and a set of classification models C (some subset of BERT, CNN, and logistic regression in our case), the ensemble first applies the "perfect precision" rule set (R1)–(R7). Only if these rules do not classify q as comparative, the models from $\mathcal{C}$ are run to classify q (with an operating point selected as a decision threshold for the probabilities returned by a classification model optimized for perfect precision on the training set). If none of these models classifies q as comparative, the third step asks whether there is a consensus among the classifiers in $\mathcal{C}$ at relaxed decision thresh-

[4] Selected hyperparameters: BERT and CNN use the Adam optimizer [98] and a minibatch size of 32. BERT fine-tuning: hidden units 768, dropout prob. 0.1, learning rate 0.00002, epochs 3, sequence length 128; CNN: filters 25, learning rate 0.0005, epochs 3, dropout prob. 0.5, loss function: binary cross-entropy loss, sequence length 15. Logistic regression: penalty='l2', solver='liblinear', C=0.01.

Table 3.2: Classification results of a ten-fold cross-validation on the training set (Russian questions) aiming for the maximal recall at a precision of 1.0 for the comparative class (decision thresholds are in parenthesis). All classifiers achieve at least 0.98 precision and 1.0 recall for the non-comparative class.

Individ. model	Recall	F1	Ensembles	Recall	F1
Logistic (0.418)	0.55	0.71	Ens.-B+L (0.632)	0.63	0.77
CNN (0.99447)	0.55	0.71	Ens.-C+L (0.418)	0.63	0.77
BERT (0.99766)	0.49	0.66	Ens.-B+C+L (0.99447)	0.60	0.75

olds (aiming for a combined best possible precision tuned on the training set). If these relaxed-threshold classifiers do not reach a unanimous consensus of q being comparative or not, the fourth step just takes the decision of the high-recall but slightly imperfect-precision rule (r8) (combined precision of rules (r1-8) is 0.9986, combined recall is 0.62).

Varying the decision probability threshold of each classifier from 0 to 1 in Step 3 of the ensemble approach, its three variants performed particularly well in the pilot experiments: (1) Ensemble-B+L with $C = \{\mathrm{BERT}, \mathrm{Logistic}\}$, (2) Ensemble-C+L with $C = \{\mathrm{CNN}, \mathrm{Logistic}\}$, and (3) Ensemble-B+C+L with $C = \{\mathrm{BERT}, \mathrm{CNN}, \mathrm{Logistic}\}$. The precision-recall curves for the complete four-step ensembles on the Yandex training set are shown in Figure 3.2 (bottom), the parameter settings and recall values for the individual perfect-precision classifiers and for the complete ensembles are given in Table 3.2. The Ensemble-B+L and Ensemble-C+L outperformed all other classification models, achieving a recall of 0.63.

3.2.3 Evaluation on the Test Set

We then test the effectiveness of the developed classifiers to identifying comparative questions on the held-out test set (10,000 questions). We test the first 7 rules and other classifiers using operating points (selected on the training set) that classify comparative questions with a precision of 1.0. Table 3.3 reports the classification results. Not surprisingly, the individual models lose from 1 % (logistic regression) up to 5 % (BERT) of recall

Table 3.3: Classification results on the test set (Russian questions). The goal is to achieve the highest recall at a precision of 1.0 on the comparative class (decision thresholds are in parenthesis). All classifiers achieve at least 0.98 precision and 1.0 recall for the non-comparative class.

Individ. model	**Recall**	**F1**	**Ensembles**	**Recall**	**F1**
Logistic (0.418)	0.54	0.70	Ens.-B+L (0.632)	0.60	0.75
CNN (0.99447)	0.52	0.68	Ens.-C+L (0.418)	0.59	0.74
BERT (0.99766)	0.44	0.61	Ens.-B+C+L (0.99447)	0.55	0.71
Rules (r1–7)	0.44	0.61			

(exception is the rules (r1)–(r7) that gain 2 % of recall), as well as the ensembles—from 3 % (Ensemble-B+L) up to 5 % (Ensemble-B+C+L).

We further test the relatively faster Ensemble-C+L "in the wild" by classifying the 1.5 billion questions from the Yandex log and manually check the assigned labels for another 5,000 comparative questions: about 1 % are misclassifications (cf. Section 3.3.1).

3.3 Fine-Grained Classification of Russian Comparative Questions

The ten proposed categories of comparative questions (cf. Section 3.1) are meant to describe different types of comparative information needs. These types help to better recognize the genuine user intents and to decide what should be presented as an answer and how [149]. For instance, answers to factual and probably also many reason questions (e.g., "What is common between proteins and amino acids?") can possibly be found in knowledge bases and scientific publications and can be presented on a result page as a short direct answer [52] with the linked evidence sources. By contrast, answering opinion and argumentative questions that also ask for a preference (e.g., "Which one to buy, an iPhone or a Samsung, and why?") may trigger a search for fitting (answered) questions on some question and answer fora or a search for multiple pieces of evidence using multi-hop question-answering approaches [43, 63, 73] with summaries stemming from several documents [217]. The potential answers from different documents can be then aggregated and grouped by their stance towards different comparison objects. Finally, an answer to a method question ("How to distinguish faux fur from real?") might also be found on question and answer platforms [205] or in how-to collections like wikiHow[5] and will most likely be presented as step-wise instructions.

Existing studies that address answering comparative requests [169, 184, 191] mostly have dealt with queries (not questions) where users explicitly provide two items to be compared. This is similar to questions that we call direct comparisons (e.g., "Who is a better soccer player, Messi or Ronaldo?"), but other categories besides direct comparisons have been largely overlooked. Our fine-grained categorization aims to close this gap. For instance, superlative questions (e.g., "Who is the best soccer player?") can trigger a search over a group of all possible options (all soccer players) in order to find a single superior one. Sometimes, an aspect for the comparison could be explicitly stated in the question (e.g., "Who is the best soccer player when it comes to goals scored?") or not be mentioned at all which then requires some "guesswork" at the search engine side or trigger clarification. In addition, context like 'for a 4-year-old' as in "Which is better to buy for a 4-year-old, a remote control car or a table soccer?" can also further narrow down the search for an answer. This is similar to the multi-aspect dense retrieval idea by Kong et al. [100], who proposed to extend a retrieval pipeline with aspect embeddings and an aspect fusion network. Finally, a preference in a comparative question (or the absence of a preference) indicates whether the answer should explicitly mention some particular choice option along with a justification (e.g., "Which one to buy, an iPhone or a Samsung and why?") or whether providing several options along with a parallel comparison of their properties is preferred (e.g., "What are the main differences between mobile phones?").

[5] `https://www.wikihow.com/`

Table 3.4: Extended set of comparative Yandex questions with merged categories (6,250 questions in total).

Opinion/argumentative	4,101 (66%)	Preference	4,351 (70%)
Reason/factual	2,074 (33%)	Direct	3,511 (56%)
Context/aspect	1,675 (27%)	Superlative	484 (8%)
Method	332 (5%)		

3.3.1 Enlarging the Set of Comparative Questions and Refining the Question Taxonomy

The manually labeled 50,000 Yandex questions contain 1,405 comparative questions with some of the question categories containing only few examples (see Table 3.1 for the category distribution). To have a larger training set for neural approaches to classify the fine-grained categories, we thus decided to collect additional comparative questions from the Yandex log. In particular, we choose Ensemble-C+L to classify all the archived 1.5 billion questions, since it was the fastest and most accurate classifier in our experiments (approximate estimation of the runtime to classify the entire Yandex log: a few hours vs. several days when BERT is included). From the questions classified as comparative by Ensemble-C+L in the year-long Yandex question log, our annotators labeled another random 5,000 questions with the ten fine-grained categories. Since again some questions were labeled as inappropriate by our annotators and since about 1 % of the questions actually were not comparative (an expected small number of misclassifications), we ended up with a total of 6,250 comparative questions labeled with fine-grained categories.

Since some question categories were mentioned as closely related by our annotators (in fact, the annotators could assign multiple classes, and some classes then overlapped to a large extent), we decided to re-think the question taxonomy. For instance, more than 90 % of opinion questions were also labeled as argumentative questions. In discussion with the annotators, they reported believing that answers to questions like "Which to choose for vacation, Goa or UAE?" may contain both personal opinions as well as arguments supporting one or the other comparison object. We thus merge the closely related categories *opinion* and *argument*, *factual* and *reason*, as well as *aspect* and *context*, since for the merged categories also the extraction and presentation of answers will be rather similar. The distribution of the resulting seven updated categories is shown in Table 3.4.

3.3.2 Classifiers for the Comparative Question Categories

To classify comparative questions into the fine-grained categories, we use neural BERT and CNN models since they were by far more effective than the feature-based classifiers in pilot experiments (hyperparameters identical to Section 3.2.2). In particular, the BERT classifier is set up in a one-vs-rest manner (i.e., one model is trained for every category) [26], while the CNN classifier follows a multi-label approach (lower computational effort at the same effectiveness as a one-vs-rest CNN). Instead of a softmax activation, a sigmoid activation function is used.

Since the seven fine-grained categories are not mutually exclusive (i.e., questions have several class labels), and since there is no intricate result page format change at stake (whether a question is comparative will be classified beforehand), we do not treat the fine-grained classification as a precision-oriented task. Instead, we optimize the hyperparameters for the micro-averaged F1—the common practice for multi-label classification [103, 212].[6] The precision-recall curves of the classifiers on the 5,000 questions training set are shown in Figure 3.3. The largest categories of the interesting and probably also challenging to answer non-factual comparative questions (i.e., opinion/argument) can be identified very reliably, as well as whether the question asks for a preference and whether comparison objects are explicitly mentioned. In contrast, with relatively way fewer available examples, classifying whether aspects or context are mentioned seems to be hard, with the BERT-based models being slightly better than the CNN models on some question categories (however, the CNN models require less compute time).

[6] CNN: 200 filters, 4 epochs; all other CNN and BERT hyperparameters and the parameter selection procedure are identical to Section 3.2.2.

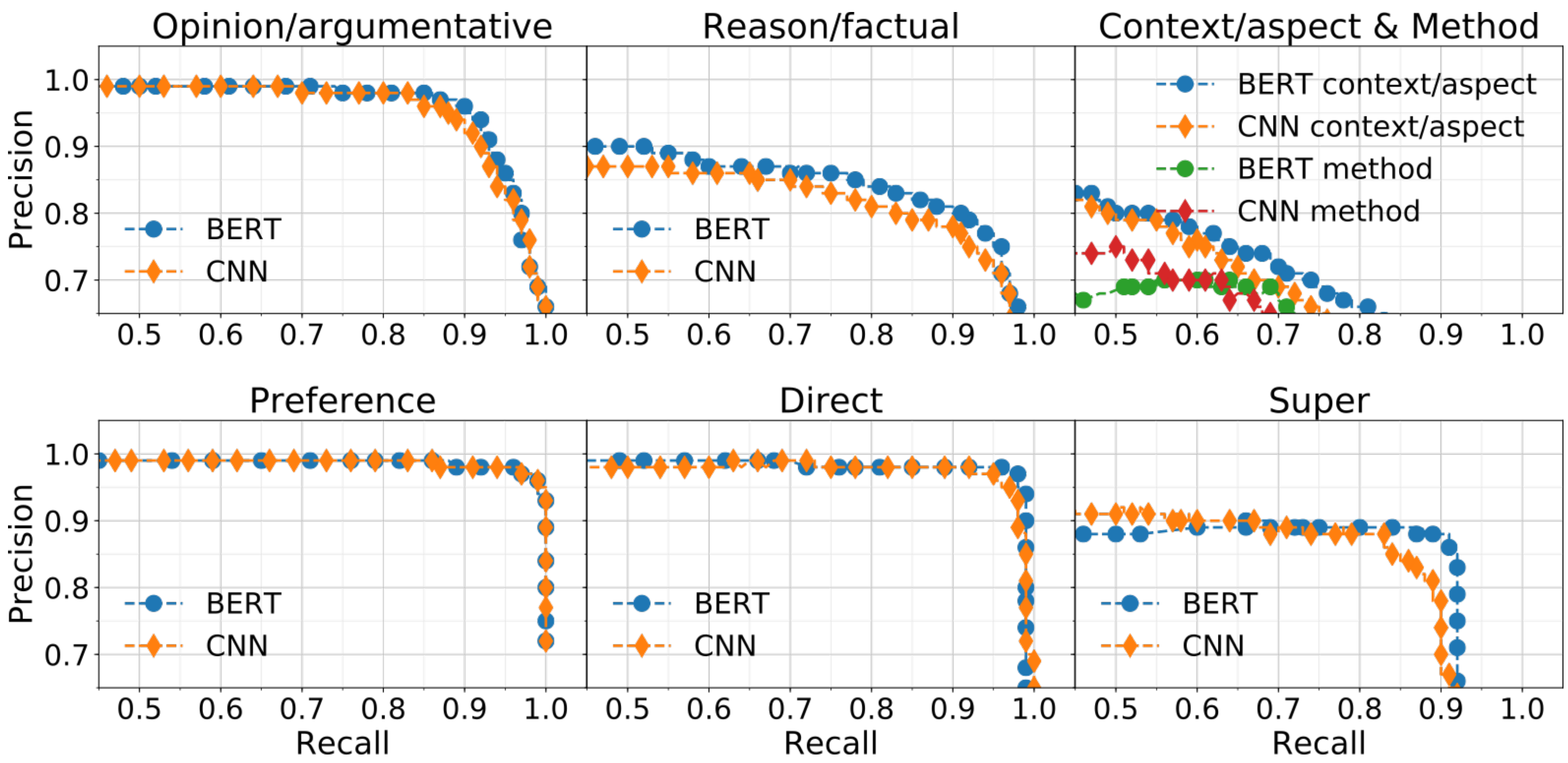


Figure 3.3: Precision-recall curves for the comparative questions' categories on the training set; 5,000 questions, ten-fold cross-validation.

Table 3.5: Results of classifying comparative question categories on the test set (1,250 Yandex comparative questions).

	CNN			BERT		
Category	Prec.	Rec.	F1	Prec.	Rec.	F1
Opinion/argumentative	0.93	0.90	0.92	0.92	0.91	0.91
Reason/factoid	0.85	0.79	0.82	0.82	0.89	0.86
Context/aspect	0.88	0.52	0.62	0.75	0.74	0.74
Method	0.79	0.80	0.79	0.75	0.82	0.78
Preference	0.97	0.98	0.97	0.96	1.00	0.97
Direct	0.95	0.96	0.96	0.95	0.98	0.97
Superlative	0.93	0.79	0.86	0.92	0.86	0.89
Micro-averaged	0.92	0.88	0.90	0.90	0.93	0.91

3.3.3 Evaluation on the Test Set

Table 3.5 reports the classification results of the neural models on the test set for the fine-grained categories of comparative questions. The respective models are trained in the settings and with the hyperparameters as described in Section 3.3.2. The three prevalent categories in the training dataset, *opinion/argumentative*, *preference*, and *direct* (each has more than 3,500 training samples) achieve the highest classification results. An exception is the underrepresented *superlative* category, which is identified relatively well by the models, probably due to the presence of adjectives and adverbs in a superlative form that are easily recognizable by the classifiers.

3.4 Analyzing Comparative Questions in the Yandex Log

We now conduct a qualitative analysis of the comparative questions in the year-long Yandex query log. This provides first insights into what comparative information needs users try to satisfy with a search engine.

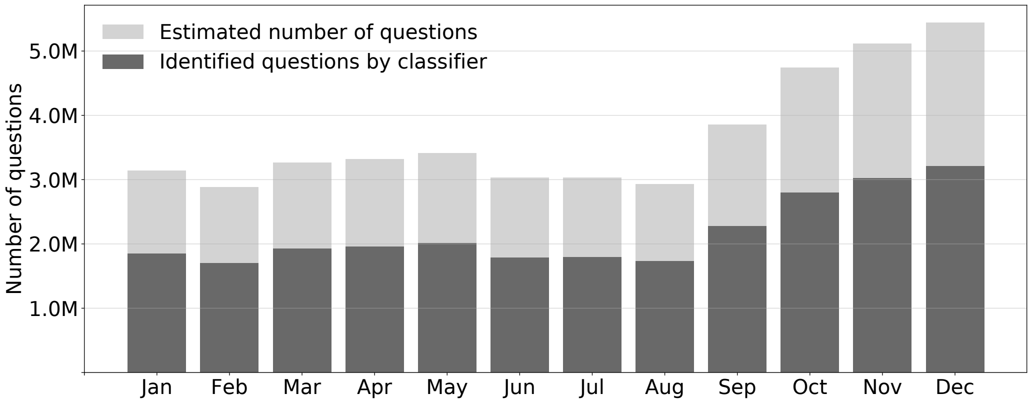


Figure 3.4: Monthly distribution of comparative questions in the Yandex log.

3.4.1 Volumes, Dynamics, and Topics

To analyze monthly distributions and seasonal effects of the comparative questions in the year-long Yandex log, we apply the (almost) "perfect precision" Ensemble-C+L on the filtered 1.5 billion questions. Figure 3.4 shows the numbers of comparative questions per month as identified by the classifier (dark shade) along with an estimated total (light shade), based on the classifier's recall of 0.59 on the test set. The estimated ratio of the comparative questions is relatively constant and is close to 3 % throughout the year (we obtained 2.8 % by random sampling) and shows an upward trend within the volume of all questions submitted to Yandex.

The comparative questions most frequently submitted to Yandex are shown in Table 3.6. A substantial part of them has the form "Which `<item>` is `better/best` to `buy/choose/watch?`" (many recalled by rule (R1)). Such questions target an informed choice, calling for opinions and arguments as pros and cons; they are hard to answer since they do not directly specify the options to be compared but require an analysis of all possible options within a set of options. Also, these questions often do not specify a comparison aspect and hence require to consider all involved items' features.

To gain further insights, we categorize the recalled comparative Yandex questions into a scheme resembling the topical categories used on the Otvety forum. Following Völske et al. [198], we use a multinomial naïve Bayes classifier trained on the Otvety data (14 merged topical categories)

Table 3.6: Ten most frequently asked comparative questions in the Yandex log.

Comparative question query	# Occur.
Which pilot was the first to surpass a supersonic speed?	176,372
Which comedy is it better/best to watch?	39,039
Which is better, Xbox or PS?	26,781
Which tablet is it better/best to buy?	24,443
Anti-radar, which one is better?	21,483
Which phone is it better/best to buy?	20,550
Which antivirus is better/best?	19,634
What is the difference between a netbook and a laptop?	18,165
Which British colony was latest to receive independence?	17,274
Which laptop is it better/best to buy?	16,775

to categorize the comparative questions in the Yandex log. The categories with the relatively frequent comparative questions (ratio to the overall amount of questions in the category) are 'consumer electronics', followed by 'cars & transportation', 'home & garden', and 'education' (cf. Table 3.7). As Figure 3.5 with the absolute numbers shows, the 'consumer electronics' category exhibits the largest increase at the end of the year—the number of comparative questions submitted in December doubles the February's number. This indicates a clear seasonal trend: People tend to purchase electronics closer to the Christmas and New Year's holidays. The Russian school summer break from June through August explains the significant drop in 'education' questions during these months, while in September and October they are asked almost as often as 'consumer electronics' questions. Most of the topical categories remain constant or undergo a decrease during the summer months, indicating a stagnation or drop in online activities during holidays and summer vacation time.

To dig deeper into seasonal patterns, we also look at changes in the most frequent questions throughout the year. In March, the question "When is it better to jog, in the mornings or in the evenings?" is not among the top-20 of the most frequently asked ones, but it jumps to rank 13 in April (probably because of the more comfortable weather conditions and the approach-

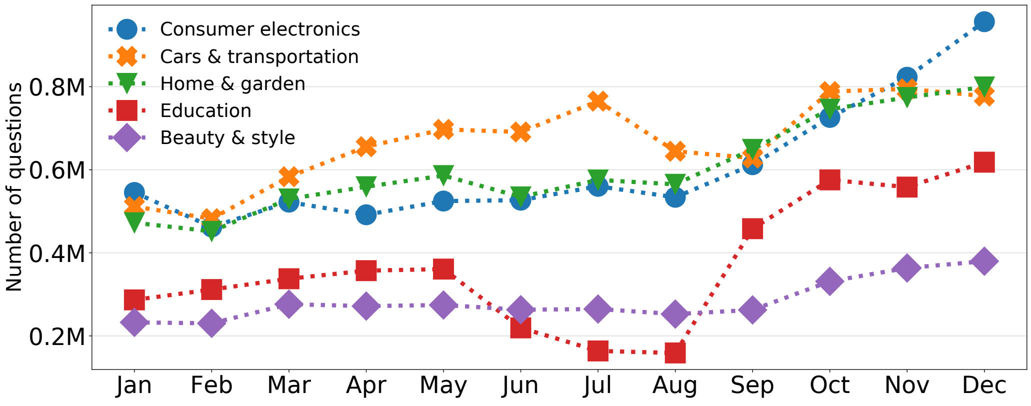


Figure 3.5: Yearly trend of the number of comparative Yandex questions in the largest topical categories.

ing summer bathing season), stays at rank 11 in May, and disappears in June. Similarly, the question "Which camera is it better/best to buy?" is the 13th most frequent question in June, then moves down to rank 17 in July, and stays at rank 18 in August. The question "What place at the Black Sea is better/best to go for vacation?" reaches rank 8 in May, moves up to rank 3 in June, goes down to rank 6 in July, and leaves the top-20 in August, coinciding with the summer vacations. The mushroom picking season is indicated by the question "How can one distinguish honey fungi from deadly skullcaps?" jumping to rank 8 in September from being out of the top-50 in August, while the approaching winter is indicated by the question "Which tires are better/best for winter?" reaching rank 7 and "Which is better, winter tire with metal studs or without?" reaching rank 12 in October from being out of the top-50 in September. Interestingly, the question "Which pilot was the first to surpass a supersonic speed?" is the most frequently asked question throughout the entire year, occupying rank 1 in every month except for January—an observation which we cannot really explain except that 2012 was the 65th anniversary of this achievement. The quite delicate questions of asking for the best ways of committing suicide (see Table 3.7) appears in January at rank 7, in March at rank 9, moves down to rank 12 in April, and disappears from the top-20 for the rest of the year. Such sensitive questions should be identified and treated appropriately by the search engine; this however is out of the scope of this dissertation.

Table 3.7: Total number of questions in millions, percentage of comparative questions, and the most frequently asked comparative questions per topical category in the Yandex log.

Topical category	Quest. mln.	Comp. %	Most frequently asked question
Consum. electronics	105.4	6.3	Which tablet is it better/best to buy?
Cars & transportation	143.7	5.2	Anti-radar, which one is better?
Home & garden	166.7	4.0	Which vacuum cleaner is it best to buy?
Education	101.8	3.9	Which pilot was the first to surpass a supersonic speed?
Beauty & style	93.7	3.3	When is it best to cut hair?
Sports	45.8	3.1	Which time of the day is most suitable for doing sports?
Family & relationships	68.5	2.7	What is the best way to commit suicide?
Health	128.2	2.4	When is it best to conceive a baby?
Adult	53.9	2.3	What is the difference between men and women friendships?
Business & finance	133.7	2.0	In which bank is it best to take a loan?
Computers & internet	152.4	2.0	Which antivirus is the best?
Society & culture	95.1	1.8	Which British colony was latest to receive independence?
Entertainment & music	90.2	1.5	Which comedy is it best to watch?
Games & recreation	122.0	1.3	Which is better, Xbox & PS?

3.4.2 Towards Answering Comparative Questions

The above insights about the comparative questions' types and their topical and temporal distribution can help search systems to better "understand" the respective information needs and, in particular, to present the answers in an appropriate way. While answers containing pro/con arguments to the most frequently asked questions could be stored in an external knowledge base, comparative questions also have a very long tail of rather rare intents. Our analysis of the compared items and the question (topical) categories shows that the comparison interests reach way beyond the traditionally studied areas of consumer products or factual questions.

Our study of the comparative web search questions reveals that more than 65 % of the questions are non-factual (cf. Table 3.4) and demand argumentation and opinions in an answer (e.g., "Which is better, Xbox or PS?" or "How are dogs better than cats?"). One possible approach to tackle such questions is to extract "ready-to-use" answers from question-answering fora. To test how well such an extraction approach might work, we index the cleaned set of all 5.5 million Otvety questions that have selected "best answers" with Elasticsearch (BM25 as retrieval model). The 4,101 comparative Yandex questions labeled as opinion/argumentative are then used as search queries against this index (stop words removed). Our human assessors then labeled the answer to the top-ranked Otvety question as relevant or not for the Yandex question. It turns out that for about 48 % of the comparative opinion/argumentative questions submitted to Yandex the top-ranked Otvety questions with the best answer are relevant. The conclusions that can be drawn from this finding are two-fold. On the one hand, a substantial amount of non-factual comparative questions cannot be "simply" answered by retrieving a "ready-to-use" answer from question and answer fora (note that this assumption should be taken with a grain of salt since our experiments are limited to a single search engine log and a single forum archive). On the other hand, to avoid a particular standpoint bias in the results, answers to non-factual questions should provide diverse points of view [6], making retrieval of one answer from some forum infeasible. These ideas prepare the ground for the argument retrieval methodology discussed in detail in Chapter 6 as part of the overall workflow for answering comparative questions.

3.5 Identifying English Comparative Questions

In the following section, we describe the classification approach to identifying English comparative questions following the ideas applied to the Russian questions that are described in Section 3.2. Thinking of a potential switch to a comparative result interface of search systems, we follow the same idea of a precision-oriented classifier and combine handcrafted rules with feature-based and neural classifiers.

To classify comparative questions, we start with labeling English questions, then we handcraft high-precision rules and subsequently apply feature-based classifiers, as well as more recent BERT variants like RoBERTa [118], ALBERT [107], SBERT [154], and BART [112]. We then combine predictions of these different classifiers in a cascading ensemble. The main reason for using a different set of neural network architectures is that the experiments on the English questions were conducted approximately two years later after we had proposed classifiers for their Russian counterparts, and new, more effective models emerged.

3.5.1 Data Annotation

We randomly sample a dataset of 31,000 English questions from the MS MARCO [134] and Google Natural Questions [106] datasets (these two data sources contain question-like queries submitted to the Bing and Google search engines) as well as questions asked on Quora [84], Yahoo! Answers,[7] and Stack Exchange fora.[8] These questions represent genuine real-life information needs that people express on the Web.

In a pilot kappa-test training phase, our three volunteer annotators labeled the same 150 randomly sampled questions as comparative or not. The annotators achieved a Fleiss' κ of 0.51 (moderate agreement). After

[7] http://webscope.sandbox.yahoo.com

[8] https://archive.org/details/stackexchange

(R1) [what|which] ∧ [is|are] ∧ SUPER

(R2) COMP ∧ [the best] ∧ ¬[or|vs|versus] ∧ ¬[for the best|how is]

(R3) [what|which] ∧ [is|are] ∧ [difference|differences|pros|good|
advantages|better|similarities]

(R4) [difference between|compare to]

(R5) [distinguish|differ|differentiate|difference(s)?|strengths
|weaknesses] ∧ [or|and|from|between|vs|and|versus] ∧ ¬[how]

(R6) [(which)? is|are] ∧ [a|an|the] ∧ COMP|SUPER

(R7) [what are (some)? good]

(R8) [who was (the|a|an)?] ∧ COMP|SUPER

(R9) [which] ∧ [should i]

(R10) SUPER posn$_1$ ∨ [the|a|an] posn$_1$ ∧ SUPER posn$_2$

Figure 3.6: Ten lexico-syntactic rules to classify English questions as comparative or not.[9]

discussion and refining annotation guidelines, the annotators individually labeled distinct question subsets (i.e., one vote per question), resulting in a total of 3,500 comparative questions.

3.5.2 Rule-Based Classification

Following the idea of the rule-based classification (cf. Section 3.2.1), we handcrafted rules on an 80 % subset of the labeled questions, such that each rule should identify comparative questions with a perfect precision of 1.0 (see Figure 3.6). In a pre-processing step, we remove punctuation and POS-tag the questions using the neural Stanza tagger [148]. Our potential 10 rules consist of regular expressions over question tokens, comparative (COMP) (Penn Treebank tags: JJR and RBR [167]) and superlative (SUPER) tags (Penn Treebank tags: JJS and RBS), token positions (posn$_n$, e.g., in the rule (R10), the expression SUPER posn$_1$ means that a superlative adjec-

[9] Expressions in [] are in a regular expression syntax: so, a question matching (R1) must contain the tokens *what* or *which* and *is* or *are* and an adjective or an adverb in a superlative form.

Table 3.8: Effectiveness of the rules on the 80% training set (English questions). Reported are recall for each rule and cumulative recall by applying the rules sequentially. Precision of classifying the comparative question class is always 1.0.

Rule	r1	r2	r3	r4	r5	r6	r7	r8	r9	r10
Recall (rule)	0.35	0.23	0.11	0.09	0.08	0.06	0.03	0.01	0.01	<0.01
Recall (cumul.)	0.35	0.38	0.48	0.50	0.51	0.52	0.53	0.54	0.55	0.55

tive or adverb must be the first token in the sequence of question tokens), and logical operators (AND $\wedge$, OR $\vee$, and NOT $\neg$). When creating the rules, we also noticed that all the rules can be systematically violated by a limited list of words. Thus for every rule, we additionally check that the following holds: A rule does not classify a question as comparative if (1) a question starts with 'how long', 'how many', or 'how much' or (2) it contains at least one indicator for song lyrics, movie names, etc. from the following list of words: lyrics, wrote, mean, cover(s), covered, cast, play(s), played, season, episode, award, sing(s), sang, song, album, movie. Our 10 handcrafted rules shown in Figure 3.6 all achieve a precision of 1.0 at classifying comparative questions and are ordered by a descending recall. Note that the rules' classification decisions may partially overlap (i.e., a question can be classified as comparative by more than one rule).

For example, the rule (r6) from our set classifies a question as comparative if it contains a comparative or superlative adjective or adverb and the optional term `which`, auxiliary verbs `is` or `are` and the articles (e.g., "_Is_ _a_ cat or _a_ dog a better_JJR friend?"). Table 3.8 reports a recall for each rule on the train set separately and a cumulative recall by applying the rules sequentially, e.g., (r1) alone recalls 35 % of the comparative questions, whereas a combination of (r1)–(r3) recalls 48 % of the questions (precision is always 1.0). Since we do not gain any further reasonable recall with (r10), we do not consider adding any new rules. Combining all the 10 rules, a question is classified as comparative when at least one rule classifies it as comparative. This yields a recall of 52 % at a precision of 1.0 on the 20 % of our dataset not used to handcraft the rules (three points less than on the train set) and a recall of 54 % when applied on the full dataset as a first step in the cascading ensemble (cf. Table 3.9 (a)).

3.5.3 Feature-based Classifiers

To further increase the recall, we experiment with feature-based classifiers applied after the rules: logistic regression, naïve Bayes, and SVM. The classifiers are trained and evaluated in a 10-fold cross-validation on those questions of the full dataset that the rules do not classify as comparative. The underlying rationale is that, in the practical application, any classifier after the rules will never see comparative questions that the rules detect. Instead, the more "sophisticated" classifiers are meant to identify the more "difficult" comparative questions.

Among the feature-based classifiers, logistic regression was by far the most effective; we used a grid search to select the features (tf or tf-idf weighted word or lemma n-grams, and combined with POS-tags), and to optimize the hyperparameters, as well as the probability threshold of the precision-optimized operating point.[10] Adding the best configuration[11] as a cascade step after the rules improved the recall to 62 % at a precision of 1.0 (cf. Table 3.9 (a)).

3.5.4 Neural Classifiers

Afterwards, we experiment with neural classifiers on the questions not classified as comparative by the rules or the ones remaining after the logistic regression—to improve the recall on the "most difficult" comparative questions. In a 10-fold cross-validation setup, the transformer-based classifiers BERT, RoBERTa, and ALBERT running at perfect-precision operating points only achieve recall values of at most 1 % on the questions remaining after the rules or the logistic regression. Since this does not really help to increase the overall recall, we thus further experimented with pre-trained

[10] Probability found by gradually decreasing the decision threshold starting from 1.0.

[11] Logistic regression: tf word 4-grams; C=48, penalty="l2", solver="liblinear"; thresh=0.9037. The parameter ranges in a grid search are identical to Section 3.2.2; exception C-values for log. regression: {40. . .100; step 1}.

Table 3.9: Effectiveness of classifying English questions as comparative or not. (a) Aggregated recall of our 7-step cascading ensemble (full dataset; 10-fold cross-validation; precision always is 1.0; probability thresholds for the perfect precision operating points given in the column "Thresh."). (b) Effectiveness of individual classifiers on the full dataset (10-fold cross-validation); if a classifier has no perfect precision operating point, the given probability threshold indicates the 0.95-precision operating point. Subscripts: base (`B`) or large (`L`) pre-trained model, `CLS`-token (`C`) or mean (`M`) of all token-embeddings.

(a)

Cascade step	**Thresh.**	**Rec.**	**F1**
Rules		0.54	0.70
10-fold trained on questions remaining after the rules			
Logistic regr.	0.9037	0.62	0.76
RoBERTa$_{\text{BC}}$	0.9881	0.63	0.77
BART$_{\text{LM}}$	1.0	0.66	0.80
10-fold trained on questions remaining after logistic regr.			
SBERT$_{\text{LM}}$	1.0	0.67	0.80
BART$_{\text{LM}}$	1.0	0.69	0.82
Final averaging step	0.89	0.71	0.83

(b)

Classifier	**Thresh.**	**Prec.**	**Rec.**	**F1**
Logistic regr.	0.916	1.0	0.45	0.62
Embedding-based				
SBERT$_{\text{LM}}$	0.9637	0.95	0.68	0.79
RoBERTa$_{\text{BC}}$	0.769	0.95	0.67	0.79
BART$_{\text{LM}}$	0.9146	0.95	0.66	0.78
Fine-tuned				
ALBERT	0.995114	0.95	0.87	0.91
BERT	0.999929	0.95	0.62	0.75
RoBERTa	0.99988	0.95	0.44	0.60

transformer models to only create representations and trained a logistic regression and a feedforward deep neural network (DNN) on the embeddings. The best DNN configuration[12] performed better than any logistic regression setup, such that we decided on DNN. As representations, we used CLS-token embeddings and the mean of all token embeddings [159].

The following classifiers achieved a recall of at least 5 % at a precision of 1.0 and were thus added as further cascade steps (see Table 3.9 (a)): (1) On the questions remaining after the rules: (a) RoBERTa (base model, CLS-token embeddings, DNN; 5 % recall), and (b) BART (large model, pre-trained on the news summarization dataset, mean of all token embeddings, DNN; 11 % recall); (2) on the questions remaining after logistic regression: (a) SBERT (Sentence-BERT with Siamese BERT Networks; large model, mean token embeddings, DNN; 5 % recall), and (b) BART (configured as above; 12 % recall). Extending the cascade with the above classifiers in the given order improved the aggregated recall to 69 %.

3.5.5 Final Cascading Combination of Classifiers

To further improve the recall after the above steps (rules, logistic regression, neural), we add a final step to the cascade that gets as input the queries not identified as comparative after the second BART_{LM} classifier. As its decision criterion, the final step simply averages the decision probabilities of the logistic regression and the embedding-based classifiers, and 10-fold cross-validates yet another decision threshold to recall some further comparative questions at a perfect precision. With this final step, the whole cascade achieves an overall recall of 0.71 at a still perfect precision of 1.0 (cf. Table 3.9 (a) for the complete 7-step cascade).

3.5.6 Individual Classifiers on the Full Dataset

To also support scenarios where the complete cascade may be too costly for identifying comparative questions, we also evaluate less expensive in-

[12] DNN: 3 hidden layers with output units: 256, 64, 16, activation="relu", epochs=100 with early stopping, batch size=5, loss="binary_crossentropy", optimizer="adam", optimization metric: "true positives". Tested configurations: hidden layers: $\{1 \ldots 5; \text{step } 1\}$.

dividual classifiers on the full dataset in a 10-fold cross-validation setup. As most classifiers cannot recall many comparative questions at a precision of 1.0 (except for logistic regression with a recall of 0.45), we set their operating points to a precision of 0.95. The results in Table 3.9 (b) show that among the embedding-based and fine-tuned models, the ALBERT-based classifier[13] is most effective, recalling 87 % of the comparative questions at a precision of 0.95. Applying our simple rule set to the questions not identified as comparative by the ALBERT-based classifier can further slightly increase the recall up to 88 %.

3.6 Summary

The first contribution of this chapter is the datasets of Russian and English questions fetched from the search engine and question and answer fora archives that were manually annotated as comparative or not. To distinguish comparative questions from others, we have trained high-precision classifiers. We have shown that using handcrafted rules that are based on the syntax of comparison structures, we can reliably identify about 50 % of comparative questions. Subsequently adding feature-based and neural classifiers trained on more difficult examples that are not captured by the rules increases the recall to over 60 % for Russian and to over 70 % for English comparative questions at perfect precision, however increasing the computational cost at the same time. Our proposed stepwise cascading combination of different classifiers allows for a flexible configuration by using only its parts, which depends on the specific requirements and computational resources. We presume that the classification effectiveness can be further improved by designing new rules and enlarging training data.

To explore what kinds of comparative questions (and how frequently) are asked on the Web, we have studied comparative questions submitted to Yandex over the period of one year. We found that comparative questions constitute a non-negligible portion of the questions that Yandex received (about 3 %). By analyzing a random sample of about 3,000 comparative

[13] ALBERT, BERT, and RoBERTa: large model, learning rate=0.00002, epochs=10, batch size=8, max sequence length=64.

questions from Yandex and Otvety, we proposed a taxonomy that represents different comparative information needs. Our study showed that these information needs reach far beyond just comparing products to buy or just expecting simple facts as answers (more than 65 % of the comparative questions are clearly non-factual). Moreover, we found that at least half of comparative questions do not have explicitly mentioned comparison objects and thus are ambiguous. These findings motivate the investigation of clarification approaches to refine initial unclear questions and to study argument retrieval and analysis approaches (e.g., argument quality estimation and stance detection) that we address in the subsequent chapters.

4
Parsing Comparative Questions and Answer Stance Detection

In the previous chapter, we analyzed comparative questions in a yearlong search engine log to gain first insights into what kinds of such questions users submit to search engines and elaborated on possible reactions of a search system to these different question types. We found that at least 3 % of the questions submitted to search engines can be comparative (a non-negligible amount), ranging from simple factual ones like “Did Messi or Ronaldo score more goals in 2022?” to life-changing and probably highly subjective questions like “Is it better to move abroad or stay?”. We also found that about half of all comparative questions fall into the category of non-factual comparison requests (often called *subjective* questions [27, 51, 114]). Moreover, we suggested that answers to subjective comparative questions should ideally show diverse opinions so that the searchers can come to a well-informed, less biased decision [6, 169]. Thus, search result presentation for comparative questions could be in the form of showing side-by-side different facts, opinions, or arguments for and against the comparison objects. However, to put some opinion or argument on the “correct” side in such a result, comparison objects (stance targets) and the stance of the respective text passages need to be determined.

First, this chapter investigates the methodology for parsing comparative questions to better understand their underlying information needs by identifying important question constituents like the mentioned comparison objects, comparison aspects, and predicates. For example, a question like "Is a cat or a dog a better friend?" should be identified as comparative and non-factual. The terms 'cat' and 'dog' should be tagged as the comparison objects, 'friend' as the aspect, and 'better' as the predicate. Then, an answer candidate like "Cats can be quite affectionate and attentive, and thus are good friends" should be classified as pro the 'cat' object, while "Cats are less faithful than dogs" as supporting the 'dog' object. Such question parsing and result analysis will allow the formulation of an answer that covers diverse opinions. Instead of a short, direct answer extracted from a single source, search engines might benefit from extracting and analyzing diverse points of view for non-factual comparative questions. They might even change the result presentation by combining and highlighting several pros and cons towards the compared objects. In doing so, the detected comparison aspect(s) indicate whether a particular objects' property should be emphasized when searching for potential result nuggets on the Web, while the predicate(s) guide the direction of the answer composition (e.g., whether a better or worse option should be presented).

An early proposed solution for comparative information needs was "comparative web search" [191]: a web interface to submit each comparison option as a separate keyword query to compare the retrieved "ten blue links" results side-by-side. Recently, a slightly more sophisticated search system to tackle comparative information needs: a comparative argumentative machine, has been proposed [169]. However, it cannot process comparative questions but expects the user to enter in separate boxes the options to be compared along with the comparison aspects. An important step towards actually showing a pro/con result presentation for comparative *questions* would be the identification of the question components.

We begin with the description of data preparation: in Section 4.1, we present a corpus of 3,500 English comparative questions that were manually labeled with the comparison objects, aspects, and predicates on the token level. For 950 comparative questions, we also collected "best answers" from question and answer fora and annotated whether the answer stance is neutral or pro first/second object, or whether no stance is entailed. Our respective classifiers are trained and tested on these annotations.

To tag the comparison objects, aspects, and predicates in comparative questions, we develop a token-level classifier based on the RoBERTa model [118] (cf. Section 4.2). To further improve its effectiveness, we propose to pre-classify comparative questions as 'direct' and 'indirect' before the actual object tagging and 'with an aspect' before the aspect tagging.

Further, this chapter addresses subjective comparative questions and focuses on detecting the stance of potential textual answers that is described in Section 4.3. To detect the stance towards the comparison objects, we fine-tune RoBERTa and Longformer models [19] that predict one of the four stance classes: 'pro first object', 'pro second object', 'neutral', and 'no stance'. Our best sentiment-prompted RoBERTa-based stance detector achieves an accuracy of 0.63 and leaves room for future improvements. In a post hoc evaluation, we also used GPT-3 [41] for stance detection, which achieved a slightly higher accuracy of 0.65.

Finally, Section 4.4 concludes this chapter and discusses the implications of the contributions, and elaborates on open questions and future work.

4.1 Data Annotation

In the dataset of 31,000 English questions, we manually labeled 3,500 instances as comparative (cf. Section 3.5). We further label the comparison objects, aspects, and predicates in the comparative questions, and whether the question is rather factual or subjective (asks for opinions/arguments). For 950 questions, we also label the answer's stance towards the question's objects. Table 4.1 shows the characteristics of our labeled data.

Before the labeling task, we rethink the question taxonomy described in Chapter 3 and focus on the categories that are most important for question parsing and stance detection. In Section 3.3, we already justified the merge of some closely-related categories. For instance, from the application perspective, both opinion and argumentative questions require some subjective viewpoint on the matter like comparison options, and thus po-

Table 4.1: Characteristics of our dataset. (a) Subtypes of comparative questions with frequencies. (b) Number of tokens in comparative questions labeled as objects, aspects, and predicates. (c) Number of answer stance labels.

(a) (31,000 questions)

Type	#
Comparative	3,500
- Subjective	1,690
- With aspect	1,435
- Direct	1,470

(b) (3,500 questions)

Token	#
Object	14,480
Aspect	4,594
Predicate	3,822
None	14,765

(c) (950 questions)

Stance	#
Pro Object1	322
Pro Object2	274
Neutral	285
None	69

tential answers might need stance detection. Hence, we propose to distinguish between *factual* comparative questions and non-factual—often called *subjective* questions [27, 51, 114]. The answer to a subjective comparative question, e.g., "Is a cat or a dog a better friend?" is not settled and will contain opinions or arguments that support one or another comparison object or both. Whereas the answer to a factual question, e.g., "Did Messi or Ronaldo score more goals in 2022?" is fixed. Similarly, a comparison aspect and a context, both specify some qualities of the comparison objects over which they should be compared. For instance, for the comparison "Is a cat or a dog a better friend?", it is important to compare cats with dogs specifically using their quality of being a friend. We thus distinguish between comparative questions with and without a comparison *aspect* (a question without an aspect is, e.g., "Is a cat or a dog better?"). And finally, we distinguish between questions that explicitly mention the comparison objects, e.g., 'dog' versus 'cat', and questions that refer to a larger group of potential to-be-compared options like "What pet is the best?". We thus propose the categories of *direct* and *indirect* comparative questions. Note that direct comparative questions and questions with aspects are part of the taxonomy described in Section 3.3. Later in Section 4.2, we show that this subcategorization helps to improve question parsing.

For labeling, we recruited three volunteers, grad and undergrad computer science students, two of whom had a background in linguistics. Our annotation guidelines are grounded in linguistic research, opinion mining, and information retrieval. As for the comparison objects (linguists often call them comparands [10, 188]), we follow the common approach of previous opinion mining and information retrieval studies [85, 137, 169] and consider any lexical items that are intended to be compared when mentioned in a comparative question, including products, named entities, verbal or noun phrases, etc. For example, in the question "Is a cat or a dog a better friend?", the terms 'cat' and 'dog' are the first and second comparison objects, respectively. Comparison relations between the objects are established by predicates [10] (e.g., the term 'better' in the example). Finally, from a psychological perspective, a comparison is considered as contrasting the common and distinctive features, or attributes, of some items [197]. In opinion mining and information retrieval, these features have various names: comparison points [12], comparison attributes [71], features [85, 86], or more often—aspects [14, 109, 169]. In our guidelines for the labeling, we follow the aspect terminology and label an aspect of a comparison as the objects' shared property over which the objects should be compared (e.g., the term 'friend' in the example). Finally, we instructed the annotators to distinguish factual comparisons that can be answered from some "standard" knowledge base from the comparisons that need more textual elaboration (i.e., opinions and arguments).

In a pilot annotation phase, we let our three annotators label the same 150 randomly sampled questions. The annotators achieved a Fleiss' κ=0.57 (moderate agreement) for labeling the objects, κ=0.73 for the aspects (substantial), κ=0.62 for the predicates (substantial), and κ=0.87 for factual vs. subjective comparative questions (almost perfect). After discussions and refining the annotation guidelines, the annotators individually labeled distinct question subsets. The labels '(in)direct' or 'with(out) aspect' were inferred from the annotated objects and aspects.

For stance detection, we sampled another 950 questions from archives of Yahoo! Answers and Stack Exchange where a "best" or "accepted" answer of at least ten words is selected. Since our focus is answers to non-factual comparisons, for sampling, we fine-tuned a BERT-based classifier [59] to identify subjective comparative questions. For fine-tuning, we translated 1,400 Russian comparative questions using the Yandex Translate API[1] that were already labeled as subjective (opinion/argumentative) or factual (cf. Section 3.1). We manually removed misclassified questions and kept only those that contained two comparison objects until we had sampled 1,000 such questions. We manually cleaned the answers and removed 50 questions in this process that did not have meaningful answers. The remaining 950 answers are on average 138 words long. We replaced HTML characters with ASCII equivalents and replaced links with a [REF] placeholder. For diversity, we ensured sampling from the domains such as academia, computer science, gardening, music, cooking, software engineering, software recommendations, computers, and traveling.

In a pilot phase for the answer stance annotation, the three annotators labeled 120 answers with respect to the comparison objects mentioned in the questions as (a) pro first object (answer expresses a stronger positive attitude towards the first object using a predicate like 'better'), (b) pro second object (stronger positive attitude towards the second object), (c) neutral (both comparison objects are equally good or bad), and (d) no stance (no attitude / opinion / argument towards the objects is entailed). The annotators achieved a Fleiss' κ=0.61 for the stance labels (substantial agreement). After discussing the annotations and refining the guidelines, each annotator labeled a subset of the remaining answers individually. In total, the answers have almost equal ratios of 'pro first object' (34 %), 'pro second object' (29 %), and 'neutral' (30 %), and only a small fraction of 'no stance' labels (7 %). The annotation results are shown in Table 4.1.

4.2 Parsing Comparative Questions

To better understand comparative questions (i.e., what objects should be compared over which aspects), we develop token-level classifiers that identify the important terms in comparative questions. We first experiment with a multi-class token classifier for the comparison objects, aspects, and predicates. To further improve the classification effectiveness, we train

[1] https://yandex.com/dev/translate/

Table 4.2: Per-class effectiveness of identifying comparison objects, aspects, predicates, and other tokens (NONE) using a multi-class classifier.

	BiLSTM			**RoBERTa**		
Token	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
OBJ	0.74	0.84	0.80	0.92	0.93	**0.93**
ASP	0.64	0.44	0.52	0.81	0.80	**0.80**
PRED	0.86	0.82	0.85	0.97	0.99	**0.98**
NONE	0.98	0.98	**0.98**	0.95	0.93	0.94

separate binary classifiers for each token class and propose to pre-classify questions as direct or indirect comparisons and as with or without aspects.

4.2.1 Multi-Class Token Classification

So far, studies on detecting the objects, aspects, and predicates in comparative sentences [14, 48, 85, 86, 91, 92, 113] only considered cases of exactly two explicitly mentioned objects. Differently, besides direct questions that explicitly mention the intended comparison objects ("Is a cat or a dog a better friend?"), we also address indirect questions that just mention a general concept (e.g., "Which pet is the best friend?"). Our classifiers will tag each token in a question as an object, aspect, predicate, or none (somewhat similar to POS tagging). In 10-fold cross-validation pilot experiments, we compared a one-layer BiLSTM baseline classifier with 300-dimensional GloVe embeddings [14] with several fine-tuned transformer models pre-trained for token classification: BERT [59], ALBERT [107], RoBERTa [118], and ELECTRA [54]—RoBERTa performed best.[2] The results in Table 4.2 show that the BiLSTM baseline is more accurate at classifying the 'none' tokens while the fine-tuned RoBERTa is more accurate at identifying the classes of interest—predicates (almost perfect F1 of 0.98), objects (F1 of 0.93), and aspects (F1 of 0.80). We thus further experiment with RoBERTa.

[2] RoBERTa: large, learning rate=0.00003, epochs=10, batch size=8, max seq length=64.

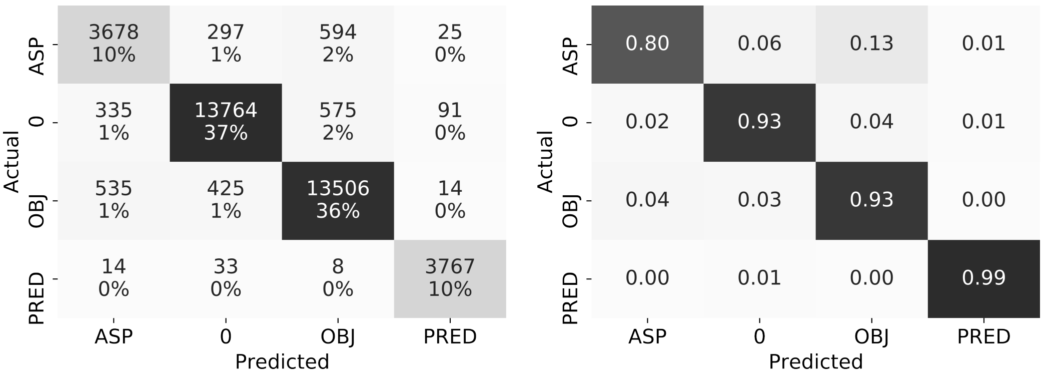


Figure 4.1: Confusion matrices: without normalization (left) and with normalization by a class support size, i.e., the number of elements in each class (right).

Error Analysis

Figure 4.1 shows a confusion matrix and its normalized version by a class support size of our multi-class RoBERTa token classifier. Below, we provide a few examples of the most common errors of confusing the comparison objects and aspects. Consider an example question "Which is the best [OBJ: online platform] for [ASP: information about import and export data in Malaysia]?", in which 'online platform' is labeled as an object and a whole noun phrase after 'for' as an aspect. The classifier, however, predicts the aspect-part as an object. This error might occur since often in this type of questions, the objects are whole noun phrases that follow after the 'best'-token like in "Which is the best [OBJ: washing machine brand in India]?". The classifier tags the latter example correctly. One possible solution to tackle such errors one might think about is to use rules in the post-processing step. For instance, by tagging noun phrases after 'for' as aspects. However, some compound aspects can contain 'for' inside them like in "Which [OBJ: book] is best to [ASP: use for preparing for a math exam]?". In another example, 'for' is part of a labeled object: "What are the best [OBJ: technologies for mobile phones]?". The classifier incorrectly tags 'mobile phones' as an aspect. Still, developing rules for post-processing to fix classification errors is an interesting avenue for future work.

Table 4.3: Effectiveness of RoBERTa classifiers trained for each class separately on: (a) full set of comparative questions; (b) subsets of (in)direct questions for object identification, and on questions with aspects for aspect identification.

(a)

Token	Prec.	Rec.	F1
OBJ	0.93	0.94	0.93
ASP	0.83	0.77	0.80
PRED	0.97	0.98	0.98

(b)

OBJ	Prec.	Rec.	F1
Direct	0.94	0.95	0.95
Indirect	0.92	0.93	0.92
ASP	**Prec.**	**Rec.**	**F1**
With ASP	0.90	0.90	0.90

4.2.2 Per-Class Token Classification

In the attempt to improve the identification of the comparison objects and aspects, we fine-tune RoBERTa-based classifiers[3] for each token class separately in a 10-fold cross-validation. The results in Table 4.3 (a) show that the individual classifiers do not really improve upon the multi-class variant. To still achieve a better classification effectiveness, we experiment with a two-step procedure: first, classifying a question as direct or indirect (i.e., mentioning concrete comparison objects or only a general concept), and classifying whether a question contains an aspect or not, and only then tagging the objects or aspects with individual classifiers for these subclasses. The hypothesis is that separate object taggers for direct and for indirect questions, or an aspect tagger only for questions that actually contain aspects, will be more effective. To test this hypothesis before developing the actual classifiers for the first step, we simply use the respective manual labels to simulate perfect "oracle-style" classifiers. We fine-tune and evaluate RoBERTa-based binary taggers with the same hyperparameters as before. The results in Table 4.3 (b) show that the object identification indeed benefits for direct questions (F1 gain of 0.02) but is almost unchanged for indirect questions. Not too surprisingly, identifying aspects in questions that actually contain aspects yields a large F1-increase of 0.1. These possible gains show that developing actual classifiers to replace the "oracle"

[3] RoBERTa: large, learning rate=0.00002, epochs=10, batch size=8, max seq length=64.

CONJ = [or|vs|versus|between|from|over|and|than]

(a) Rules to classify *indirect* comparative questions:

(R1) SUPER ∧ ¬ CONJ

(R2) [who] ∧ [first] ∧ ¬ CONJ

(R3) [what are good|the top|what are some|advantages of|benefits of] ∧ ¬ CONJ

(R4) [advantages and disadvantages of] ∧ ¬ CONJ

(b) Rules to classify *direct* comparative questions:

(R1) COMP ∧ CONJ

(R2) [best|first] ∧ [or]

(R3) [different|difference(s)?] ∧ CONJ

(R4) [advantages and disadvantages] ∧ CONJ

(R5) [same|similar|compare] ∧ CONJ

(R6) distinguish|replace ∧ CONJ

Figure 4.2: Ten lexico-syntactic rules to classify English comparative questions as direct or indirect.[4]

from the pilot experiments with actual classifiers for direct/indirect comparisons and questions with/without aspects is a worthwhile effort.

4.2.3 Comparative Question Pre-Classification

In our dataset, direct comparative questions often contain the object separators like 'or', 'vs', etc., such that we handcraft high-precision rules: six rules for direct and four rules for indirect questions. For instance, if a comparative question contains a comparative adjective or adverb and a separator like 'or', 'vs', etc., the question is direct (rule set is in Figure 4.2).

[4] Expressions in [] are in a regular expression syntax: so, a question matching (R1) (indirect) must contain an adjective or an adverb in a superlative form and must not contain any conjunction from CONJ.

Table 4.4: Effectiveness of classifying comparative questions as direct or indirect.

	Direct			Indirect		
	Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
Rules	1.0	0.76	0.87	1.0	0.63	0.77
RoBERTa	0.99	0.99	0.99	0.99	0.99	0.99

Additionally, we fine-tune RoBERTa[5] in a 10-fold cross-validation setup. The results in Table 4.4 show that the rules recall 76 % of the direct and 63 % of the indirect comparative questions with a precision of 1.0. However, RoBERTa achieves a near-perfect F1 of 0.99 that might be difficult to further improve—combination with the rules yields no improvement.

As for the questions with aspects, we did not observe any prominent lexical cues in our dataset that could be used in a rule-based approach. We thus experiment with the same feature-based and neural classifiers used for classifying comparative questions (cf. Section 3.5) in a 10-fold cross-validation setup. Table 4.5 shows the results of the three most effective approaches: RoBERTa (F1 of 0.84 for questions with aspects and 0.90 without) followed by logistic regression and a DNN trained on the RoBERTa-embeddings (RoBERTa$_{\text{LC}}$: large model with CLS-token embeddings).[6]

We also experimented with two high-precision ensembles for the two classes (cf. ENSEMBLE$_{\text{PREC}}$ in Table 4.5), including predictions of BERT and ALBERT (same hyperparameters as RoBERTa). For each classifier, we select the operating points via the probability thresholds so that they each have a precision of 1.0 for the respective class. The predictions of the individual classifiers are averaged similarly to the last step of the cascade described in Section 3.5.5. As a result, the ensembles recall 16 % of the questions with comparison aspects and 12 % of the ones without at perfect precision. Further improving the recall of the ensembles might be a promising

[5] RoBERTa: large, learning rate=0.00002, epochs=10, batch size=8, max seq length=64.

[6] RoBERTa: same hyperparameters as for the (in)direct questions. Logistic regression: representation: tf lemma 1–4-grams, C=0.0002637, penalty="l2", solver="liblinear". DNN: 3 hidden layers with output units: 256, 64, 16, activation="relu", epochs=100 with early stopping, batch size=5, loss="binary_crossentropy", optimizer="adam", optimization metric: "accuracy".

Table 4.5: Effectiveness of classifying comparative questions as with or without comparison aspects.

	With ASP			Without ASP		
	Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
RoBERTa	0.85	0.84	0.84	0.89	0.90	0.90
Logistic regr.	0.86	0.74	0.80	0.84	0.92	0.88
$\text{RoBERTa}_{\text{LC}}$	0.81	0.73	0.77	0.83	0.88	0.86
$\text{ENSEMBLE}_{\text{PREC}}$	1.0	0.16	0.28	1.0	0.12	0.22

direction for future work. Still, already the current versions might be helpful in systems that can ask clarifying questions (cf. Chapter 5), when the classifiers are not sure whether an aspect is contained.

4.3 Detecting Answer Stance

To allow answering subjective comparative questions asking for opinions and arguments using more diverse viewpoints (pro, con, neutral) in the answers, we experiment with classifiers that identify such questions and that detect the stance of potential answers. In our pilot experiments, we tested several transformer models and found that RoBERTa [118] and Longformer [19] are the most effective for these tasks.

To detect the answer stance towards the comparison objects, we evaluate several transformer-based classifiers. As inputs, we experiment with only answers or pairs of questions and answers. Since stance detection requires explicit targets (comparison objects in our case), we focus on direct comparative questions (each question in our 950 annotated question–answer pairs has exactly two comparison objects). Additionally, we experiment with masking the comparison objects in questions and answers with special placeholders (objects manually labeled by our annotators). Our experiments show that object masking helps the classifiers to better learn textual stance cues regardless of the concrete objects.

4.3.1 Identifying Subjective Questions

To distinguish subjective comparative questions (e.g., "Is a cat or a dog a better friend?") from factual ones (e.g., "Do cats live longer than dogs?"), we fine-tune RoBERTa[7] (most effective in our pilot experiments) in a 10-fold cross-validation setup on our labeled dataset (initial F1 of 0.93 on both classes). Since subjective questions are the main target of the answer stance detector, we then select the operating point to maximize the precision on this class while keeping the maximum possible recall. The classifier with the best precision–recall trade-off (threshold=0.999927) recalls 92 % of the subjective comparative questions with a precision of 0.98 (other options: precision of 1.0, recall 0.02; precision of 0.99, recall 0.62).

4.3.2 Baseline Stance Detector

As a baseline stance detector, we use a pre-trained classifier from the IBM Debater project via its API [18]. For a pair of `(text, topic)` as input, it scores from -1 (strong con) to +1 (strong pro) to which extent the text supports the given topic. This stance detector is a BERT-based classifier that was trained on 400,000 labeled examples [17, 195].

Since we deal with two stance targets (two comparison objects), we prompt the API to return two scores for each answer. We create the input in two ways: (1) only a comparison object as a topic, and (2) an object appended with the sentiment phrase "is good" as the topic (e.g., "<object> is good"). We query the API with the unmasked and masked objects in questions and answers. Finally, on the pairs of scores for each answer, we fit a linear SVM[8] on our manually annotated four stance classes (pilot experiments showed that SVM was more accurate than logistic regression and feedforward deep neural network). Different from the previous classification setups, here we use 80 / 20 train–test splits instead of cross-validation due to the smaller amount of just 950 annotated question–answer pairs. The results in Table 4.6 show that the baselines are quite good at classi-

[7] RoBERTa: large, learning rate=0.00002, epochs=10, batch size=8, max seq length=64.

[8] SVM hyperparameters selected with a grid search and 5-fold cross-validation on the train split: C=1.0, penalty="l2", loss="squared_hinge".

fying the 'pro first object' stance but never correctly predict the 'no stance' class. Furthermore, they are more effective for unmasked objects and when the topic is extended with a sentiment like "<object> is good".

4.3.3 Classifiers with Transformer Embeddings as Representations

We first experiment with logistic regression and DNN trained on transformer embeddings used to represent questions and answers. In pilot experiments, we evaluated several transformer architectures, including BERT and XLNet [214] and found that RoBERTa (large) and Longformer (large), which also overcomes the 512-token input sequence length limit, worked best; both using the mean of all token embeddings (more accurate than using only the `CLS`-embedding). To evaluate whether a comparative question itself contributes to the stance detection effectiveness, we either represent only the answer via embeddings or the concatenation of a question and its answer (subscripts `A` and `QA` in Table 4.6). On the embeddings, logistic regression and DNN are trained as the classifiers.[9]

In Table 4.6, we report the effectiveness of classifiers that are either the most accurate on the four stance classes (evaluated using accuracy) or they achieve the highest F1 for one of the stance classes either within a respective type of classifiers or across all models. The evaluation results on the test set show that the classifiers trained on transformer embeddings are generally more accurate at predicting the 'neutral' and 'no stance' classes compared to the baselines. But even though logistic regression with Longformer-represented concatenations of questions and answers without object masking achieves the highest F1 of 0.46 for the 'no stance' class across all models, concatenating questions and answers on average does not help, while object masking improves the overall accuracy by about 0.1.

[9] Logistic regression hyperparameters selected with a grid search and 5-fold cross-validation on the train split: (a) Unmasked: RoBERTa$_{\text{A}}$: C=100; Longformer$_{\text{A}}$: C=1.0, solver="liblinear"; Longformer$_{\text{QA}}$: C=15, solver="lbfgs"; (b) Masked: RoBERTa$_{\text{A}}$: C=0.07, solver="lbfgs"; both Longformer: C=100, solver="lbfgs". In all penalty="l2". DNN: 3 hidden layers with output units: 256, 64, 16, activation="relu", epochs=100 with early stopping, batch size=5, loss="categorical_crossentropy", optimizer="adam", optimization metric: "accuracy".

Table 4.6: Effectiveness of answer stance detection on the test set with unmasked or masked comparison objects: overall accuracy (Acc.) and F1 per class (O1: pro first object, O2: pro second object, Neu.: neutral, None: no stance); best values in bold. Subscripts: object (`O`), first object (`O1`), second object (`O2`), sentiment prompt "is good" (`GOOD`), sentiment prompt "is better" (`BETTER`), input uses the separator token (`SEP`), only answer (`A`), question & answer (`QA`). GPT-3 results that are the same or higher than the best ones of our fine-tuned stance detectors are underlined.

	Unmasked					Masked				
	Acc.	**O1**	**O2**	**Neu.**	**None**	**Acc.**	**O1**	**O2**	**Neu.**	**None**
Baselines										
IBM + SVM	0.46	0.57	0.46	0.34	0.00	0.44	0.54	0.46	0.33	0.00
IBM$_{\text{O GOOD}}$ + SVM	0.50	0.61	0.56	0.27	0.00	0.47	0.58	0.48	0.38	0.00
Classifiers with Transformer embeddings as representations										
RoBERTa$_{\text{A}}$ + DNN	0.42	0.41	0.31	0.52	0.27	0.48	0.56	0.26	0.51	0.36
Longformer$_{\text{A}}$ + DNN	0.40	0.44	0.30	0.44	0.35	0.49	0.59	0.51	0.29	**0.46**
Longformer$_{\text{A}}$ + Log. Regr.	0.42	0.41	0.40	0.53	0.00	0.51	0.56	0.45	0.52	0.45
Longformer$_{\text{QA}}$ + Log. Regr.	0.39	0.38	0.37	0.41	**0.46**	0.49	0.57	0.46	0.43	0.45
Fine-tuned Transformers as classifiers										
RoBERTa$_{\text{A}}$	0.46	0.48	0.38	0.56	0.00	0.57	0.60	0.64	0.55	0.00
RoBERTa$_{\text{SEP QA}}$	0.46	0.48	0.36	0.57	0.18	0.60	0.65	0.67	0.54	0.28
Longformer$_{\text{A}}$	0.45	0.45	0.43	0.52	0.17	0.49	0.60	0.42	0.51	0.00
Longformer$_{\text{SEP QA}}$	0.45	0.53	0.32	0.54	0.00	0.56	0.62	0.55	0.54	0.20
Transformers with sentiment prompt										
RoBERTa$_{\text{SEP O1 GOOD}}$	0.58	0.60	0.61	0.60	0.29	**0.63**	**0.70**	0.67	0.53	0.40
RoBERTa$_{\text{SEP O1 BETTER}}$	**0.59**	0.62	**0.63**	0.58	0.19	0.62	0.68	**0.69**	**0.56**	0.36
RoBERTa$_{\text{SEP O2 GOOD}}$	0.54	0.55	0.52	**0.61**	0.29	0.57	0.60	0.67	0.50	0.31
Longformer$_{\text{SEP O1 GOOD}}$	0.56	**0.63**	0.55	0.54	0.21	0.56	0.63	0.55	0.54	0.21
GPT-3 (post hoc evaluation)										
GPT-3$_{\text{ZERO-SHOT}}$	0.59	0.64	0.67	0.61	0.11	0.54	0.58	0.58	0.58	0.21
GPT-3$_{\text{FEW-SHOT}}$	0.65	0.72	0.75	0.61	0.38	–	–	–	–	–

4.3.4 Fine-tuned Transformers as Classifiers

In the next set of experiments, we fine-tune pre-trained RoBERTa and Longformer models[10] using as input only answers or question–answer pairs in the form of `question [SEP] answer` (subscript `SEP QA` in Table 4.6)—the reverse input `answer [SEP] question` yields lower accuracies. The results in Table 4.6 show that the classifiers predict the 'neutral' and 'no stance' classes more accurately than the baselines and that object masking again improves the overall accuracy. For unmasked objects, the joint question–answer representations do not seem to help. However, with the masked objects, the classifiers benefit from a combined question–answer input. Interestingly, Longformer-based classification results are not better than the RoBERTa-based ones. Possible explanations could be that the most important information for the stance detection is concentrated at the beginning of an answer and that, due to the GPU memory limitations, we fine-tuned a Longformer base model but a large model for RoBERTa.

4.3.5 Transformers with Sentiment Prompts

Having observed that the baseline classifiers are more effective with some sentiment prompts, we add one of the two sentiment prompts: "is good" or "is better" to the comparison objects before fine-tuning the transformer models. The results in Table 4.6 show that such stance detectors achieve the highest accuracies for the neutral and the O1 and O2 classes, as well as the overall best accuracy values. As before, masking the objects yields better results—but this time only slightly better—and Longformer is less effective than RoBERTa—the reason again might be that we use a Longformer base model but a large model for RoBERTa. An interesting observation is that extending the first comparison object with the two different sentiment prompts (subscripts: `SEP O1 GOOD` and `SEP O1 BETTER`) yields better results than prompting for the second object (hence, not many results for prompting the second object are shown in the table). Another interesting

[10] RoBERTa large and Longformer base (due to the GPU memory limitation), hyperparameters selected with 10-fold cross-validation on the train split: learning rate=0.00002, epochs=10, batch size=4.

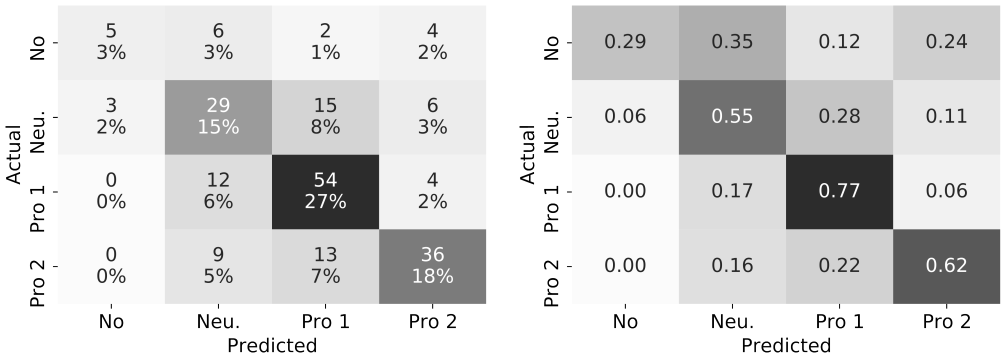


Figure 4.3: Confusion matrices: without normalization (left) and with normalization by a class support size, i.e., a number of elements in each class (right).

observation is that using the first comparison object is not only important for the overall accuracy and the 'pro first object' class but also for the 'pro second object' class. A reason might be the ways of how humans formulate comparative answers with two choice options—definitely an interesting direction for deeper investigations in future work.

Error Analysis

Figure 4.3 shows a confusion matrix and its normalized version for our best stance predictor $\text{RoBERTa}_{\text{SEP O1 GOOD}}$ with masked objects (cf. Table 4.6). We now review an example for which the stance detector mistakenly predicted a 'pro first object' stance, while the labeled stance is 'pro second object'—one of the most frequent confusions. For the question "Which is better for an undergraduate in PhD admission, [FIRST_OBJECT: a low-quality paper] or [SECOND_OBJECT: no paper]?", the answer is:

If the quality of the work is low, the student should neither publish it in a lower tier conference nor publish it as a technical report. They should either make the time to improve it or toss it in the trash. [FIRST_OBJECT: *A bad publication*]*, no matter what venue it's published in, is worse than* [SECOND_OBJECT: *no publication*] *at all. Similarly, a "publication" listed in a CV or described in a statement of purpose that isn't retrievable via google* (*unlike most technical reports, which are googlable*) *is also worse than* [SECOND_OBJECT: *no publication*] *at all, because we can't tell if the applicant is lying.* (*Sadly, some applicants are lying.*)

It is possible that the classifier did not "learn" to correctly recognize the 'worse'-relation between the comparison objects. Indeed, in the train set, there are only 13 passages (out of 760) that contain the word 'worse', and only 2 contain the phrase 'worse than'. Whereas 199 passages contain 'better' and 42 contain 'better than'. However, other configurations of the stance detector classify the stance correctly; for instance, those that use the prompts "object 2 is good" and "object 1 is better", and that takes a whole question and an answer as input.

Another frequent confusion is when the classifier predicts a 'pro first object' stance when the text passage is labeled as 'neutral'. For instance, for the question "Are [FIRST_OBJECT: MACs] really better than [SECOND_OBJECT: PCs]?", the answer is:

When was the last virus to attack [FIRST_OBJECT: *a MAC*]*? one within 20 years. It's a matter of opinion. I have been running* [SECOND_OBJECT: *a PC*] *for 10 years, and am fed up of them. I plan to purchase* [FIRST_OBJECT: *a MAC*] *next, as I realize it will be so much easier and simpler to operate.* [SECOND_OBJECT: *PCs*] *and* [FIRST_OBJECT: *a MACs*] *have there place. It all depends what you want to do. Playing Games a lot? then choose* [SECOND_OBJECT: *a PC*]*. Do a lot of video editing or are you creative? Get* [FIRST_OBJECT: *a MAC*].

While the first part of the answer example indeed contains a strong 'pro first object' argument, our annotators might have labeled the overall answer as neutral due to the concluding part of the answer, which suggests that the choice depends and each comparison object is good for something else. An interesting future work may be to investigate the importance of different parts of text passages and their contribution to the overall stance, somewhat similar to the idea of a sentiment flow in reviews [199].

4.3.6 Post Hoc Stance Detection with GPT-3

After having our own stance detectors evaluated, we post hoc investigate if using pre-trained generative large language models can improve the stance classification effectiveness. For experiments, we use the GPT-3 API [41] with default hyperparameters.[11] We test the model using a zero-shot (i.e., no examples are given) and a few-shot (i.e., several examples are provided) prompting. For the zero-shot prompting, we use the following input:

I have a question comparing {obj1} and {obj2}: {question}
Identify whether the following text is pro {obj1}, pro {obj2}, neutral, or no stance. Please, answer only with "pro {obj1}", "pro {obj2}", "neutral", or "no stance": {answer text}

The curly brackets {...} in the prompt indicate placeholders that are filled with the comparison objects, the question, and the answer text from our test set. Analogously to the stance detection experiments described above, we use the actual comparison objects and their special masking tokens. Results in Table 4.6 show that GPT-3 (zero-shot) achieves the same accuracy of 0.59 as our sentiment-prompted RoBERTa-based stance detector using the original comparison objects. Whereas using the objects' masking tokens decreases the model effectiveness. This is most likely due to the fact that GPT-3 was trained on naturally written texts. Since masking the comparison objects is unsuccessful in the zero-shot setting, we further experiment with the original objects, questions, and answers.

For the few-shot prompting, we use the following input to the model:

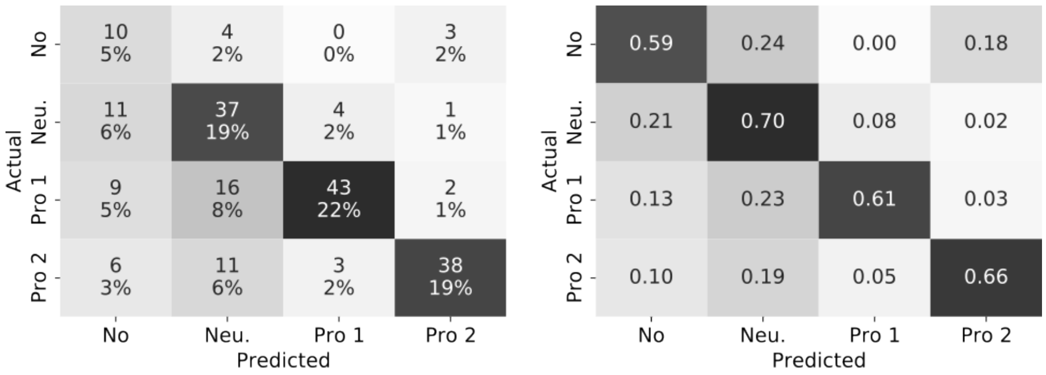
You will be shown a text passage that compares two objects. Decide if the passage provides arguments pro first object, pro second object, neutral, or no stance is given. First, we start with examples and definitions. Please read them carefully.
QUESTION: Apple vs Microsoft: which do you like better?
ANSWER PASSAGE: I switched from PC to Mac about 2 years ago, after becoming familiar with Macs using my sister's computer. I will NEVER go back to PCs. I also like that Macs are simplified for basic things such as photos, music, internet and e-mail. Truthfully, the only programs I have issues with are Microsoft applications like Word and IE. I think Apple's superiority comes from the fact that Macs are inherently more stable systems.
FIRST OBJECT: Apple; SECOND OBJECT: Microsoft.

[11] Parameters: model=text-davinci-003, temperature=0.0, max_tokens=64, top_p=1.0, frequency_penalty=0.0, presence_penalty=0.0.

Explanation: The answer provides a strong pro argument (opinion) for MAC (which is referred to as Apple). Note, that the text passage may not use the same object names as the question, e.g., it can contain synonyms or abbreviations or just mention only one object. Stance: PRO FIRST OBJECT
[...]
Now, I have a question comparing FIRST OBJECT: {obj1} and SECOND OBJECT: {obj2}
QUESTION: {question}
Identify whether the following text is "PRO FIRST OBJECT", "PRO SECOND OBJECT", "NEUTRAL", or "NO STANCE". Please, answer only with "PRO FIRST OBJECT", "PRO SECOND OBJECT", "NEUTRAL", or "NO STANCE":
ANSWER PASSAGE: {answer}
Stance:

As before, the curly brackets {...} indicate placeholders that are filled with the comparison objects, the question, and the answer text from our test set. In the prompt, we include in total four examples (manually selected from our train set), one for each stance (for brevity, in the example prompt above, we only show an example for one stance label).

Using GPT-3 as a stance detector by providing the instruction-like examples improves the overall classification accuracy by 2 points compared to our most effective sentiment-prompted RoBERTa-based classifier that uses object masking (achieved accuracy of 0.65 vs. 0.63; cf. Table 4.6). This result improvement is, however, not significantly better (paired Student's t-test, p=0.05). While the GPT-3 stance detector is more accurate at predicting the 'pro first object', 'pro second object', and 'neutral' stance classes, it makes more errors at predicting the 'no stance' class, often confusing it with the 'neutral' class (cf. confusion matrices in Figure 4.3 and Figure 4.4). Moreover, while confusing the 'pro first' and 'pro second object' stances less often, GPT-3 tends to predict 'neutral' and 'no stance' more frequently.



Actual \ Predicted	No	Neu.	Pro 1	Pro 2
No	10 5%	4 2%	0 0%	3 2%
Neu.	11 6%	37 19%	4 2%	1 1%
Pro 1	9 5%	16 8%	43 22%	2 1%
Pro 2	6 3%	11 6%	3 2%	38 19%

Actual \ Predicted	No	Neu.	Pro 1	Pro 2
No	0.59	0.24	0.00	0.18
Neu.	0.21	0.70	0.08	0.02
Pro 1	0.13	0.23	0.61	0.03
Pro 2	0.10	0.19	0.05	0.66

Figure 4.4: Confusion matrices: without normalization (left) and with normalization by a class support size, i.e., a number of elements in each class (right).

4.4 Summary

This chapter introduced a dataset of 3,500 comparative questions labeled with comparison objects, aspects, and predicates and with question categories: subjective/factual, direct/indirect, and with/without aspects. For 950 questions the stance of potential answers is labeled as supporting the first or second comparison object, being neutral, or taking no stance.

In pursuit of developing an approach to parsing comparative questions, we have trained classifiers that detect their important components such as the objects to be compared, the comparison aspects (i.e., the objects' shared properties over which they are intended to be compared), and the predicates (i.e., the terms that establish a comparison relation between the objects). These classifiers help to better "understand" the information need behind a comparative question. Our fine-tuned RoBERTa-based classifier identifies the comparison predicates with an F1 of 0.98—almost perfect classifier—followed by the object identification with an F1 of 0.93. The most challenging remains the aspect identification with an F1 of 0.80. While tagging the aspects in only comparative questions with aspects (based on manual labels) increases the effectiveness of aspect identification by 0.1 in terms of F1, classifying comparative questions as 'with an aspect' remains challenging. One straightforward solution to increase the effectiveness of the aspect identification that can be investigated in future work is to enlarge the training dataset. Another solution is to enhance a search system with a clarification feature to ask the user back when the classifier is "not sure" whether an aspect is contained.

A particular focus of this chapter then is comparative questions that require subjective answers in form of opinions and arguments. We have trained a high-precision classifier to identify subjective comparative questions, and in a study on 950 such questions, we trained and evaluated an answer stance classifier. Our most accurate stance detector that does not require comparison object identification is RoBERTa fine-tuned on the answers with unmasked objects—achieving an overall accuracy of 0.46. However, identifying the first comparison object in a question and extending it with a sentiment prompt improves the accuracy to 0.59, while the overall most effective approach (accuracy of 0.63) is to identify and mask the comparison objects in questions and answers. Though promising, a limitation of the masked approaches' experimental results is that we so far have relied on the manual labeling and matching of the comparison objects in questions and answers. Since the objects in questions and answers could have quite different syntactic forms (e.g., "operating system" in the question and "OS" in the answer), an actual masking-based stance detector will need an automatic highly accurate object matching component—an important direction for future work. In the post hoc evaluation, we also used GPT-3 for stance detection, which achieved slightly higher accuracy compared to our most effective RoBERTa-based stance detector configuration. This gain is, however, not significantly better. Given that generative language models are sensitive to prompts, further experiments using different prompting (e.g., by providing more examples, using different formulations, etc.) can be an interesting direction for future work.

Since we also observed that the highest F1 scores on the different stance classes are achieved by different classifiers and prompts, further studies of combinations or ensembles of the individual classifiers and different prompting ideas will probably improve the effectiveness. Also, identifying the parts of an answer that are the most important for stance detection might be an interesting direction to pursue. Finally, for an actual search engine, receiving a subjective comparative question, determining the confidence for a detected answer stance might help to, in doubt, rather retrieve some other text passages that are easier to classify for the overall presentation in a comparative result interface.

Our combined set of approaches forms the first step towards understanding and answering comparative questions. When recognizing that different opinions are expressed in information nuggets on the Web (i.e., different stances towards the objects in a comparative question), combining representatives of the different stances can be a powerful means to mitigate the risk of one-sidedness when just showing some direct answer extracted from some single web document. Instead, for comparisons, search engines could highlight different opinions/arguments side-by-side to allow a user to easily get an overview of the diversity of stances. Still, the actual answer stance detection leaves room for improvement.

5 Clarifying Comparative Questions

Vague or ambiguous queries can make it difficult for a search system like a web search engine to correctly interpret a user's underlying information need. A relatively "simple" solution then is result diversification to cover different interpretations, while in more "conversational" search interfaces, the user can be prompted to clarify their original request. In this chapter, we study clarification in the scenario of comparative questions. In our experiment that reflects a conversational search interface with a clarification component, 70 % of the study participants find clarifications useful to retrieve relevant results for questions with unclear comparison aspects (e.g., "Which is better, Bali or Phuket?") or without explicit comparison objects and aspects (e.g., "What is the best antibiotic?").

Since the very early question-answering systems were developed [178], brevity and ambiguity of human language have been big challenges. To return personalized and more relevant results for vague requests, search engines usually use disambiguation techniques such as result diversification in the sense of including results for different potential intents [168] or query suggestions to let the user select a better query [119]. Though these techniques are rather common in current web search interfaces, their application on mobile devices or in voice search might be hard. To address this issue, recently, new ideas have been proposed like query reformulation in

User: Which is better, Bali or Phuket?

System: Over which aspect do you want me to compare them?
People usually compare Bali vs. Phuket over:
(1) nightlife, (2) prices, (3) breakfast.
Or do you want **(4) a general comparison?**

User: Night life.

System: Both, Bali and Phuket offer a vibrant nightlife.
Bali has more of a sophisticated spirit with many
beach side and rooftop clubs, while Phuket has more
go-go bars and casual nightclubs.

Figure 5.1: Conceptual conversation of a search system that interacts with a user by suggesting clarification options.

a conversational context [96] or clarification [9, 101, 104, 218, 219, 220, 225]. Several studies have already shown that in the case of conversational search, users appreciate systems (be it a voice search or a traditional web search) that ask for clarification [37, 89, 94, 95, 218, 220].

We study clarification specifically for comparative questions like “Which is better, Bali or Phuket?” that often represent the need to come to an informed decision about choosing one or another option. As our search engine log analysis in Chapter 3 showed, a majority of comparative questions does not specify an aspect on which the comparison should be based (e.g., “Which is better, Bali or Phuket?”) and does not clearly state the to-be-compared objects (e.g., superlative questions like “What is the best antibiotic?”). Then, in Chapter 4, we proposed approaches to classify comparative questions as with or without comparison objects and aspects. For such comparison scenarios, where the to-be-compared objects or the comparison aspects initially were not specified, in this chapter, we study whether clarification requests and option suggestions from an interactive search system can help searchers to find more satisfactory answers.

Figure 5.1 depicts an example for the underspecified question “Which is better, Bali or Phuket?”. In the clarification request, the system proposes three aspects for the comparison or to search without the clarification (option 'general comparison' in Figure 5.1). A few previous studies showed that users seem to appreciate three clarification suggestions [94, 95], but Zamani et al. [220] found no correlation between Bing users' engagement rates and the number of clarification options. In the example, for the chosen aspect 'nightlife', the system then returns an answer.

An existing search system that helps with comparative information needs is CAM (comparative argumentative machine) [169]. It accepts from the user two to-be-compared objects and optional comparison aspects in separate input boxes. The search results vary depending on the specified comparison aspects. The system also offers some potential further aspects but does not proactively clarify unspecified aspects. We close this gap by addressing the question of whether clarification interactions improve user satisfaction in comparative search scenarios.

To investigate to what extent clarification of ambiguous comparative questions helps searchers in finding more satisfactory answers, we conduct two user studies: A searcher interacts with a "conversational" system that actively tries to clarify unspecified comparison objects and aspects (cf. Figure 5.2). In Section 5.1, we overview the data collection used for the user study. Then in Section 5.2, we give information about the study participants, the study design, and discuss the main findings. Finally, Section 5.3 concludes this chapter by summarizing the main contributions and discussing open questions and future work.

With our focus on the specific use case of comparative searches, we complement previous more general clarification studies. Those studies, for example, found that search engine users find clarifications useful (functional and emotional benefits) [218], the users are less dissatisfied with search results when interacting with clarifications [220], and that clarification interactions between users at Stack Exchange are usually helpful to get better answers to their original questions [192]. Our main results on clarifications in comparative search scenarios are similar. The participants of our study use one of the three suggested clarification options in at least 70 % of the cases. More than 85 % of the participants enjoyed their experience with the system and indicated that the clarification options were helpful to find satisfactory answers for at least 75 % of their assigned tasks.

5.1 Data for the User Study

Realistic comparative search scenarios for our user study were selected as follows. Using the ALBERT-based [107] classifier (ALBERT was fine-tuned on 31,000 questions annotated as comparative or not), we first found a total of 64,000 likely comparative questions in the MS MARCO [134] dataset (Bing search questions), the Google Natural Questions dataset [106], and in the Stack Exchange archive.[1] Focusing on questions that might need clarification, we then ran the RoBERTa-based classifier from Section 4.2.3 (RoBERTa [118] fine-tuned on comparative questions manually labeled as with or without comparison aspects or objects) and found 22,500 questions that mention comparison objects but have an unclear aspect (e.g., "Which is better, Bali or Phuket?") and 20,000 questions without comparison objects and aspects (e.g., "What is the best antibiotic?"). We randomly sampled 15 questions for the aspect clarification and 10 questions for the object and aspect clarification. Each question was manually checked and replaced in case of misclassification until we had found 15 with unclear aspects and 10 without objects and aspects. In the selection process, we also manually ensured that the comparative questions covered diverse topical domains like cars, food, electronics, travel, sports, health, arts, and occupation.

To select object clarification options for the 10 queries with missing comparison objects (e.g., "What is the best occupation?" or "What is the best antibiotic?"), we scraped entities from 'list of' Wikipedia articles[2] (e.g., list of occupations) and searched for Wikidata entries via 'instance of' (P31) queries against the Wikidata query service[3] (e.g., instance of antibiotics (Q12187)). From the obtained entities, we selected the pairs with the highest sentence-wise co-occurrence frequencies in the Common Crawl snapshot 2014-15[4] (e.g., 'drummer' and 'guitarist' for occupation or 'amoxicillin' and 'ciprofloxacin' for antibiotics).

As for the clarification options for missing comparison aspects, we manually identified the compared objects in the selected questions and used

[1] https://archive.org/details/stackexchange

[2] https://en.wikipedia.org/wiki/List_of_lists_of_lists

[3] https://query.wikidata.org/

[4] http://commoncrawl.org/2014/07/april-2014-crawl-data-available/

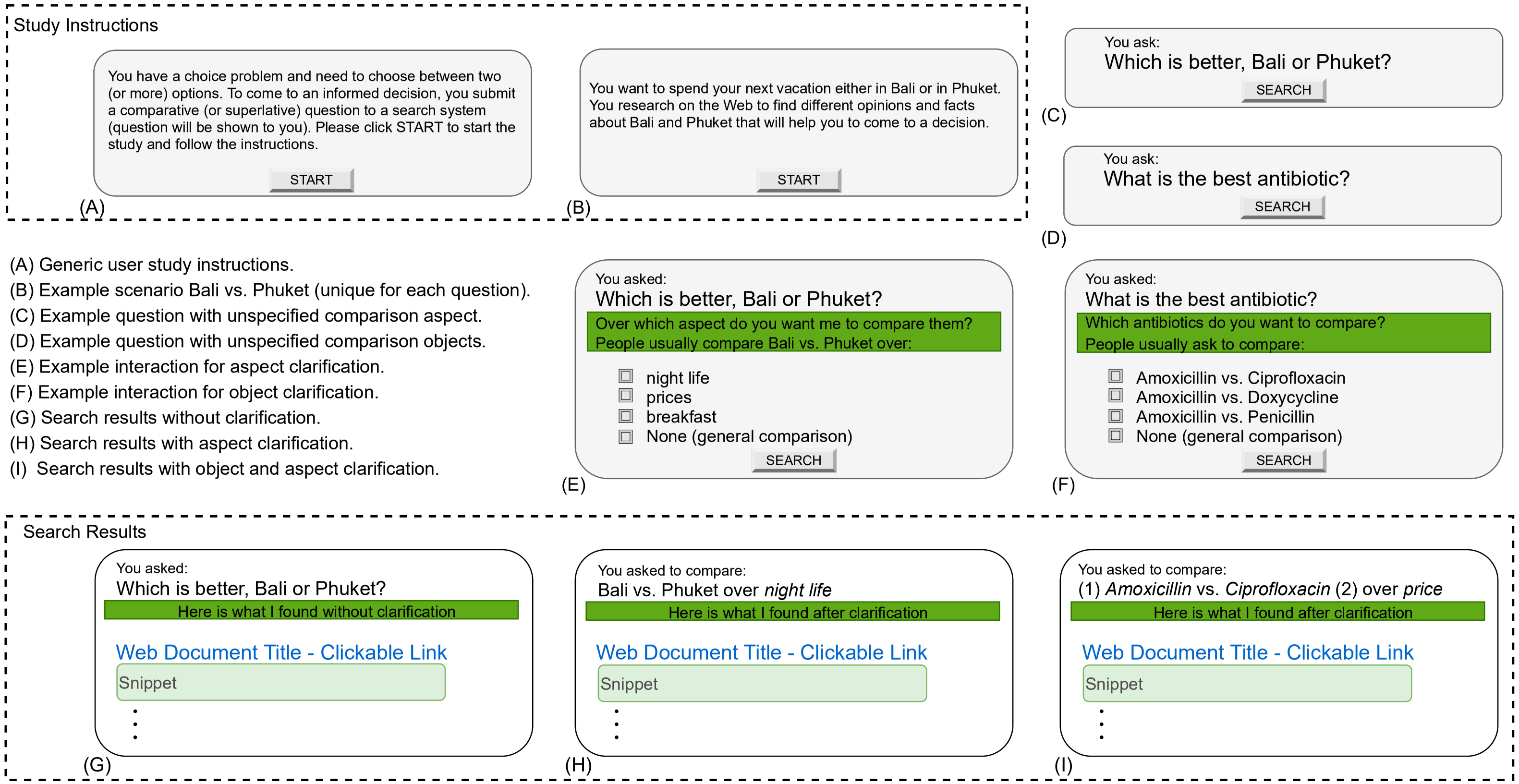
the following two strategies. (1) We queried the API of CAM [169][5] with the object pair (e.g., 'Bali' vs. 'Phuket') and collected the returned aspect suggestions (CAM finds them in comparative sentences using patterns like "`Object 1` is better than `Object 2` for `Aspect`"). (2) We searched for manually annotated comparison aspects in the existing corpora of comparative sentences [14, 85, 86]. For all aspects found by these two strategies, we manually checked their validity until we found three options per question.

The search result pages that should be shown to the study participants were created before the actual study since the possible clarification options were also pre-computed, as explained above. We manually submitted the original comparative questions and versions with included clarification options to Google, stored the HTML files of the search results pages, and extracted the document titles, the URLs, and the snippets to show web search-like results but leaving out the ads displayed by Google, etc.

5.2 User Study Design and Results

To address the question of whether clarifications improve the user "satisfaction" in comparative search scenarios, we conduct a user study for the cases: (1) comparative questions with unspecified comparison aspects (e.g., "Which is better, Bali or Phuket?") and (2) questions without explicitly specified to-be-compared objects and without aspects (e.g., "What is the best antibiotic?"). In particular, we study whether clarifications are helpful in finding satisfactory answers to underspecified comparative questions and whether a search interface with comparison aspect and object clarifications overall is pleasant to use. The user interface for the study (inspired by the studies of Zamani et al. [218, 220]) reflects an "interactive" way of questions and answers that allows the system to express uncertainty about a specific part of the question (e.g., comparison object and aspects) and to suggest some clarification options (see Figure 5.2).

[5] `http://ltdemos.informatik.uni-hamburg.de/cam/api-info`



You have a choice problem and need to choose between two (or more) options. To come to an informed decision, you submit a comparative (or superlative) question to a search system (question will be shown to you). Please click START to start the study and follow the instructions.

You want to spend your next vacation either in Bali or in Phuket. You research on the Web to find different opinions and facts about Bali and Phuket that will help you to come to a decision.

You ask:
Which is better, Bali or Phuket?

You ask:
What is the best antibiotic?

(A) Generic user study instructions.
(B) Example scenario Bali vs. Phuket (unique for each question).
(C) Example question with unspecified comparison aspect.
(D) Example question with unspecified comparison objects.
(E) Example interaction for aspect clarification.
(F) Example interaction for object clarification.
(G) Search results without clarification.
(H) Search results with aspect clarification.
(I) Search results with object and aspect clarification.

You asked:
Which is better, Bali or Phuket?
Over which aspect do you want me to compare them?
People usually compare Bali vs. Phuket over:

□ night life
□ prices
□ breakfast
□ None (general comparison)

You asked:
What is the best antibiotic?
Which antibiotics do you want to compare?
People usually ask to compare:

□ Amoxicillin vs. Ciprofloxacin
□ Amoxicillin vs. Doxycycline
□ Amoxicillin vs. Penicillin
□ None (general comparison)

You asked:
Which is better, Bali or Phuket?
Here is what I found without clarification
Web Document Title - Clickable Link
Snippet
⋮

You asked to compare:
Bali vs. Phuket over *night life*
Here is what I found after clarification
Web Document Title - Clickable Link
Snippet
⋮

You asked to compare:
(1) *Amoxicillin* vs. *Ciprofloxacin* (2) over *price*
Here is what I found after clarification
Web Document Title - Clickable Link
Snippet
⋮

Figure 5.2: User study interface design.

5.2.1 Study Participants

For the user study, we recruited seven volunteers: five males and two females between 20 and 39 years old. Two of them had a completed Bachelor's degree, three held a Master's degree, and two had no completed college or university degree. For all participants, English was not their mother tongue—two participants stated to have an intermediate level of English, one stated upper-intermediate, and four stated to have an advanced level of English. The study participants had diverse occupational and educational backgrounds, including bioinformatics, computer science, construction works, service industry, and web development. All the participants originated from or lived in Europe and Asia.

5.2.2 Study Setup

We developed the study interface in Python using the graphical user interface package tkinter.[6] At the beginning of the study, the participants were notified that the study participation is voluntary, that they could refuse to participate or continue at any point without providing a reason, that their names or email addresses were not collected (their identity could not be determined), and that the collected data was used solely for research purposes. After accepting these conditions, each participant saw the general description of the study scenario (cf. Figure 5.2 (A)): They would need to assume that they are facing a choice problem and want to make an informed decision based on submitting a comparative question to a search engine and that the actual question will be predefined. When clicking on 'Start', the actual study began by showing a description of a random scenario from our set of 15 questions without aspects followed by the 10 questions without objects and aspects (each study participant worked on every question; order randomized in the two question groups). The brief scenario descriptions (cf. Figure 5.2 (B) for an example) were manually created. After starting a topic, the respective initial question was displayed: either one with unclear aspects (cf. Figure 5.2 (C)) or one with unclear objects and aspects (cf. Figure 5.2 (D)). Participants were not limited in time.

[6] https://docs.python.org/3/library/tkinter.html

5.2.3 Clarifying Comparison Aspects

After reading a short search scenario description, the participants were shown the respective comparative question (with an unspecified comparison aspect in this part of our study; example in Figure 5.2 (C)). After clicking on the search button, the participants were shown the results for that underspecified question (cf. Figure 5.2 (G); similar to standard web search results pages: ten results with snippets and clickable document titles linking to the original web pages) along with a clarification prompt that asks "Over which aspect do you want me to compare them?" suggesting the three predefined aspect clarification options (cf. Figure 5.2 (E)). The participants could explore the original results and decide whether an aspect clarification could be useful. In case of choosing a clarification option, another result page for the question with the clarified aspect was shown (cf. Figure 5.2 (H)). Afterwards, the participants were asked to provide their feedback. During answering the survey questions, the result page(s)—before and, if chosen, after clarification—were available for comparison or further inspection. We asked the participants to answer the following two questions on each scenario: (1) whether they found a satisfactory answer to their question ('Yes, I found the answer to my question', 'More or less: I found something useful, but might search further', 'No, I did not find anything useful at all', and 'I don't know') and (2) how useful / helpful clarification was in case they selected one of the clarification options ('Yes, I found the answer to my question using clarification', 'More or less: Results after clarification gave me some useful additional information', 'No, results after clarification did not provide any useful additional information'). Finally, after completion of the 15 questions without aspects, we asked the participants to rate the overall experience using the system (whether the system was pleasant to use with the options 'yes', 'more or less', or 'no'). After this exit question, the ten questions without objects and aspects followed in the next study (cf. Section 5.2.4).

Table 5.1: Results of the user study on clarifying comparative questions without aspects (15 questions, 7 participants).

Search result quality		*Aspect clarification*		*Overall*	
I found an answer:	(%)	**Clarification helpful:**	(%)	**Pleasant to use:**	(%)
Yes	76	Yes	41	Yes	15
More or less	23	More or less	28	More or less	85
No	1	No	21	No	0
Don’t know	0	Don’t know	0	Don’t know	0
		Clarification not used	10		
	α=0.42		α=0.32		

Study Results

The results of our user study on clarifying comparative questions without comparison aspects are summarized in Table 5.1. In 76 % of the 105 cases, the participants stated that they were able to find satisfactory answers to the questions; in 23 %, they found only partial answers and would want to search for more information. The initial vague questions were refined with a suggested comparison aspect in 90 % of the cases. For a majority of the cases with the used clarification option, the participants found the clarification helpful to obtain good results. All the participants enjoyed using the system, however, only 15 % were entirely satisfied. The actual agreement between the participants' votes per question is rather low (cf. the Krippendorff's α [102] values in Table 5.1) indicating that assessing the clarification results and the overall clarification usefulness is a subjective task. Still, the votes on whether a satisfactory answer was found have a slightly higher agreement than the ones on the helpfulness of aspect clarification.

5.2.4 Clarifying Comparison Objects and Aspects

To evaluate the usefulness of clarifications for comparative questions that do not explicitly mention the to-be-compared objects and that have no aspects (e.g., "What is the best antibiotic?"), we conduct a second part of the user study with the same seven participants. Similar to the first part, the participants started by "submitting" the original query (cf. Figure 5.2 (D)) but this time the results were complemented by three suggestions for object clarification (e.g., Amoxicillin vs. Ciprofloxacin, Figure 5.2 (F)). If a participant selected an object clarification option, the system then showed results for the adjusted question and suggested clarification options for the comparison aspect similar to the first part of the study (cf. Figure 5.2 (E)). If a participant then also had selected a clarification for the aspect, the respectively adjusted final query was submitted to show results that match both clarifications (cf. Figure 5.2 (I)). For the final assessment, all search result pages that a participant had used were available (without clarification, with object clarification if selected, and with object + aspect clarification if selected). We asked the participants to evaluate whether they found a sat-

Table 5.2: Results of the user study on clarifying comparative questions without objects and aspects (10 questions, 7 participants).

Search result quality		*Object clarification*		*Aspect clarification*		*Overall*	
I found an answer:	(%)	**Clarification helpful:**	(%)	**Clarification helpful:**	(%)	**Pleasant to use:**	(%)
Yes	43	Yes	37	Yes	34	Yes	14
More or less	41	More or less	33	More or less	20	More or less	72
No	3	No	1	No	6	No	14
Don’t know	13	Don’t know	0	Don’t know	0	Don’t know	0
		Clarification not used	29	Clarification not used	40		
	α=0.49		α=0.45		α=0.27		

isfactory answer, whether the object and aspect clarifications were helpful (if used), and about their overall satisfaction with the system (same answer options as in Section 5.2.3).

Study Results

The results in Table 5.2 show that 71 % of the participants (out of 70 study cases in total) decided to use one of the suggested object clarification options. The lower ratio compared to the 90 % in the aspect-only clarification of the first part might be explained by the observation that the search results for superlative questions (i.e., "What is/are the best ...?") often contain a single "best" option or a list of several "best options". Some participants simply found that to be sufficient. Only the participants who had selected an object clarification then received aspect clarification options; 60 % of the participants decided to use both, an object and an aspect clarification. About 86 % of the participants enjoyed the system in this second part of our study (vs. 100 % for the first part), and 84 % of the participants stated that they had found a satisfactory answer (vs. 99 % in the first part of the study). Again, the agreement of the participants' votes per query is rather low (cf. the Krippendorff's α values in Table 5.2) with slightly higher rates for the satisfaction with the answers and the object clarifications.

5.3 Summary

In this chapter, we described a user study on clarifications in the scenario of vague comparative questions (i.e., without comparison objects or aspects). In our user study, we mimicked an interactive interface of a search engine that proactively suggests clarification options (comparison aspects and objects) for the underspecified comparative questions. Our study results are similar to previous more general studies: In at least 70 % of the cases, the participants decided to use clarifications to refine search results for initial queries. The majority of the participants also enjoyed their experience with the system's clarification component and found clarifications to be helpful for finding satisfactory answers. Even though we used realistic

Google search results and had participants with diverse backgrounds (education, occupation, etc.), our current small study (7 participants, 25 questions each) should be viewed as a pilot experiment with interesting initial results that justify a larger and deeper exploration.

An open question remains if a better, more suitable comparative clarification interface is possible. Although we attempted to design the system's interface based on the findings of previous work, we did not compare its different configurations. For instance, a future study could investigate what number of suggested clarification options is optimal. Another possible option is to additionally accept user input (e.g., comparison objects and aspects) if the suggested clarification options are unsatisfactory.

Nevertheless, since the general feedback about the clarification helpfulness was rather positive, a natural next step for future work is to develop the actual approaches that generate clarifying questions and clarification suggestions for comparison aspects and objects and then repeat the study with more participants for such a real system.

6
Argument Retrieval for Comparative Questions

Decision-making and opinion formation are natural routine tasks for many that often involve weighing arguments for or against different options. Any decision is usually grounded in personal prior knowledge and experience, but often also requires searching and processing new knowledge about the alternatives [7, 133]. With ubiquitous access to various kinds of information on the Web—from facts and opinions to anecdotes to arguments—everybody has the chance to acquire new information on almost any topic. Specifically, when searching for documents that can help answer subjective comparative questions, retrieval approaches should not only account for topical relevance but also should evaluate argument quality, and ideally analyze the stance of arguments towards the comparison objects in questions. The results of these steps can then be included in the search result presentation. A retrieved document, for instance, can be marked with a stance label and argument quality score. To foster the development of the respective retrieval and argument analysis methods, we have organized the Touché shared task on Argument Retrieval for Comparative Questions. Our analysis of the submitted approaches to the task shows that the most effective approaches to argument retrieval all share common characteristics. For example, most use various strategies for query reformulation and expansion, such as synonyms, relevance feedback, or generating new queries with pre-trained language models. An interesting observation is that re-ranking first-stage retrieval results based on a quality assessment of the arguments almost always improves retrieval effectiveness. Specifically, re-ranking based on important terms such as comparison objects and aspects or argument units in documents (premises and claims) is successful. Also, including in retrieval pipelines a re-ranking step based on the predicted stance has some promising effects on the retrieval effectiveness.

This chapter studies argument retrieval for comparative questions: We first introduce the overall setup of the Touché shared tasks in Section 6.1 and describe the task on argument retrieval for comparative questions in Section 6.2, including the task definition, search topics, and evaluation. Further, Section 6.3 provides an overview and analysis of the submitted approaches by the task participants and the evaluation results. The final Section 6.4 concludes the chapter, elaborates on open questions, and discusses future research directions.

6.1 Touché: Argument Retrieval

In this section, we briefly introduce the Touché argument retrieval labs[1] that we organized as part of the Conference and Labs of the Evaluation Forum (CLEF)[2] from 2020 through 2022 [32, 33, 35]. The goal of Touché is to support the development of argument retrieval and argument analysis technologies through providing test collections, submission and evaluation tools, and organizing collaborative events such as workshops. As part of Touché, we have organized the shared tasks and workshops on the topics such as argument retrieval for controversial and for comparative questions and image retrieval for arguments. Touché follows the conventional TREC (Text REtrieval Conference)[3] tasks methodology: Collections of documents and search topics are provided to the task participants, who then

[1] https://touche.webis.de

[2] https://www.clef-initiative.eu/

[3] https://trec.nist.gov/

submit their results (runs) for each topic to be annotated by human assessors. The corpora, search topics, and manual judgments created at Touché are freely available to the research community and can be found on the shared tasks' website.[4] Parts of the data are also already available via the BEIR [193] and `ir_datasets` [122] resources.

6.2 Shared Task on Argument Retrieval for Comparative Questions

In this section, we provide the details of the Touché task on argument retrieval for comparative questions. We describe the task definition and its goals, provide an example of a search topic, and explain the process of creating manual judgments used to evaluate participants' approaches.

6.2.1 Task Definition

The main goal of the task is to foster the development of technologies to support individuals' personal decisions in everyday life that can be formulated as comparative questions in the form "Is X better than Y with respect to Z?" and that do not have a single factual answer. Such questions can, for instance, be found on community question and answer platforms like Quora, but are also submitted as queries to search engines (cf. Chapter 3). Traditional search engines then often show text passages extracted from the content of fora discussions or from some web document as a direct answer above the classic "ten blue links". However, a problem of such attempts is that the retrieved passages may not always provide a diverse and sufficient overview of all possible options with well-formulated arguments, nor will all the extracted information be credible—a broader set of such issues also forms the dilemma of direct answers [145]. Thus, working on the technologies to retrieve and present diverse, credible arguments towards the comparison options constitutes the goals of this shared task.

[4] `https://webis.de/events.html?q=Touche#shared-tasks`

The participants of the task were asked to retrieve and rank documents from the ClueWeb12 collection (733 million English web pages; 27.3TB uncompressed)[5] or from the collection of about 1 million text passages coming from ClueWeb12 documents (passage retrieval was used in the third task year) that help answer comparative questions from search topics. Participants were allowed to submit up to 5 result rankings (runs). Ideally, the retrieved documents should contain relevant convincing arguments of high quality for or against the comparison options in a given question. Participation was also possible without indexing the entire ClueWeb12 on the participants' side since we provided easy access to the document collection through an API of the BM25F-based search engine ChatNoir [23].[6] To identify arguments in documents (premises and claims), the participants were not restricted to any system; they could use their own technology or any existing argument tagger of their choice. To lower the entry barriers for participants new to argument mining, we offered support via the neural argument tagger `TARGER` [50] hosted on our servers.

6.2.2 Search Topics

Each year, we used 50 search topics—in the second year, we used new topics and in the third year, we re-used 50 topics sampled from the two topic sets, but changed the retrieval corpus. To create the topics, we selected comparative questions from questions submitted to search engines or asked on question and answer platforms (from our dataset presented in Section 3.5), each covering some personal decision from everyday life. For every question, we formulated a respective TREC-style topic with a question as a title, a description of the search context and information need, and a narrative describing what makes a result relevant (i.e., serving as a guideline for human assessors). An example topic is shown in Table 6.1. During the topic creation, we ensured through manual spot checks that the ClueWeb12 collection actually contains possibly relevant documents.

[5] https://lemurproject.org/clueweb12/

[6] https://www.chatnoir.eu/

Table 6.1: Example topic for the Touché shared task on argument retrieval for comparative questions.

Number	1
Title	Should I major in philosophy or psychology?
Description	A soon-to-be high-school graduate finds themself at a crossroads in their life. Based on their interests, majoring in philosophy or in psychology are the potential options and the graduate is searching for information about the differences and similarities, as well as advantages and disadvantages of majoring in either of them (e.g., with respect to career opportunities or gained skills).
Narrative	Relevant documents will overview one of the two majors in terms of career prospects or developed new skills, or they will provide a list of reasons to major in one or the other. Highly relevant documents will compare the two majors side-by-side and help to decide which should be preferred in what context. Not relevant are study program and university advertisements or general descriptions of the disciplines that do not mention benefits, advantages, or pros/cons.

6.2.3 Manual Judgments

In the first year of organizing the shared task, we only evaluated the relevance of the retrieved documents but not any other argument quality dimensions. Using a top-5 pooling strategy of the submitted runs, including a baseline, a total of 1,783 unique results were judged by human assessors. We recruited seven graduate and undergraduate student volunteers, all with a computer science background. We used a kappa test of five documents from five topics to calibrate the annotators' interpretations of the guidelines (i.e., topics including the narratives) and the three relevance labels: 0 (not relevant), 1 (relevant), and 2 (highly relevant). The original Fleiss' κ of 0.46 indicates a moderate agreement such that a follow-up discussion among the annotators was invoked to adjust their individual interpretations and to emphasize that documents should not be judged as highly relevant when they do not provide well-formulated evidence support. After the training phase, each annotator judged the results for disjoint subsets of the topics (i.e., each topic was judged by one annotator only).

In the second year, we assessed the relevance and argument quality of 2,076 unique documents fetched again using a top-5 pooling strategy. Our eight volunteer annotators labeled documents for their topical relevance (three labels; 0: not relevant, 1: relevant, and 2: highly relevant) and whether rhetorically well-written arguments were contained (three labels; 0: low quality or no arguments in the document, 1: sufficient quality, and 2: high quality). The rhetorical quality [201], i.e., "well-writtenness" of the argument was defined by the following aspects: (1) Whether a document contains arguments (i.e., argumentative support is provided) and whether the text has a good style of speech (formal language is preferred over informal), (2) whether the text has a proper sentence structure and is easy to read and follow and whether it can be well understood, and (3) whether it includes profanity, has typos, and makes use of other detrimental styles.

Our eight volunteer assessors went through an initial kappa test on 15 documents from 3 topics (5 documents per topic): the observed Fleiss' κ values of 0.46 for relevance (moderate agreement) and of 0.22 for quality (fair agreement) are similar to previous related studies [72, 200, 201]. Again, however, we had a follow-up discussion with all the annotators to clarify some potential misinterpretations. Afterwards, each annotator independently judged the results of disjoint subsets of the topics (i.e., each topic was judged by one annotator only).

Finally, in the third year, we assessed the relevance, argument quality, and document stance of 2,107 unique text passages (again, top-5 pooling). Our six volunteers labeled the passages' relevance with three labels: 0 (not relevant), 1 (relevant), and 2 (highly relevant). They also assessed whether arguments were present in a passage and whether they were rhetorically well-written [201] with three labels: 0 (low quality, or no arguments in a passage), 1 (average quality), and 2 (high quality). Finally, we asked the assessors to label passages with respect to the topic's comparison objects as (a) pro first object, (b) pro second object, (c) neutral (both comparison objects are equally good or bad), and (d) no stance. As before, our assessors went through a training pilot annotation and discussion before labeling disjoint subsets of the topics. Fleiss' κ values in the pilot annotation phase were 0.30 for the relevance, 0.24 for the argument quality, and 0.39 for labeling the stance (all are fair agreement).

6.3 Task Evaluation and Results

In this section, we provide an overview of the submitted approaches by the participants in the argument retrieval shared task for comparative questions. We analyze the evaluation results, discuss what methodology proved to be successful, and summarize our key findings from the past three years of organizing this task. By understanding which approaches have been successful, we can gain valuable insights into the field of argument retrieval and continue to improve existing methods in the future.

6.3.1 Survey of Submissions at Touché 2020

Five teams submitted a total of eleven results to the task (ten of which plus the additional ChatNoir retrieval baseline were used to create the judgment pool). All approaches use the BM25F-based search engine ChatNoir [23] to retrieve candidate documents that are then re-ranked using machine learning models of different complexity in basically three steps: (1) Represent documents and queries using language models, (2) identify arguments and comparative structures in documents, and (3) assess argument quality. In Table 6.2, we report evaluation results of submitted participants' approaches using nDCG@5. Only one team was able to achieve a slightly higher score than the baseline approach. The result is however not significantly better. Below, we briefly describe the baseline retrieval approach and summarize the task participants' submissions. In the first task year, no training data was provided to the participants by us.

Puss in Boots is the baseline retrieval approach that simply uses the results that ChatNoir [23] returns for the original topic's title (a comparative

Table 6.2: Results for the task on comparative argument retrieval in 2020. Reported are the results of a team's best run according to relevance. The baseline approach is in bold. None of the results are significantly better compared to the baseline (paired Student's t-test, $p = 0.05$, Bonferroni correction).

Team	nDCG@5	Team	nDCG@5
	(Relevance)	(continued)	(Relevance)
Bilbo Baggins [2]	0.580	Katana [45]	0.564
Puss in Boots [23]	**0.568**	Frodo Baggins [177]	0.450
Inigo Montoya [82]	0.567	Zorro [173]	0.446

question). ChatNoir is an Elasticsearch-based search engine that indexes the complete ClueWeb12 (and also other web collections) by processing raw HTML documents using main content extraction, language detection, and extraction of metadata (keywords, headings, hostnames, etc.). During the retrieval step, ChatNoir combines BM25 scores of multiple fields (title, keywords, main content, and the full document) and uses the documents' SpamRank [56] as a threshold to remove spam.

As for the participants' submissions, two out of five participating teams use query expansion before querying ChatNoir. While team Frodo Baggins [177] simply augments each query term (comparative questions from the topics) with its nearest neighbor according to the cosine similarity using GloVe [140] embeddings, team Bilbo Baggins (submitted the most effective result) [2] uses a more sophisticated technique. After identifying using spaCy POS-tagger[7] the to-be-compared entities (nouns like laptop, desktop, etc.) and comparative terms (comparative adjectives or adverbs like better, best, etc.) in the topics' questions and extracting synonyms and antonyms (for the entities and comparative terms) from WordNet [125], the team creates three additional queries using different combinations of the identified important query constituents. Other teams simply use topics' questions as queries without further processing. For re-ranking the initially retrieved results, various argumentativeness and comparativeness features are used: (1) Number of comparative sentences in documents

[7] `https://spacy.io/`

(Bilbo Baggins and Katana [45]), (2) number of comparison objects, aspects, and predicates in documents (Katana), (3) argument ratio [154], document "credibility", number of sentences that support claims [155] (Bilbo Baggins), and (4) whether a document is classified as argumentative or not (Zorro [173]). Team Frodo Baggins uses a cosine similarity between the retrieved documents and generated ones using GPT-2 [150] conditioned on the topics' questions.

Team *Inigo Montoya* [82] takes a different approach: For the top-20 results by ChatNoir, `TARGER` [50] is used to identify argument units (premises and claims) which are combined in a new document for each original result. These new documents are then indexed with BM25 (default parameters b=0.75 and k_1=1.2, document body: the set of arguments from the original document, document title: document ID from the ClueWeb12). This index is then queried with the topic titles as a disjunctive `OR`-query.

For the sake of completeness, below we provide a more detailed description of the overall most effective approach submitted to the task.

Team *Bilbo Baggins* [2] uses a two-step retrieval pipeline consisting of: (1) A query expansion to increase the recall of the candidate retrieval and (2) re-ranking the candidate documents using three feature types: relevance, credibility, and support features. Before querying ChatNoir, Bilbo Baggins expands topic titles with synonyms and antonyms from WordNet [126] for entities (e.g., laptop, desktop, etc.) and comparison aspects (e.g., better, best, etc.) detected with spaCy. Then, ChatNoir is queried with four queries for each topic: (1) The original topic title, (2) all identified entities as one conjunctive `AND`-query, (3) all entities and comparison aspects as one disjunctive `OR`-query, and (4) all entities, aspects, their synonyms, and antonyms as one disjunctive `OR`-query. The set of the top-30 results of each of these four queries is then re-ranked using: (1) "relevance features" (PageRank, number of comparative sentences as identified by an XGBoost classifier [49] with InferSent embeddings [55] as proposed by Panchenko et al. [137] and argument ratio [154]), (2) document "credibility" (SpamRank score returned by ChatNoir), and (3) "support" features (number of sentences that support claims [155]), where features are respective numerical scores. The final ranking is created over the sums of the scores multiplied by the weighting values set heuristically.

Conclusions

In the first task year, the relatively "simple" argumentation-agnostic BM25-based baseline was on par with the most effective, more sophisticated participants' approaches. However, some minor improvements were achieved by using query expansion, argumentativeness assessment, and the identification of comparative textual features in documents, but there still is room for future improvements. Further research on argument retrieval thus seems well-justified. In future task editions, the participants will be able to use the first year's relevance judgments to develop and fine-tune new approaches. As we shall see in the next section, this indeed allows the task participants to improve even more over the baseline retrieval results.

6.3.2 Survey of Submissions at Touché 2021

In the second edition of the task, we used the same document collection for retrieval, ClueWeb12, but formulated 50 new search topics. Additionally, the manual relevance judgments from the previous task year were available for the participants. We again asked the task participants to retrieve relevant documents that comprise convincing argumentation of high rhetorical quality for or against one comparison option or the other.

In the second year, six teams submitted their results (19 runs plus one baseline run) that all again used ChatNoir for initial document retrieval. The results of the runs with the highest nDCG@5 scores per participating team are reported in Table 6.3. The baseline run *Puss in Boots* was the same ChatNoir [23] retrieval as in the first task year.

All the task participants used relevance judgments from Touché 2020 to train classifiers or to optimize models' parameters. Majority of the participating teams use various query processing and expansion techniques (5 out of 6; team Katana [46] does not do any query processing). The most common methods include query stop word and punctuation removal, lemmatization (all teams), and expanding comparative query terms (e.g.,

Table 6.3: Results for the task on comparative argument retrieval in 2021. The left part (a) shows the evaluation results of a team's best run according to the results' relevance, while the right part (b) shows the best runs according to the results' argument quality. An asterisk (*) indicates that the runs with the best relevance and the best quality differ for a team. The baseline ChatNoir ranking is shown in bold. None of the results are significantly better compared to the baseline approach (paired Student's t-test, $p = 0.05$, Bonferroni correction).

(a) Best relevance score per team

Team	**nDCG@5**	
	Relevance	Quality
Katana* [46]	0.489	0.675
Thor [176]	0.478	0.680
Rayla* [8]	0.473	0.670
Jack Sparrow [206]	0.467	0.664
Mercutio [80]	0.441	0.651
Puss in Boots [23]	**0.422**	**0.636**
Prince Caspian (no paper)	0.244	0.548

(b) Best quality score per team

Team	**nDCG@5**	
	Quality	Relevance
Rayla* [8]	0.688	0.466
Katana* [46]	0.684	0.460
Thor [176]	0.680	0.478
Jack Sparrow [206]	0.664	0.467
Mercutio [80]	0.651	0.441
Puss in Boots [23]	**0.636**	**0.422**
Prince Caspian (no paper)	0.548	0.244

adjectives and adverbs in a comparative form) with synonyms or antonyms using WordNet [125] and word2vec [124] (Jack Sparrow [206], Mercutio [80], Rayla [8], Thor [176]) or generated queries from scratch using GPT-2 [150] (team Mercutio).

All the teams re-rank the ChatNoir's initially ranked results using the following features, including: (1) A document argumentativeness score, e.g., a ratio of premises and claims in documents (Jack Sparrow, Mercutio, Rayla), (2) an approximation of a document trustworthiness using PageRank and SpamRank scores (Jack Sparrow, Rayla), (3) a document comparativeness score, e.g., number of comparison objects, aspect, and predicates (Katana), (4) a cosine similarity score between a query and argumentative sentences in documents using SBERT embeddings (Rayla), or (5) simple tf-idf weighted 1- to 4-grams (Prince Caspian). The documents then are re-ranked using a linear combination of weighted features (Mercutio, Rayla), XGBoost, LightGBM, and Random Forests classifiers (Katana), SVM (Jack Sparrow), and logistic regression (Prince Caspian). Relevance judgments from Touché 2020 are used to train the respective classifiers. Team Thor [176] proposes a different approach built on the idea by Huck [82] from the first Touché edition. They create a new document index based on the premises and claims mined from the initially retrieved documents with ChatNoir, use query processing and expansion, and optimize BM25 parameters by a grid search on the Touché 2020 judgments.

For the sake of completeness, below we provide a more detailed description of the overall relevance-wise and argument quality-wise most effective approaches submitted to the task.

Team *Katana* [46] re-ranks the top-100 initial ChatNoir results (original questions as queries) using different feature-based and neural classifiers or rankers to predict the final relevance labels: (1) an XGBoost [49] classifier (overall relevance-wise most effective run), (2) a LightGBM [90] classifier (team Katana's quality-wise best run), (3) Random Forests [38], and (4) a BERT-based ranker from OpenNIR [121]. The feature-based approaches are trained on the relevance judgments from Touché 2020, employing a range of relevance features (e.g., ChatNoir relevance score) and comparativeness features (e.g., the number of identified comparison objects, aspects, and predicates using the classifier proposed by Chekalina et al. [48]). The BERT-based ranker is trained on the ANTIQUE question-answering dataset that contains 34,000 text passages with relevance annotations for 2,600 open-domain non-factual questions [79]. A total of six runs were submitted by the team.

Team *Rayla* [8] uses two query processing / expansion techniques: (1) Removing stop words and punctuation, and then lemmatizing the remaining tokens with spaCy, and (2) expanding adjectives and adverbs in a comparative form (POS-tagged with spaCy) with a maximum of five synonyms and antonyms. The final re-ranking is created by linearly combining different scores such as a ChatNoir's relevance score, PageRank, and SpamRank (both also returned by ChatNoir), an argument support score (ratio of argumentative sentences (premises and claims) in documents found with a custom DistilBERT-based [165] classifier), and a similarity score (averaged cosine similarity between the original query and every argumentative sentence in the document represented with SBERT embeddings [154]). The weights of the individual scores are optimized on the Touché 2020 topics and judgments. A total of four runs were submitted.

Conclusions

In the second task year, most of the participating teams used the judgments from the first task year to fine-tune their re-ranking pipelines. Overall, the majority of the participating teams could improve upon the argumentation-agnostic BM25 baseline (even though none of the results were significantly better), indicating that some progress was achieved. The most successful methods not only estimated the argumentativeness and comparativeness of documents but also addressed the argument quality dimensions such as credibility and argumentative support. These approaches form the basis for the next, final iteration of the shared task. As we shall see in the next section, submitted approaches that build on the previous Touché results are even more effective for argument retrieval.

6.3.3 Survey of Submissions at Touché 2022

While the first two Touché editions focused on the retrieval of complete web documents, the third edition focused on text passages. The task was: Given a collection of text passages and a comparative topic with two comparison objects, retrieve relevant argumentative passages of high argument quality for or against one or both objects, and detect the passages' stances with respect to the objects. We provided 50 topics (selected from the 100 topics from the previous task years) that describe scenarios of personal decision making, extending the topics with a pair of comparison objects that could be used for the stance detection of the retrieved passages.

Document Collection

The retrieval collection in 2022 was a corpus containing 868,655 passages extracted from ClueWeb12 (different from the previous task editions). We constructed this passage corpus using all 37,248 documents from the top-100 pool of all runs submitted to the task in the previous Touché editions. Using the TREC CAsT tools,[8] we split the documents at sentence boundaries into fixed-length passages of approximately 250 terms, since ranking fixed-length passages has been shown to be more effective than that of variable-length passages [88]. From the initial 1,286,977 passages, we removed near-duplicates with CopyCat [69] to mitigate unwanted side-effects of near-duplicates on retrieval effectiveness [67, 68], resulting in the final collection of 868,655 passages. We also provided the participants with a second version of the corpus, in which the passages were expanded with queries generated by the docT5query model [135].

Participant Approaches

Seven teams submitted their results to the task (25 valid runs). Interestingly, only two participating teams used relevance judgments from the previous task editions to fine-tune their models or to optimize parameters. The others either manually labeled a sample of retrieved documents

[8] https://github.com/grill-lab/trec-cast-tools

themselves or relied on zero-shot approaches like the transformer-based model T0++ [166]. Most teams used the standard passage collection, but two teams also used the docT5query-expanded collection provided by us. Overall, the main trend of that year was the usage of transformer-based models for ranking and re-ranking (e.g., ColBERT [93] or monoT5 and duoT5 [146]), while our baseline approach was BM25, as in the previous years. For the optional subtask of stance detection, five of the seven teams submitted results. They either trained their own classifiers on the provided stance dataset (described in Section 4.1) or directly used pre-trained models as zero-shot classifiers. Our baseline stance detector was a simple always-'no stance' predictor (majority class).

Table 6.4 shows the results of each team's most effective runs with respect to the topical relevance and argument rhetorical quality. For stance detection, for each team, we evaluated all passages that were part of the manual judgment pool (top-5 pooling) and for which the team had predicted the stance (i.e., the stance of a passage returned at Rank 3 by some Team X (and thus part of the judgment pool) was also used in the stance evaluation of Team Y, even when the document was only on Rank 6 or lower (and thus not actually part of the pool for that run). Note that this yields different numbers of passages used for the stance evaluation per team. Below, we briefly describe the participating teams' submitted approaches.

Puss in Boots is our baseline retrieval model that uses the BM25 implementation in Pyserini [115] with default parameters ($k_1 = 0.9$ and $b = 0.4$) and original topic titles as queries. The baseline stance detector simply assigns 'no stance' to all documents in the ranked list.

In the third task year, again almost all the teams (exception is the team Katana [47]) used query processing and expansion techniques analogous to the approaches submitted in the second year, including removing stop words and punctuation, lemmatization, stemming, and expanding with synonyms or antonyms of comparison objects, aspects, and predicates. A few new techniques include adding extra terms using pseudo-relevance feedback and using queries generated with docT5query (teams Captain Levi [152], Aldo Nadi [1]), query expansion using terms from the LDA-

Table 6.4: Results for the task on comparative argument retrieval in 2022. (a) Evaluation results of a team's best run according to the results' topical relevance. (b) Best runs according to the results' argument quality. (c) Stance detection results (the teams' ordering is the same as in (b)). An asterisk (*) indicates that the runs with the best relevance and the best quality differ for a team. The baseline BM25 ranking is shown in bold; the baseline stance detector always predicts 'no stance'. A (†) indicates statistically significantly better results compared to the baseline (paired Student's t-test, $p = 0.05$, Bonferroni correction). Since stance detection results were calculated for different numbers of predictions for each team, we do not test the statistical significance of the differences.

(a) Best relevance score per team

Team	**nDCG@5**	
	Rel.	Qual.
Captain Levi [152]	0.758†	0.744
Aldo Nadi* [1]	0.709†	0.748
Katana* [47]	0.618†	0.643
Captain Tempesta* [53]	0.574†	0.589
Olivier Armstrong [151]	0.492	0.582
Puss in Boots (BM25)	**0.469**	**0.476**
Grimjack [153]	0.422	0.403
Asuna [160]	0.263	0.332

(b) Best quality score per team

Team	**nDCG@5**	
	Qual.	Rel.
Aldo Nadi* [1]	0.774†	0.695
Captain Levi [152]	0.744†	0.758
Katana* [47]	0.644†	0.601
Captain Tempesta* [53]	0.597†	0.557
Olivier Armstrong [151]	0.582	0.492
Puss in Boots (BM25)	**0.476**	**0.469**
Grimjack [153]	0.403	0.422
Asuna [160]	0.332	0.263

(c) Stance

F1 macro	
Rank	Score
	—
1	0.261
3	0.220
	—
4	0.191
5	**0.158**
2	0.235
6	0.106

generated topics (team Asuna [160]), and using newly generated queries with T0++ (team Grimjack [153]).

Since for this task iteration, we provided a new corpus of text passages, all the teams index the corpus themselves. Even though BM25 is again the first choice for initial ranking, some participants also use DirichletLM [222] (team Grimjack) and neural ranking models like ColBERT [93] (team Katana) and TCT-ColBERT [116] (a variant of ColBERT with knowledge distillation) followed by a re-ranking using monoT5 and duoT5 [146] (team Captain Levi). To create final ranked lists, the teams exploit the following strategies: (1) Combining relevance scores with predicted argument quality scores (e.g., using the IBM Project Debater API [18] and DistilBERT [165] fine-tuned on the Webis-ArgQuality-20 corpus [72]) either by multiplying the scores or using a Reciprocal Ranking Fusion [57] (Aldo Nadi, Asuna), (2) multiplying the relevance scores by the number of linguistic properties of text such as a non-informative symbol frequency (hashtags, emojis, etc.) and adjective as well as comparative adjective frequencies (Captain Tempesta [53]), and (3) considering a ratio of premises and claims in documents (Olivier Armstrong [151], Asuna), spam score (Asuna), and averaged cosine similarity between the original query and every premise and claim (Olivier Armstrong). Team Grimjack uses axiomatic re-ranking based on the argumentativeness axioms that "prefer" documents with more premises and claims [24, 30], newly proposed comparativeness axioms that "prefer" documents with more comparison objects or their earlier occurrence in premises and claims, or changing document positions based on the predicted stance, such as the 'pro first object' document is followed by the 'pro second object' followed by 'neutral' stance. Team Katana fine-tune a pre-trained ColBERT model on the relevance and quality judgments from the previous Touché editions.

Five teams also predicted the document stance. Team Captain Levi uses a RoBERTa-Large-MNLI model [118] pre-trained on the Multi-Genre Natural Language Inference corpus [208], whereas team Katana uses a pre-trained XGBoost classifier that is part of the comparative argumentative machine [137, 169]. Team Olivier Armstrong trains an LSTM-based neural network with one hidden layer on the stance dataset provided by us, and

Asuna fine-tunes DistilBERT on the same dataset. Team Grimjack predicts the stance in zero-shot settings using the T0++ model [166].

For the sake of completeness, below we provide a more detailed description of the overall relevance-wise and argument quality-wise most effective approaches submitted to the task.

Team *Captain Levi* [152] submitted the relevance-wise most effective run. They first retrieve 2,000 documents using Pyserini's BM25 [115] ($k_1 = 1.2$ and $b = 0.68$) by combining top-1000 results for the original query (topic title) with the results for modified queries, where they use alternative strategies: (1) Only removing stop words (using the NLTK [25] stop word list), (2) replacing comparative adjectives with synonyms and antonyms found in WordNet [125], (3) adding extra terms using pseudo-relevance feedback, (4) using queries generated with the docT5query model [135] provided by us. Queries and corpus are also processed by using stop words and punctuation removal and lemmatization (WordNet lemmatizer). The initially retrieved results are then re-ranked using monoT5 and duoT5 [146]. Additionally, TCT-ColBERT [116] (a variant of ColBERT [93] with knowledge distillation) is also used for initial ranking for unmodified queries (topic titles). Captain Levi submitted in total five runs that differ in the aforementioned strategies of modifying queries, initial ranking models, and final re-ranking models. Their most effective run in terms of relevance and quality is the initial ranking with TCT-ColBERT. Finally, stance is detected using a RoBERTa-Large-MNLI model [118], pre-trained on the Multi-Genre Natural Language Inference corpus [208] without further fine-tuning in two steps: (1) Detecting if the document has a stance, and then (2) for documents that were not classified as 'neutral' or 'no stance', detecting which comparison object the document favors. This stance detector achieved the highest macro-averaged F1 score.

Team *Aldo Nadi* [1] submitted the quality-wise most effective run. They re-rank passages that are initially retrieved with BM25F [157] (default Lucene implementation with $k_1 = 1.2$ and $b = 0.75$) on two document fields: Text of the original passages and passages expanded with the docT5query-generated queries. All texts are stemmed with the Porter stemmer [142] and stop words are removed using different lists: (1) A

Snowball list [143], (2) a default Lucene stop word list, (3) a custom list containing the 400 most frequent terms in the retrieval collection, excluding the comparison objects. Queries (topic titles) are expanded using a relevance feedback method based on the Rocchio Algorithm [158]. For the final ranking, the team experiments with two re-ranking techniques (involving up to top-1000 documents from the initial results): (1) Exploiting the argument quality estimation, i.e., they multiply the document relevance and the quality scores, and (2) Reciprocal Ranking Fusion [57]. The argument quality scores are predicted using the IBM Project Debater API [18]. Aldo Nadi submitted five runs, which vary by different combinations of the proposed methods, e.g., using different stop word lists for pre-processing, using relevance feedback or not, and using the quality-based re-ranking or fusion. The team's most effective run in terms of relevance exploits the relevance feedback, and the most effective run in terms of quality is based on Reciprocal Ranking Fusion. The team does not detect the document stance.

6.3.4 Post Hoc Stance Detection with RoBERTa and GPT-3

Since the overall effectiveness of stance detection across all approaches is rather poor (cf. Table 6.4), we decided to test our most effective stance classifiers described in Chapter 4: sentiment-prompted RoBERTa (cf. Section 4.3.5) and GPT-3 (cf. Section 4.3.6). We thus first fine-tune RoBERTa on the full labeled dataset (cf. Section 4.1) with the masked-object input as `[OBJECT_1] is good [SEP] answer` and then classify all 2,107 manually labeled text passages from the task. While the classifier achieves a higher macro-averaged F1 of 0.34 than the best participant stance detector (macro-averaged F1 of 0.26), its accuracy is lower compared to our results in Chapter 4 (0.37 vs. 0.63). One possible reason is a different class distribution: the dataset used for experiments in Chapter 4 contains just 7 % of the 'no stance' labels, while in the Touché data 'no stance' is a majority class (48 %).

Further, we test GPT-3 using the same default model hyperparameters and the same few-shot prompting strategy as in our previous experiments described in Section 4.3.6. Predicting the stance of all 2,107 manually labeled text passages, GPT-3 achieves a macro-averaged F1 of 0.49 (accuracy

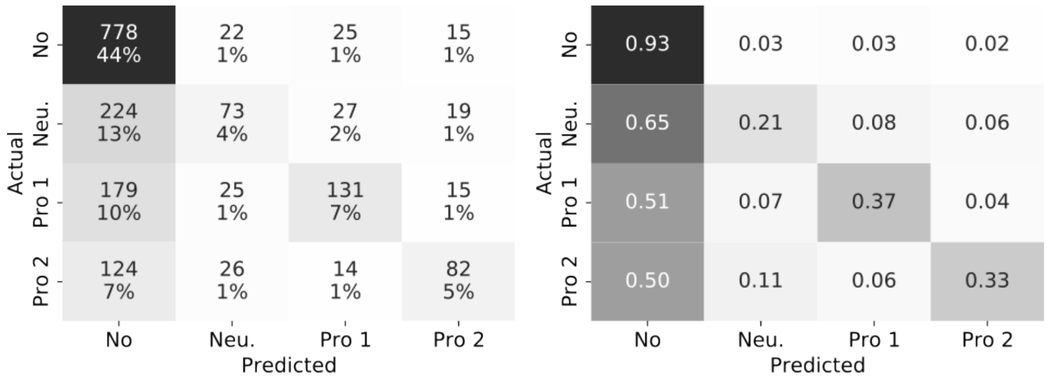


Figure 6.1: Confusion matrices: without normalization (left) and with normalization by a class support size, i.e., a number of elements in each class (right).

of 0.6), outperforming all the participants' results. The majority class is 'no stance' (48 %) followed by 'neutral' (20 %) and 'pro first object' (19 %), with the minority 'pro second object' class (13 %). Interestingly, while in our previous experiments described in Section 4.3.6, GPT-3 tended to often mistakenly predict the 'neutral' class, now it more frequently predicts a wrong 'no stance' class (cf. confusion matrices in Figure 6.1). Since neural models are known to be more like "black-boxes", understanding the reason for such a difference in predictions is rather difficult.

Conclusions

In the third task year, many more participants were able to build argument retrieval approaches for comparative information needs that were more effective than the argumentation-agnostic BM25 baseline in terms of topical relevance and argument quality. In addition to sparse retrieval and various query processing, reformulation, and expansion methods, the proposed approaches have increasingly focused on transformer-based models and re-ranking techniques. An interesting observation is that re-ranking first-stage retrieval results based on the quality assessment of arguments almost always improves the retrieval effectiveness. Also, re-ranking based on important comparative terms such as comparison objects and aspects or argument units in documents (premises and claims) was successful. The stance detection was a new subtask, and one participating team included a re-ranking step based on the predicted stance in the retrieval pipeline, which had some promising effects on improving the retrieval effectiveness. However, the overall still rather low effectiveness of the stance detection approaches leaves room for future improvements.

6.4 Summary

In this chapter, we provided an overview of the Touché shared task on argument retrieval for comparative questions that we organized for three years. The task's goal was to support answering comparative questions in personal decision-making situations by developing retrieval approaches that address the argument's topical relevance, argument quality, and stance towards the to-be-compared options. In pursuit of understanding what methods can enhance the effectiveness of argument retrieval for comparative questions, we analyzed the proposed retrieval pipelines from 18 task participants and evaluated in total 54 submitted result rankings (plus the task baseline approaches). Comparing the participants' results with the argumentation-agnostic BM25 baseline, we have observed how the submitted approaches evolved from being no better than the baseline to the majority developing retrieval pipelines that are more effective. While in the first task year, no labeled data was provided to the participants, in the later task iterations, participating teams used manual judgments (relevance and quality of arguments) to train and optimize their pipelines. In addition to more traditional retrieval models like BM25, (re-)ranking approaches such as the recent transformer-based models have been applied. Other successful re-ranking strategies used combining the topical relevance with the document "argumentativeness" score, predicted argument quality, or stance.

For the web document retrieval task (first two task iterations), the relevance-wise most effective approach was to re-rank BM25 results using an XGBoost classifier trained with the features such as BM25 relevance score and comparativeness features like the number of comparison objects, aspects, and predicates. Whereas the argument quality-wise most effective approach used query expansion and re-ranking based on the argument ratio score, predicted argument quality, and similarity over SBERT embeddings. As for the passage retrieval task (third task iteration), a TCT-ColBERT ranker (in terms of relevance) and a combination of BM25 relevance scores and predicted argument quality (in terms of quality) were the most effective. Overall, the most effective argument retrieval approaches for comparative questions used various strategies for query reformulation and expansion and exploited re-ranking based on the estimation of argument quality or document "argumentativeness". Detecting the stance towards the comparison objects remains challenging (the highest macro-averaged F1 score of 0.26 was achieved by the RoBERTa-based classifier). Predicting the stance using GPT-3 achieves a macro-averaged F1 of 0.49. The manual annotations created at Touché can be used in future work for training stance detectors to improve their effectiveness.

The task results, corpora, topics, and manual judgments created at Touché are freely available to the research community and can be found on the shared tasks' website.[9] Parts of the data are also available via the BEIR [193] and `ir_datasets` [122] resources. These test collections and initial findings of the Touché shared tasks provide the ground for further research in argument retrieval for answering comparative questions.

[9] https://touche.webis.de/data.html

7
Conclusion

This dissertation has addressed a specific type of information needs on the Web—comparisons—that are often formulated as comparative questions, which people may ask when seeking solutions to decision-making tasks. In this chapter, I will conclude this dissertation and overview its main findings and contributions. In the subsequent sections, I will first wrap up the main contributions in Section 7.1, and then in Section 7.2, I will discuss the remaining open questions and future research directions.

7.1 Main Findings and Contributions

Chapter 3 contributed the analysis of comparative questions that were asked online in question and answer fora and that were submitted to a search engine. By manually labeling questions fetched from the fora and search engine query logs as comparative and by analyzing the comparative questions, we developed a taxonomy of comparative information needs comprising five categories from existing question taxonomies like factual and opinion questions and five new categories that are specific to comparative questions like direct comparisons and questions with comparison aspects. This categorization then provides the ground for the next steps of answering comparative questions. For instance, if a question is recognized as subjective (i.e., asking for opinions or arguments in answers), the result presentation may aggregate and show side-by-side arguments that support one or the other comparison option from the question. This in turn requires the search system to include argument analysis steps like identifying argumentative texts or detecting the argument stance. However, the first step of the system that addresses answering comparative questions will be identifying and categorizing comparative information needs.

Accordingly, a further contribution of Chapter 3 is the approaches to identifying comparative information needs and analyzing their different types: We proposed classifiers that distinguish comparative questions from others and that classify comparative questions into fine-grained categories. With the idea in mind of changing the result presentation of search systems (e.g., web search engines) for comparative questions, we aimed for building a precision-oriented classifier that reliably identifies comparative information needs. Inspired by the work in linguistics that studied comparison structures, we first handcrafted a set of lexico-syntactic classification rules. These rules exploit textual cues that indicate the presence of comparatives in questions like adjectives and adverbs in a comparative form. As evaluation showed, the rules can accurately classify about half of the comparative questions in our labeled datasets at perfect precision of 1.0. To further increase the recall, we complemented the rules with feature-based and neural classifiers. Again, focusing on the precision of classifying comparative questions, we selected the operating points of the classification models (in terms of their decision probability thresholds) such that they always predict the class of comparative questions with a precision of 1.0. At each consequent classification step, we trained classifiers only on the (more difficult) examples that were not captured in the preceding step. When combined in cascading ensembles, the classifiers were able to recall 60 % of Russian and 71 % of English comparative questions at perfect precision. Moreover, using the CNN-based and BERT-based classifiers, we could reliably classify comparative questions into seven fine-grained categories with convincing micro-averaged F1 scores of about 0.9.

Furthermore, in Chapter 3 we also analyzed comparative questions identified by our classifiers in the archived year-long Yandex log. This analysis aimed for a better understanding of how frequently comparative questions can be received by search engines and what types of such questions are often being asked. Our analysis showed that at least 3 % of question queries search engines receive may be comparative. The majority of them are clearly non-factual, about half of them do not explicitly specify the to-be-compared options, and more than 70 % do not contain comparison aspects. These findings motivated further research goals that constituted further contributions of this dissertation. In particular, these included bringing argument analysis methods such as stance detection into the pipeline for answering non-factual comparative questions or clarification approaches for ambiguous comparative information needs (i.e., questions without explicit comparison objects or questions without comparison aspects).

Consequent Chapter 4 addressed the tasks of parsing comparative questions and stance detection of answers to comparative questions: We elaborated on what constituents of comparative questions are important for searching and presenting answers, proposed approaches to parse questions by tagging these important question terms, and investigated methods to detect the stance of potential answers to comparative questions.

The first contribution of Chapter 4 is a dataset of comparative questions manually annotated on the token level with comparison objects, aspects, and predicates. Additionally, human-written answers fetched from the question and answer fora for a subset of comparative questions were labeled with the stance towards the comparison objects as supporting one or the other object. To parse comparative questions, we trained transformer-based token-level classifiers that achieved convincing F1 scores of more than 0.9 for identifying the comparison objects and predicates. Identifying the comparison aspects was the most challenging, with F1 reaching just 0.8. We then showed that if we were to tag the aspects in questions that were known to contain aspects (manually labeled with the category 'with an aspect') F1 score increased by 0.1. However, developing a high-precision classifier that identifies questions with aspects was challenging. This finding provided an additional argument to study clarification for the cases when the automatic identification of the comparison aspects is imprecise.

The final contribution of Chapter 4 was the stance detector that, given a comparative question with two comparison objects, classifies a text passage as being pro first object, pro second object, neutral, or whether no stance is entailed. Some challenges for such a stance detector are, e.g., there are two different stance targets: the comparison objects, and that the answer (text passage) may contain different stances in different parts. Indeed, our most effective classifier achieved an accuracy of just 0.63. Our experiments showed that successful techniques were to fine-tune transformer-based models with the objects expanded with sentiment prompts and identifying and substituting the objects with special masking tokens. We then tested GPT-3 as a stance detector that, when few-shot prompted, achieved an accuracy of 0.65, leaving room for future improvements.

Next, Chapter 5 subsequently addressed the challenges identified in the previous chapters, e.g., that the majority of comparative questions did not explicitly mention the to-be-compared objects or comparison aspects and the challenges to identify and tag the aspects in questions. We thus investigated clarification interactions that a search system can use to refine initial ambiguous comparative information needs. In particular, Chapter 5 contributed a user study, in which we asked the study participants to perform comparative searches and to compare the search results obtained with and without clarification. The user study was designed for the cases when the original questions did not contain the comparison aspects and when both the comparison objects and aspects were unclear. The study results unequivocally indicated that clarifications not only helped the searchers to find more satisfactory results for comparative search requests but also were pleasant to use. These findings overall confirmed the results of previous works that studied the usefulness of clarification in web search.

Finally, Chapter 6 is dedicated to the task of retrieving documents containing high-quality argumentation that can be useful for answering subjective comparative questions. We have introduced and described the shared tasks on argument retrieval for comparative questions that we organized for three consecutive years. Our main goals were to solicit research activities in developing new ideas for argument retrieval, to create test collections, and analyze and evaluate participants' developed approaches.

To understand what methods are important for argument retrieval for comparative questions, we evaluated and analyzed a total of 54 result rankings from 18 task participants over three years. Each year, we provided 50 test search topics that described comparative searches represented by comparative questions like "Should I major in philosophy or psychology?". Given a collection of documents (web documents from a web crawl or text passages), we asked the participants to retrieve and rank topically relevant documents that contained high-quality argumentation. In the third task year, we additionally asked the participants to detect the stance of the top-ranked text passages towards the comparison objects.

We compared the participants' submitted ranking results to the task with the argumentation-agnostic BM25 baseline retrieval (we calculated nDCG@5 using the relevance and argument quality manual judgments). Result evaluation showed that in the first task year, none of the participating teams could develop a more effective retrieval pipeline than the baseline. One possible reason is that no training data was available at the beginning. In the later task iterations, the majority of participants could already improve over the baseline; they used manual judgments to train and optimize their retrieval pipelines. Overall the most successful solutions combined the relevance scores, often obtained with BM25 used for first-stage retrieval, with: (a) Comparativeness scores and features like a number or a ratio of comparison objects, aspects, and predicates in documents, (b) Argumentativeness scores and features such as a ratio of argument units (e.g., premises and claims) in documents, and (c) predicted argument quality scores, for which supervised classifiers were trained. Detecting the stance turned out to be the most challenging task. And since we organized the task only once, we did not have the chance to see the improvement over time. For stance detection, our participants "simply" trained various classifiers: RoBERTa was one of the most effective among them. In the post hoc evaluation, using GPT-3 as a stance detector almost doubled a macro-averaged F1 compared to the "best" participants' classifier (0.49 vs. 0.26).

7.2 Open Questions and Future Work

We now will conclude this chapter and the dissertation at hand, will discuss open questions, comment on the limitations of the presented results and contributions of this dissertation, and elaborate on potentially interesting follow-up research directions.

While discussing different types of comparative information needs in Chapter 3, we justified the need to differentiate between different question categories like subjective versus factual questions. We looked at these categories from two perspectives: Whether the result presentation to the searcher should be different (e.g., for comparative questions vs. others or factual comparative questions vs. subjective ones) and whether the search for answers on the system's side should be different. For instance, existing works in computational argumentation (in particular, in argument search) have already emphasized the benefits for users of presenting diverse viewpoints in terms of pro and con arguments for debated topics like climate change, making search results more diverse and possibly less biased. The still open question is whether other types of comparative questions like indirect (i.e., questions without explicit comparison objects) or without comparison aspects need different ways of result presentation. Thus, an interesting follow-up research avenue can be to compare a more conventional way of showing a list of some "best" options for questions like "What are the best ...?" with presenting side-by-side the "most popular" items comparing them over their different features (i.e., comparison aspects).

Evaluation results described in Chapter 4 showed that the stance detection towards the comparison objects is challenging. Thus, improving the effectiveness of stance detectors for comparative answers is a worthwhile research direction to pursue. One rather straightforward way to address this is to expand the labeled data used for training and testing classifiers. Another interesting insight from our experiments was that different configurations of the stance detectors (e.g., different transformer models or sentiment prompts) were most effective for different stance classes. Thus, interesting future experiments could be combining predictions of various classifiers in an ensemble. We then also tested GPT-3 as a stance detector. When few-shot prompted, it was slightly more effective than our best stance detector configuration. Further prompt tweaking may improve even more the effectiveness of stance detection with GPT-3. Although using foundation models for classification tasks can be promising and has been changing the research landscape, there are several aspects that should be considered. Specifically GPT-3 is (currently) a paid API-based service hosted by a third party, which raises several issues and concerns. For instance, during our experiments, we regularly experienced the service unavailability error. Also, when working with privacy-sensitive data (e.g., in financial institutions), using external services might be problematic. Additionally, the environmental impact of training and running large models should be considered. We thus suggest considering all the advantages and disadvantages of different approaches when choosing a classification model.

As discussed in Chapter 3, a comparison of several to-be-compared options is performed over their shared properties: comparison aspects. To this end, our stance detection experiments considered only the comparison objects. Thus, the *aspect-based* stance detection is an interesting research avenue. Intuitively, one comparison item can be *better* at one aspect, but *worse* at another. While aspect-based sentiment analysis is an active research field, aspect-based stance detection has been largely overlooked.

Our first results of studying clarification for ambiguous comparative information needs indicated its usefulness for finding more satisfactory answers. We, however, suggested in Chapter 5 that a larger study in terms of the number of study cases and participants is needed. This will allow a more robust conclusion that can confirm or reject our preliminary findings. Since in our user study, we simulated a clarification interface, another interesting avenue to pursue as future work is to develop the actual approaches to generate clarifying questions and clarification options.

In Chapter 6, we described the shared tasks on argument retrieval for comparative questions that we organized for three years. By analyzing participants' submitted solutions to the task, we observed the progress of the results from being on par with the argumentation-agnostic BM25 baseline retrieval to the majority of the proposed approaches achieving higher effectiveness. This was largely achieved by including components that assess document argumentativeness and estimate the argument quality of retrieved documents. One aspect of argument retrieval that we have not investigated so far is viewpoint diversity. In particular, for the cases when two options are compared, it can be interesting to investigate: (a) If the distribution of pro, con, or neutral (towards the comparison objects) documents in the top *k* retrieved results influences the user satisfaction in some way, (b) whether any kind of diversification is actually needed (somewhat similar to the idea of ranking fairness in information retrieval), and (c) what stance diversification methods for result rankings can be effective.

As discussed in this dissertation, comparative questions often represent the need to come to an informed decision by choosing one or another "better" option. This decision making process is accompanied by collecting new information about the options under consideration in the form of facts, opinions, or arguments. While the quality of an answer to factual comparisons like whether one mount is higher than the other may be insignificant, the quality of information for life-changing decisions, be it a university to choose or a treatment to undergo, is crucial. This dissertation highlighted the importance of treating comparative information needs by search systems with more care and attention. Starting with the identification and better understanding of comparative information needs and up to analyzing the quality of found information such as argument quality and stance, I proposed building blocks for a web search for comparisons.

References

[1] Maria Aba, Munzer Azra, Marco Gallo, Odai Mohammad, Ivan Piacere, Giacomo Virginio, and Nicola Ferro. Aldo Nadi at Touché 2022: Argument retrieval for comparative questions. In *Working Notes Papers of the CLEF 2022 Evaluation Labs*, pages 2904–2918. CLEF and CEUR-WS.org, 2022.

[2] Tinsaye Abye, Tilmann Sager, and Anna Juliane Triebel. An open-domain web search engine for answering comparative questions. In *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum*, CLEF and CEUR-WS.org, 2020.

[3] Eugene Agichtein, Silviu Cucerzan, and Eric Brill. Analysis of factoid questions for effective relation extraction. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and Development in Information Retrieval, SIGIR 2005*, pages 567–568. ACM, 2005.

[4] Yamen Ajjour, Henning Wachsmuth, Dora Kiesel, Patrick Riehmann, Fan Fan, Giuliano Castiglia, Rosemary Adejoh, Bernd Fröhlich, and Benno Stein. Visualization of the topic space of argument search results in args.me. In *Proceedings of the 23rd Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, pages 60–65. Association for Computational Linguistics, 2018.

[5] Yamen Ajjour, Henning Wachsmuth, Johannes Kiesel, Martin Potthast, Matthias Hagen, and Benno Stein. Data acquisition for argument search: The args.me corpus. In *Proceedings of the 42nd German Conference on Artificial Intelligence, KI 2019*, pages 48–59. Springer, 2019.

[6] Yamen Ajjour, Pavel Braslavski, Alexander Bondarenko, and Benno Stein. Identifying argumentative questions in web search logs. In *Proceedings of the 45th International ACM Conference on Research and Development in Information Retrieval, SIGIR 2022*, pages 2393–2399. ACM, 2022.

[7] Icek Ajzen. The social psychology of decision making. *Social psychology: Handbook of basic principles*, pages 297–325, 1996.

[8] Alaa Alhamzeh, Mohamed Bouhaouel, Elöd Egyed-Zsigmond, and Jelena Mitrovic. Distilbert-based argumentation retrieval for answering comparative questions. In *Proceedings of the Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum*, pages 2319–2330. CLEF and CEUR-WS.org, 2021.

[9] Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W. Bruce Croft. Asking clarifying questions in open-domain information-seeking conversations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019*, pages 475–484. ACM, 2019.

[10] Keith Allan. Interpreting English comparatives. *Journal of Semantics*, 5(1):1–50, 1986.

[11] Giambattista Amati. Frequentist and Bayesian approach to information retrieval. In *Proceedings of the 28th European Conference on IR Research, ECIR 2006*, pages 13–24. Springer, 2006.

[12] Shinya Aoki, Takayuki Yumoto, Manabu Nii, and Yutaka Takahashi. Searching for comparison points between two objects from the Web. In *Proceedings of the 3rd International Universal Communication Symposium, IUCS 2009*, pages 344–349. ACM, 2009.

[13] Aristotle and George A. Kennedy. *On rhetoric: A theory of civic discourse*. Oxford: Oxford University Press, 2006.

[14] Jatin Arora, Sumit Agrawal, Pawan Goyal, and Sayan Pathak. Extracting entities of interest from comparative product reviews. In

Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, pages 1975–1978. ACM, 2017.

[15] Ahmed Hassan Awadallah, Xiaolin Shi, Nick Craswell, and Bill Ramsey. Beyond clicks: query reformulation as a predictor of search satisfaction. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management, CIKM 2013*, pages 2019–2028. ACM, 2013.

[16] Omid Bakhshandeh Babarsad. *Language learning through comparison*. PhD thesis, University of Rochester, 2017.

[17] Roy Bar-Haim, Indrajit Bhattacharya, Francesco Dinuzzo, Amrita Saha, and Noam Slonim. Stance classification of context-dependent claims. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017*, pages 251–261. Association for Computational Linguistics, 2017.

[18] Roy Bar-Haim, Yoav Kantor, Elad Venezian, Yoav Katz, and Noam Slonim. Project Debater APIs: Decomposing the AI grand challenge. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP 2021*, pages 267–274. Association for Computational Linguistics, 2021.

[19] Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. *CoRR*, abs/2004.05150, 2020.

[20] Michael Bendersky and W. Bruce Croft. Analysis of long queries in a large scale search log. In *Proceedings of the 2009 workshop on Web Search Click Data, WSCD@WSDM 2009*, pages 8–14. ACM, 2009.

[21] Polina Berezovsakaya and Vera Hohaus. The crosslinguistic inventory of phrasal comparative operators: Evidence from Russian. In *Proceedings of Formal Approaches to Slavic Linguistics, FASL 2015*, pages 1–19. Michigan Slavic Publications, 2015.

[22] Polina Berezovskaya. Acquisition of Russian comparison constructions: Semantics meets first language acquisition. In *Proceedings of*

the Conference of the Student Organisation of Linguistics in Europe, ConSOLE 2013, pages 45–65, 2013.

[23] Janek Bevendorff, Benno Stein, Matthias Hagen, and Martin Potthast. Elastic ChatNoir: Search engine for the ClueWeb and the Common Crawl. In Leif Azzopardi, Allan Hanbury, Gabriella Pasi, and Benjamin Piwowarski, editors, *Proceedings of the 40th European Conference on IR Research, ECIR 2018*, pages 820–824. Springer, 2018.

[24] Janek Bevendorff, Alexander Bondarenko, Maik Fröbe, Sebastian Günther, Michael Völske, Benno Stein, and Matthias Hagen. Webis at TREC 2020: Health Misinformation Track. In *Proceedings of the 29th International Text Retrieval Conference, TREC 2020*, NIST Special Publication. National Institute of Standards and Technology (NIST), 2020.

[25] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python*. O'Reilly, 2009. ISBN 978-0-596-51649-9.

[26] Christopher M Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[27] Johannes Bjerva, Nikita Bhutani, Behzad Golshan, Wang-Chiew Tan, and Isabelle Augenstein. SubjQA: A dataset for subjectivity and review comprehension. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 5480–5494. Association for Computational Linguistics, 2020.

[28] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5(1):0–0, 2017.

[29] Valeria Bolotova, Vladislav Blinov, Falk Scholer, W. Bruce Croft, and Mark Sanderson. A non-factoid question-answering taxonomy. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2022*, pages 1196–1207. ACM, 2022.

[30] Alexander Bondarenko, Maik Fröbe, Vaibhav Kasturia, Michael Völske, Benno Stein, and Matthias Hagen. Webis at TREC 2019: Decision Track. In *Proceedings of the 28th International Text Retrieval Conference, TREC 2019*, NIST Special Publication. National Institute of Standards and Technology (NIST), 2019.

[31] Alexander Bondarenko, Pavel Braslavski, Michael Völske, Rami Aly, Maik Fröbe, Alexander Panchenko, Chris Biemann, Benno Stein, and Matthias Hagen. Comparative web search questions. In *Proceedings of the 13th ACM International Conference on Web Search and Data Mining, WSDM 2020*, pages 52–60. ACM, 2020.

[32] Alexander Bondarenko, Maik Fröbe, Meriem Beloucif, Lukas Gienapp, Yamen Ajjour, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, and Matthias Hagen. Overview of Touché 2020: Argument Retrieval. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction. 11th International Conference of the CLEF Association, CLEF 2020*, pages 384–395, Springer, 2020.

[33] Alexander Bondarenko, Lukas Gienapp, Maik Fröbe, Meriem Beloucif, Yamen Ajjour, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, and Matthias Hagen. Overview of Touché 2021: Argument Retrieval. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction. 12th International Conference of the CLEF Association, CLEF 2021*, pages 450–467. Springer, 2021.

[34] Alexander Bondarenko, Yamen Ajjour, Valentin Dittmar, Niklas Homann, Pavel Braslavski, and Matthias Hagen. Towards understanding and answering comparative questions. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining, WSDM 2022*, pages 66–74. ACM, 2022.

[35] Alexander Bondarenko, Maik Fröbe, Johannes Kiesel, Shahbaz Syed, Timon Gurcke, Meriem Beloucif, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, and

Matthias Hagen. Overview of Touché 2022: Argument Retrieval. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction. 13th International Conference of the CLEF Association, CLEF 2022)*, pages 311–336. Springer, 2022.

[36] Alexander Bondarenko, Ekaterina Shirshakova, and Matthias Hagen. A user study on clarifying comparative questions. In *Proceedings of the 2022 Conference on Human Information Interaction & Retrieval, CHIIR 2022*, pages 254–258. ACM, 2022.

[37] Pavel Braslavski, Denis Savenkov, Eugene Agichtein, and Alina Dubatovka. What do you mean exactly?: Analyzing clarification questions in CQA. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval, CHIIR 2017*, pages 345–348. ACM, 2017.

[38] Leo Breiman. Random Forests. *Mach. Learn.*, 45(1):5–32, 2001.

[39] Joan Bresnan. Syntax of the comparative clause construction in English. *Linguistic Inquiry Volume IV*, 4(3):275–343, 1973.

[40] Andrei Z. Broder. A taxonomy of web search. *SIGIR Forum*, 36(2): 3–10, 2002.

[41] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Proceedings of the Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, 2020.

[42] John Burger, Claire Cardie, Vinay Chaudhri, Robert Gaizauskas, Sanda Harabagiu, David Israel, Christian Jacquemin, Chin-Yew Lin,

Steve Maiorano, George Miller, et al. Issues, tasks and program structures to roadmap research in question & answering (Q&A). In *Document Understanding Conferences Roadmapping Documents*, pages 1–35, 2001.

[43] Yu Cao, Meng Fang, and Dacheng Tao. BAG: Bi-directional attention entity graph convolutional network for multi-hop reasoning question answering. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019*, pages 357–362. Association for Computational Linguistics, 2019.

[44] Karol Chia-Tien Chang, Yu-Hsuan Wu, Yi-Lin Tsai, and Richard Tzong-Han Tsai. Improving iUnit retrieval with query classification and multi-aspect iUnit scoring: The IISR system at NTCIR-11 MobileClick task. In *Proceedings of the 11th NTCIR Conference on Evaluation of Information Access Technologies, NTCIR 2014*, National Institute of Informatics, 2014.

[45] Viktoriia Chekalina and Alexander Panchenko. Retrieving comparative arguments using deep pre-trained language models and NLU. In *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum*, CLEF and CEUR-WS.org, 2020.

[46] Viktoriia Chekalina and Alexander Panchenko. Retrieving comparative arguments using ensemble methods and neural information retrieval. In *Proceedings of the Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum*, pages 2354–2365. CLEF and CEUR-WS.org, 2021.

[47] Viktoriia Chekalina and Alexander Panchenko. Retrieving comparative arguments using deep language models. In *Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum*, pages 3032–3040. CLEF and CEUR-WS.org, 2022.

[48] Viktoriia Chekalina, Alexander Bondarenko, Chris Biemann, Meriem Beloucif, Varvara Logacheva, and Alexander Panchenko.

Which is better for deep learning: Python or MATLAB? Answering comparative questions in natural language. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2021*, pages 302–311. Association for Computational Linguistics, 2021.

[49] Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2016*, pages 785–794. ACM, 2016.

[50] Artem Chernodub, Oleksiy Oliynyk, Philipp Heidenreich, Alexander Bondarenko, Matthias Hagen, Chris Biemann, and Alexander Panchenko. TARGER: Neural Argument Mining at Your Fingertips. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, ACL 2019*, pages 195–200. Association for Computational Linguistics, 2019.

[51] Aarish Chhabra, Nandini Bansal, Venktesh V, Mukesh K. Mohania, and Deep Dwivedi. Obj2sub: Unsupervised conversion of objective to subjective questions. In *Proceedings of the 23rd International Conference Artificial Intelligence in Education, AIED 2022*, pages 467–470. Springer, 2022.

[52] Lydia B. Chilton and Jaime Teevan. Addressing people's information needs directly in a web search result page. In *Proceedings of the 20th International Conference on World Wide Web, WWW 2011*, pages 27–36. ACM, 2011.

[53] Alessandro Chimetto, Davide Peressoni, Enrico Sabbatini, Giovanni Tommasin, Marco Varotto, Alessio Zanardelli, and Nicola Ferro. SE-UPD@CLEF: Team Hextech on argument retrieval for comparative questions. The importance of adjectives in documents quality evaluation. In *Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum*, pages 3041–3054. CLEF and CEUR-WS.org, 2022.

[54] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *Proceedings of the 8th International Conference on Learning Representations, ICLR 2020*. OpenReview.net, 2020.

[55] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017*, pages 670–680. Association for Computational Linguistics, 2017.

[56] Gordon Cormack, Mark Smucker, and Charles Clarke. Efficient and effective spam filtering and re-ranking for large web datasets. *Information retrieval*, 14(5):441–465, 2011.

[57] Gordon V. Cormack, Charles L. A. Clarke, and Stefan Büttcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009*, pages 758–759. ACM, 2009.

[58] Johannes Daxenberger, Benjamin Schiller, Chris Stahlhut, Erik Kaiser, and Iryna Gurevych. ArgumenText: Argument classification and clustering in a generalized search scenario. *Datenbank-Spektrum*, 20(2):115–121, 2020.

[59] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019*, pages 4171–4186. Association for Computational Linguistics, 2019.

[60] Lorik Dumani and Ralf Schenkel. Quality-aware ranking of arguments. In *Proceedings of the 29th ACM International Conference on Infor-*

mation & Knowledge Management, CIKM 2020, pages 335–344. ACM, 2020.

[61] Lorik Dumani, Patrick J. Neumann, and Ralf Schenkel. A framework for argument retrieval - ranking argument clusters by frequency and specificity. In *Proceedings of the 42nd European Conference on IR Research, ECIR 2020*, pages 431–445. Springer, 2020.

[62] Adam Robert Faulkner. *Automated Classification of Argument Stance in Student Essays: A Linguistically Motivated Approach with an Application for Supporting Argument Summarization*. Dissertation, City University of New York, 2014.

[63] Yair Feldman and Ran El-Yaniv. Multi-hop paragraph retrieval for open-domain question answering. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019*, pages 2296–2309. Association for Computational Linguistics, 2019.

[64] William Ferreira and Andreas Vlachos. Emergent: A novel dataset for stance classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016*, pages 1163–1168. The Association for Computational Linguistics, 2016.

[65] Lila J. Finney Rutten, Kelly D. Blake, Alexandra J. Greenberg-Worisek, Summer V. Allen, Richard P. Moser, and Bradford W. Hesse. Online health information seeking among us adults: Measuring progress toward a healthy people 2020 objective. *Public Health Reports*, 134(6):617–625, 2019.

[66] Susannah Fox and Maeve Duggan. Health Online 2013. *Health*, 2013: 1–55, 2013.

[67] Maik Fröbe, Janek Bevendorff, Jan Heinrich Reimer, Martin Potthast, and Matthias Hagen. Sampling bias due to near-duplicates in learning to rank. In *Proceedings of the 43rd International ACM Conference on Research and Development in Information Retrieval, SIGIR 2020*, pages 1997–2000. ACM, 2020.

[68] Maik Fröbe, Jan Philipp Bittner, Martin Potthast, and Matthias Hagen. The effect of content-equivalent near-duplicates on the evaluation of search engines. In *Proceedings of the 42nd European Conference on IR Research, ECIR 2020*, pages 12–19. Springer, 2020.

[69] Maik Fröbe, Janek Bevendorff, Lukas Gienapp, Michael Völske, Benno Stein, Martin Potthast, and Matthias Hagen. CopyCat: Near-duplicates within and between the ClueWeb and the Common Crawl. In *Proceedings of the 44th International ACM Conference on Research and Development in Information Retrieval, SIGIR 2021*, pages 2398–2404. ACM, 2021.

[70] Yarin Gal and Zoubin Ghahramani. A theoretically grounded application of dropout in recurrent neural networks. In *Proceedings of the Annual Conference on Neural Information Processing Systems, NIPS 2016*, pages 1019–1027. Curran Associates, Inc., 2016.

[71] Murthy Ganapathibhotla and Bing Liu. Mining opinions in comparative sentences. In *Proceedings of the 22nd International Conference on Computational Linguistics, COLING 2008*, pages 241–248, 2008.

[72] Lukas Gienapp, Benno Stein, Matthias Hagen, and Martin Potthast. Efficient pairwise annotation of argument quality. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 5772–5781. Association for Computational Linguistics, 2020.

[73] Ameya Godbole, Dilip Kavarthapu, Rajarshi Das, Zhiyu Gong, Abhishek Singhal, Hamed Zamani, Mo Yu, Tian Gao, Xiaoxiao Guo, Manzil Zaheer, and Andrew McCallum. Multi-step entity-centric information retrieval for multi-hop question answering. *CoRR*, abs/1909.07598, 2019.

[74] Arthur C. Graesser and Natalie K. Person. Question asking during tutoring. *American educational research journal*, 31(1):104–137, 1994.

[75] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5-6):602–610, 2005.

[76] Ido Guy. Searching by talking: Analysis of voice queries on mobile web search. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, SIGIR 2016*, pages 35–44. ACM, 2016.

[77] Andreas Hanselowski, Avinesh P. V. S., Benjamin Schiller, Felix Caspelherr, Debanjan Chaudhuri, Christian M. Meyer, and Iryna Gurevych. A retrospective analysis of the fake news challenge stance detection task. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018*, pages 1859–1874. Association for Computational Linguistics, 2018.

[78] Amanul Haque, Vaibhav Garg, Hui Guo, and Munindar P. Singh. Pixie: Preference in implicit and explicit comparisons. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, ACL 2022*, pages 106–112. Association for Computational Linguistics, 2022.

[79] Helia Hashemi, Mohammad Aliannejadi, Hamed Zamani, and W. Bruce Croft. ANTIQUE: A non-factoid question answering benchmark. In *Proceedings of the 42nd European Conference on IR Research, ECIR 2020*, pages 166–173. Springer, 2020.

[80] Daniel Helmrich, Denis Streitmatter, Fionn Fuchs, and Maximilian Heykeroth. Touché Task 2: Comparative Argument Retrieval. A document-based search engine for answering comparative questions. In *Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum*, pages 2389–2402. CLEF and CEUR-WS.org, 2021.

[81] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, 1997.

[82] Johannes Huck. Development of a search engine to answer comparative queries—Notebook for the Touché Lab on Argument Retrieval

at CLEF 2020. In *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum*. CLEF and CEUR-WS.org, 2020.

[83] Cornelia Ilie. Question-response argumentation in talk shows. *Journal of Pragmatics*, 31(8):975–999, 1999.

[84] Shankar Iyer, Nikhil Dandekar, and Kornèl Csernai. First quora dataset release: Question pairs. Retrieved at `https://quoradata.quora.com/First-Quora-Dataset-Release-Question-Pairs`.

[85] Nitin Jindal and Bing Liu. Identifying comparative sentences in text documents. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2006*, pages 244–251. ACM, 2006.

[86] Nitin Jindal and Bing Liu. Mining comparative sentences and relations. In *Proceedings of the The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference*, pages 1331–1336. AAAI Press, 2006.

[87] Karen Spärck Jones. A statistical interpretation of term specificity and its application in retrieval. *J. Documentation*, 60(5):493–502, 2004.

[88] Marcin Kaszkiel and Justin Zobel. Passage retrieval revisited. In *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 1997*, pages 178–185. ACM, 1997.

[89] Makoto P. Kato, Ryen W. White, Jaime Teevan, and Susan T. Dumais. Clarifications and question specificity in synchronous social Q&A. In *Proceedings of the 2013 ACM SIGCHI Conference on Human Factors in Computing Systems, CHI 2013*, pages 913–918. ACM, 2013.

[90] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient

gradient boosting decision tree. In *Proceedings of the Annual Conference on Neural Information Processing Systems, NIPS 2017*, pages 3146–3154, 2017.

[91] Wiltrud Kessler and Jonas Kuhn. A corpus of comparisons in product reviews. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014*, pages 2242–2248. European Language Resources Association (ELRA), 2014.

[92] Wiltrud Kessler and Jonas Kuhn. Detecting comparative sentiment expressions – A case study in annotation design decisions. In *Proceedings of the 12th Edition of the Konvens Conference, KONVENS 2014*, pages 165–170. Universitätsbibliothek Hildesheim, 2014.

[93] Omar Khattab and Matei Zaharia. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2020*, pages 39–48. ACM, 2020.

[94] Johannes Kiesel, Arefeh Bahrami, Benno Stein, Avishek Anand, and Matthias Hagen. Toward voice query clarification. In *Proceedings of the 41st International ACM Conference on Research and Development in Information Retrieval, SIGIR 2018*, pages 1257–1260. ACM, 2018.

[95] Johannes Kiesel, Arefeh Bahrami, Benno Stein, Avishek Anand, and Matthias Hagen. Clarifying false memories in voice-based search. In *Proceedings of the 2019 Conference on Human Information Interaction & Retrieval, CHIIR 2019*, pages 331–335. ACM, 2019.

[96] Johannes Kiesel, Xiaoni Cai, Roxanne El Baff, Benno Stein, and Matthias Hagen. Toward conversational query reformulation. In *Proceedings of the 2nd International Conference on Design of Experimental Search & Information Retrieval Systems, DESIRES 2021*, pages 91–101. CEUR-WS.org, 2021.

[97] Yoon Kim. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014*, pages 1746–1751. Association for Computational Linguistic, 2014.

[98] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015*. ICLR, 2015

[99] Oleksandr Kolomiyets and Marie-Francine Moens. A survey on question answering technology from an information retrieval perspective. *Information Sciences*, 181(24):5412–5434, 2011.

[100] Weize Kong, Swaraj Khadanga, Cheng Li, Shaleen Kumar Gupta, Mingyang Zhang, Wensong Xu, and Michael Bendersky. Multi-aspect dense retrieval. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2022*, pages 3178–3186. ACM, 2022.

[101] Antonios Minas Krasakis, Mohammad Aliannejadi, Nikos Voskarides, and Evangelos Kanoulas. Analysing the effect of clarifying questions on document ranking in conversational search. In *Proceedings of the 2020 ACM SIGIR International Conference on the Theory of Information Retrieval, ICTIR 2020*, pages 129–132. ACM, 2020.

[102] Klaus Krippendorff. *Content analysis: An introduction to its methodology*. Sage Publications, 2018.

[103] Ashish Kulkarni, Narasimha Raju Uppalapati, Pankaj Singh, and Ganesh Ramakrishnan. An interactive multi-label consensus labeling model for multiple labeler judgments. 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)*, pages 1479–1486. AAAI Press, 2018.

[104] Vaibhav Kumar and Alan W. Black. ClarQ: A large-scale and diverse dataset for clarification question generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 7296–7301. Association for Computational Linguistics, 2020.

[105] Vaibhav Kumar, Vikas Raunak, and Jamie Callan. Ranking clarification questions via natural language inference. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management, CIKM 2020*, pages 2093–2096. ACM, 2020.

[106] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Trans. Assoc. Comput. Linguistics*, 7: 452–466, 2019.

[107] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A lite BERT for self-supervised learning of language representations. In *Proceedings of the 8th International Conference on Learning Representations, ICLR 2020*. OpenReview.net, 2020.

[108] Thomas W. Lauer and Eileen Peacock. An analysis of comparison questions in the context of auditing. *Discourse Processes*, 13(3):349–361, 1990.

[109] Trung-Hoang Le and Hady W. Lauw. Explainable recommendation with comparative constraints on product aspects. In *Proceedings of the Fourteenth ACM International Conference on Web Search and Data Mining, WSDM 2021*, pages 967–975. ACM, 2021.

[110] Wendy Grace Lehnert. The process of question answering. Technical report, Yale University, 1977.

[111] Ran Levy, Ben Bogin, Shai Gretz, Ranit Aharonov, and Noam Slonim. Towards an argumentative content search engine using weak supervision. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018*, pages 2066–2081. Association for Computational Linguistics, 2018.

[112] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 7871–7880. Association for Computational Linguistics, 2020.

[113] Shasha Li, Chin-Yew Lin, Young-In Song, and Zhoujun Li. Comparable entity mining from comparative questions. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL 2010*, pages 650–658. The Association for Computer Linguistics, 2010.

[114] Xiao Li, Ye-Yi Wang, and Alex Acero. Learning query intent from regularized click graphs. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR 2008*, pages 339–346. ACM, 2008.

[115] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. Pyserini: A python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2021*, pages 2356–2362. ACM, 2021.

[116] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. Distilling dense representations for ranking using tightly-coupled teachers. *CoRR*, abs/2010.11386, 2020.

[117] Jing Liu, Xiaoying Wang, and Lihua Huang. Fusing various document representations for comparative text identification from product reviews. In *Web Information Systems and Applications - 18th International Conference, WISA 2021*, pages 531–543. Springer, 2021.

[118] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019.

[119] Hao Ma, Michael R. Lyu, and Irwin King. Diversifying query suggestion results. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010*. AAAI Press, 2010.

[120] Nianzu Ma, Sahisnu Mazumder, Hao Wang, and Bing Liu. Entity-aware dependency-based deep graph attention network for comparative preference classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 5782–5788. Association for Computational Linguistics, 2020.

[121] Sean MacAvaney. OpenNIR: A complete neural ad-hoc ranking pipeline. In *Proceedings of the 13th ACM International Conference on Web Search and Data Mining, WSDM 2020*, pages 845–848. ACM, 2020.

[122] Sean MacAvaney, Andrew Yates, Sergey Feldman, Doug Downey, Arman Cohan, and Nazli Goharian. Simplified data wrangling with `ir_datasets`. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2021*, pages 2429–2436. ACM, 2021.

[123] Donald Metzler and W. Bruce Croft. Analysis of statistical question classification for fact-based questions. *Inf. Retr.*, 8(3):481–504, 2005.

[124] Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In Yoshua Bengio and Yann LeCun, editors, *Proceedings of the 1st International Conference on Learning Representations, ICLR 2013, Workshop Track Proceedings*, 2013.

[125] George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, November 1995. ISSN 0001-0782.

[126] George A Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J Miller. Introduction to WordNet: An on-line lexical database. *International journal of lexicography*, 3(4):235–244, 1990.

[127] Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. Deep learning-based text classification: A comprehensive review. *ACM Comput. Surv.*, 54(3): 62:1–62:40, 2021.

[128] Junta Mizuno, Tomoyosi Akiba, Atsushi Fujii, and Katunobu Itou. Non-factoid question answering experiments at ntcir-6: Towards answer type detection for realworld questions. In *Proceedings of the 6th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access, NTCIR-6*, pages 487–492. NII, 2007.

[129] Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiao-Dan Zhu, and Colin Cherry. SemEval-2016 Task 6: Detecting stance in wweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016*, pages 31–41. The Association for Computer Linguistics, 2016.

[130] Dan Moldovan, Marius Paşca, Sanda Harabagiu, and Mihai Surdeanu. Performance issues and error analysis in an open-domain question answering system. *ACM Transactions on Information Systems (TOIS)*, 21(2):133–154, 2003.

[131] Friederike Moltmann. *Coordination and Comparatives*. PhD thesis, Massachusetts Institute of Technology, 1992.

[132] Akiyo Nadamoto and Katsumi Tanaka. A comparative web browser (CWB) for browsing and comparing web pages. In *Proceedings of the 12th International World Wide Web Conference, WWW 2003*, pages 727–735. ACM, 2003.

[133] Ben R. Newell, David A. Lagnado, and David R. Shanks. *Straight choices: The psychology of decision making*. Psychology Press, 2015.

[134] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems, NIPS 2016*. CEUR-WS.org, 2016.

[135] Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. From doc2query to docttttquery. *Online preprint*, 2019.

[136] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford University InfoLab, 1999.

[137] Alexander Panchenko, Alexander Bondarenko, Mirco Franzek, Matthias Hagen, and Chris Biemann. Categorizing comparative sentences. In *Proceedings of the 6th Workshop on Argument Mining (ArgMining 2019) at ACL*, pages 136–145. Association for Computational Linguistics, 2019.

[138] Bo Pang and Ravi Kumar. Search in the lost sense of query: Question formulation in web search queries and its temporal changes. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 135–140. Association for Computational Linguistics, 2011.

[139] Dae Hoon Park and Catherine Blake. Identifying comparative claim sentences in full-text scientific articles. In *Proceedings of the Workshop on Detecting Structure in Scholarly Discourse at ACL 2012*, pages 1–9. Association for Computational Linguistics, 2012.

[140] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014*, pages 1532–1543. ACL, 2014.

[141] Jeffrey Pomerantz. A linguistic analysis of question taxonomies. *J. Assoc. Inf. Sci. Technol.*, 56(7):715–728, 2005.

[142] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[143] Martin Porter. Snowball. `http://snowball.tartarus.org/`, 2001.

[144] Martin Potthast, Lukas Gienapp, Florian Euchner, Nick Heilenkötter, Nico Weidmann, Henning Wachsmuth, Benno Stein, and Matthias Hagen. Argument search: Assessing argument relevance. In *Proceedings of the 42nd International ACM Conference on Research and Development in Information Retrieval, SIGIR 2019*, pages 1117–1120. ACM, 2019.

[145] Martin Potthast, Matthias Hagen, and Benno Stein. The dilemma of the direct answer. *SIGIR Forum*, 54(1), June 2020. ISSN 0163-5840.

[146] Ronak Pradeep, Rodrigo Nogueira, and Jimmy Lin. The expando-mono-duo design pattern for text ranking with pretrained sequence-to-sequence models. *CoRR*, abs/2101.05667, 2021.

[147] John M. Prager, Dragomir R. Radev, Eric W. Brown, Anni Coden, and Valerie Samn. The use of predictive annotation for question answering in TREC8. In *Proceedings of The Eighth Text REtrieval Conference, TREC 1999*. NIST, 1999.

[148] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A Python natural language processing toolkit for many numan languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, ACL 2020*, pages 101–108. Association for Computational Linguistics, 2020.

[149] Chen Qu, Liu Yang, W. Bruce Croft, Falk Scholer, and Yongfeng Zhang. Answer interaction in non-factoid question answering systems. In *Proceedings of the 2019 Conference on Human Information Interaction and Retrieval, CHIIR 2019*, pages 249–253. ACM, 2019.

[150] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[151] Pavani Rajula, Chia-Chien Hung, and Simone Paolo Ponzetto. Stacked model based argument extraction and stance detection using embedded LSTM model. In *Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum*, pages 3064–3073. CLEF and CEUR-WS.org, 2022.

[152] Ashish Rana, Pujit Golchha, Roni Juntunen, Andreea Coajă, Ahmed Elzamarany, Chia-Chien Hung, and Simone Paolo Ponzetto. LeviRank: Limited query expansion with voting integration for document retrieval and ranking. In *Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum*, pages 3074–3089. CLEF and CEUR-WS.org, 2022.

[153] Jan Heinrich Reimer, Johannes Huck, and Alexander Bondarenko. Grimjack at Touché 2022: Axiomatic re-ranking and query reformulation. In *Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum*, pages 3090–3104. CLEF and CEUR-WS.org, 2022.

[154] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *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*, pages 3980–3990. Association for Computational Linguistics, 2019.

[155] Ruty Rinott, Lena Dankin, Carlos Alzate Perez, Mitesh M. Khapra, Ehud Aharoni, and Noam Slonim. Show me your evidence - an automatic method for context dependent evidence detection. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015*, pages 440–450. Association for Computational Linguistics, 2015.

[156] Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. Okapi at TREC-3. In *Proceedings of The Third Text REtrieval Conference, TREC 1994*, pages 109–126. (NIST), 1994.

[157] Stephen E. Robertson, Hugo Zaragoza, and Michael J. Taylor. Simple BM25 extension to multiple weighted fields. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management, CIKM 2004*, pages 42–49. ACM, 2004.

[158] Joseph Rocchio. Relevance feedback in information retrieval. *The Smart retrieval system-experiments in automatic document processing*, pages 313–323, 1971.

[159] Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A Primer in BERTology: What we know about how BERT works. *Trans. Assoc. Comput. Linguistics*, 8:842–866, 2020.

[160] Philipp Rösner, Niclas Arnhold, and Tobias Xylander. Quality-aware argument re-ranking for comparative questions. In *Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum*, pages 3105–3114. CLEF and CEUR-WS.org, 2022.

[161] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS 2017*, pages 3856–3866. 2017.

[162] Paul A. Samuelson and William D. Nordhaus. *Economics, 19th Edition*. McGraw-Hill/Irwin, 2009.

[163] Olivia Sanchez-Graillet, Christian Witte, Frank Grimm, Steffen Grautoff, Basil Ell, and Philipp Cimiano. Synthesizing evidence from clinical trials with dynamic interactive argument trees. *J. Biomed. Semant.*, 13(1):16, 2022.

[164] Mark Sanderson and W. Bruce Croft. The history of information retrieval research. *Proc. IEEE*, 100(Centennial-Issue):1444–1451, 2012.

[165] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019.

[166] Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M. Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Févry, Jason Alan Fries, Ryan Teehan, Stella Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. Multitask prompted training enables zero-shot task generalization. In *Proceedings of the Tenth International Conference on Learning Representations, ICLR 2022*, OpenReview.net, 2022.

[167] Beatrice Santorini. Part-of-speech tagging guidelines for the penn treebank project (3rd revision). *Technical Reports (CIS)*, 1990.

[168] Rodrygo L. T. Santos, Craig MacDonald, and Iadh Ounis. Search Result Diversification. *Found. Trends Inf. Retr.*, 9(1):1–90, 2015.

[169] Matthias Schildwächter, Alexander Bondarenko, Julian Zenker, Matthias Hagen, Chris Biemann, and Alexander Panchenko. Answering comparative questions: Better than ten-blue-links? In *Proceedings of the 2019 Conference on Human Information Interaction and Retrieval, CHIIR 2019*, pages 361–365. ACM, 2019.

[170] Ilya Segalovich. A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine. In *Proceedings of the International Conference on Machine Learning; Models, Technologies and Applications, MLMTA 2003*, pages 273–280. CSREA Press, 2003.

[171] Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. Towards facet-driven generation of clarifying questions for conversational search. In *Proceedings of the 2021 ACM SIGIR International Conference on the Theory of Information Retrieval, ICTIR 2021*, pages 167–175. ACM, 2021.

[172] Priyanka Sen, Alham Fikri Aji, and Amir Saffari. Mintaka: A complex, natural, and multilingual dataset for end-to-end question answering. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022*, pages 1604–1619. International Committee on Computational Linguistics, 2022.

[173] Mahsa S. Shahshahani and Jaap Kamps. University of Amsterdam at CLEF 2020. In *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum*, CEUR Workshop Proceedings. CLEF and CEUR-WS.org, 2020.

[174] Dou Shen, Jian-Tao Sun, Qiang Yang, and Zheng Chen. Building bridges for web query classification. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR 2006*, pages 131–138. ACM, 2006.

[175] Yangyang Shi, Kaisheng Yao, Le Tian, and Daxin Jiang. Deep LSTM based feature mapping for query classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016*, pages 1501–1511. Association for Computational Linguistics, 2016.

[176] Ekaterina Shirshakova and Ahmad Wattar. Thor at Touché 2021: Argument Retrieval for Comparative Questions. In *Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum*, CEUR Workshop Proceedings, pages 2455–2462. CLEF and CEUR-WS.org, 2021.

[177] Bjarne Sievers. Question answering for comparative questions with GPT-2. In *Working Notes of CLEF 2020 - Conference and Labs of the*

Evaluation Forum, CEUR Workshop Proceedings. CLEF and CEUR-WS.org, 2020.

[178] R. F. Simmons. Natural language question-answering systems. *Commun. ACM*, 13(1):15–30, 1970.

[179] Herbert A. Simon. *The new science of management decision*. Harper & Brothers, 1960.

[180] Amit Singhal, Steven P. Abney, Michiel Bacchiani, Michael Collins, Donald Hindle, and Fernando C. N. Pereira. AT&T at TREC-8. In *Proceedings of The Eighth Text REtrieval Conference, TREC 1999*. NIST, 1999.

[181] Parinaz Sobhani, Diana Inkpen, and Xiaodan Zhu. A dataset for multi-target stance detection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017*, pages 551–557. Association for Computational Linguistics, 2017.

[182] Swapna Somasundaran and Janyce Wiebe. Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 116–124. Association for Computational Linguistics, 2010.

[183] Swapna Somasundaran, Theresa Wilson, Janyce Wiebe, and Veselin Stoyanov. QA with attitude: Exploiting opinion type analysis for improving question answering in on-line discussions and the news. In *Proceedings of the First International Conference on Weblogs and Social Media, ICWSM 2007*.

[184] Michael Spenke, Christian Beilken, and Thomas Berlage. FOCUS: The interactive table for product comparison and selection. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology, UIST 1996*, pages 41–50. ACM, 1996.

[185] Steffen Staab and Udo Hahn. Comparatives in context. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Innovative Applications of Artificial Intelligence Conference, AAAI 97*, pages 616–621. AAAI Press / The MIT Press, 1997.

[186] Christian Stab, Johannes Daxenberger, Chris Stahlhut, Tristan Miller, Benjamin Schiller, Christopher Tauchmann, Steffen Eger, and Iryna Gurevych. ArgumenText: Searching for arguments in heterogeneous sources. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics, NAACL-HLT 2018*, pages 21–25. Association for Computational Linguistics, 2018.

[187] Christian Stab, Tristan Miller, Benjamin Schiller, Pranav Rai, and Iryna Gurevych. Cross-topic argument mining from heterogeneous sources. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, pages 3664–3674. Association for Computational Linguistics, 2018.

[188] Leon Stassen. The comparative compared. *Journal of Semantics*, 3 (1-2):143–182, 1984.

[189] Arnim von Stechow. Comparing semantic theories of comparison. *Journal of semantics*, 3(1-2):1–77, 1984.

[190] Manfred Stede and Jodi Schneider. *Argumentation mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2018.

[191] Jian-Tao Sun, Xuanhui Wang, Dou Shen, Hua-Jun Zeng, and Zheng Chen. CWS: A comparative web search system. In *Proceedings of the 15th international conference on World Wide Web, WWW 2006*, pages 467–476. ACM, 2006.

[192] Leila Tavakoli, Hamed Zamani, Falk Scholer, William Bruce Croft, and Mark Sanderson. Analyzing clarification in asynchronous information-seeking conversations. *Journal of the Association for Information Science and Technology*, 2021.

[193] Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021*, 2021.

[194] Maksim Tkachenko and Hady Wirawan Lauw. Generative modeling of entity comparisons in text. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management, CIKM 2014*, pages 859–868. ACM, 2014.

[195] Orith Toledo-Ronen, Matan Orbach, Yonatan Bilu, Artem Spector, and Noam Slonim. Multilingual argument mining: Datasets and analysis. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020*, pages 303–317. Association for Computational Linguistics, 2020.

[196] Erica Turner and Lee Rainie. Most Americans rely on their own research to make big decisions, and that often means online searches, 2020. Retrieved at `https://pewrsr.ch/2VO7bQn`.

[197] Amos Tversky. Features of similarity. *Psychological Review*, 84(4): 327–352, 1977.

[198] Michael Völske, Pavel Braslavski, Matthias Hagen, Galina Lezina, and Benno Stein. What users ask a search engine: Analyzing one billion Russian question queries. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015*, pages 1571–1580. ACM, 2015.

[199] Henning Wachsmuth, Johannes Kiesel, and Benno Stein. Sentiment flow–A general model of web review argumentation. *Proceedings of the 20th Conference on Empirical Methods in Natural Language Processing, EMNLP 2015*, pages 601–611. Association for Computational Linguistics, 2015.

[200] Henning Wachsmuth, Nona Naderi, Ivan Habernal, Yufang Hou, Graeme Hirst, Iryna Gurevych, and Benno Stein. Argumentation quality assessment: Theory vs. practice. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017*, pages 250–255. Association for Computational Linguistics, 2017.

[201] Henning Wachsmuth, Nona Naderi, Yufang Hou, Yonatan Bilu, Vinodkumar Prabhakaran, Tim Alberdingk Thijm, Graeme Hirst, and Benno Stein. Computational argumentation quality assessment in natural language. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017*, pages 176–187. Association for Computational Linguistics, 2017.

[202] Henning Wachsmuth, Martin Potthast, Khalid Al-Khatib, Yamen Ajjour, Jana Puschmann, Jiani Qu, Jonas Dorsch, Viorel Morari, Janek Bevendorff, and Benno Stein. Building an argument search engine for the Web. In *Proceedings of the 4th Workshop on Argument Mining, ArgMining@EMNLP 2017*, pages 49–59. Association for Computational Linguistics, 2017.

[203] Henning Wachsmuth, Benno Stein, and Yamen Ajjour. "PageRank" for argument relevance. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017*, pages 1116–1126. Association for Computational Linguistics, 2017.

[204] Wei Wang, TieJun Zhao, GuoDong Xin, and YongDong Xu. Exploiting machine learning for comparative sentences extraction. *International Journal of Hybrid Information Technology*, 8(3):347–354, 2015.

[205] Ingmar Weber, Antti Ukkonen, and Aris Gionis. Answers, not links: Extracting tips from Yahoo! Answers to address how-to web queries. In *Proceedings of the Fifth International Conference on Web Search and Web Data Mining, WSDM 2012*, pages 613–622. ACM, 2012.

[206] Jan-Niklas Weder and Thi Kim Hanh Luu. Argument retrieval for comparative questions based on independent features. In *Work-*

ing Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum, CLEF 2021, pages 2403–2416. CEUR Workshop Proceedings. CEUR-WS.org, 2021.

[207] Ryen W. White, Matthew Richardson, and Wen-tau Yih. Questions vs. queries in informational search tasks. In *Proceedings of the 24th International Conference on World Wide Web Companion, WWW 2015*, pages 135–136. ACM, 2015.

[208] Adina Williams, Nikita Nangia, and Samuel R. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018*, pages 1112–1122. Association for Computational Linguistics, 2018.

[209] Zhibiao Wu and Martha Stone Palmer. Verb semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, ACL, 1994*, pages 133–138. Morgan Kaufmann Publishers / ACL, 1994.

[210] Liqiang Xiao, Honglun Zhang, Wenqing chen, Yongkun Wang, and Yaohui Jin. MCapsNet: Capsule network for text with multi-task learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, pages 4565–4574. Association for Computational Linguistics, 2018.

[211] Kaiquan Xu, Stephen Shaoyi Liao, Jiexun Li, and Yuxia Song. Mining comparative opinions from customer reviews for competitive intelligence. *Decision Support Systems*, 50(4):743–754, 2011.

[212] Bishan Yang, Jian-Tao Sun, Tengjiao Wang, and Zheng Chen. Effective multi-label active learning for text classification. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2009*, pages 917–926. ACM, 2009.

[213] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, pages 2369–2380. Association for Computational Linguistics, 2018.

[214] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. XLNet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019*, pages 5754–5764, 2019.

[215] Bengong Yu, Qingtang Xu, and Peihang Zhang. Question classification based on MAC-LSTM. In *Proceedings of the Third IEEE International Conference on Data Science in Cyberspace, DSC 2018*, pages 69–75. IEEE, 2018.

[216] Hong Yu and Vasileios Hatzivassiloglou. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing, EMNLP 2003*, pages 129–136. Association for Computational Linguistics, 2003.

[217] Evi Yulianti, Ruey-Cheng Chen, Falk Scholer, W. Bruce Croft, and Mark Sanderson. Document summarization for answering non-factoid queries. *IEEE Trans. Knowl. Data Eng.*, 30(1):15–28, 2018.

[218] Hamed Zamani, Susan T. Dumais, Nick Craswell, Paul N. Bennett, and Gord Lueck. Generating clarifying questions for information retrieval. In *Proceedings of the Web Conference, WWW 2020*, pages 418–428. ACM / IW3C2, 2020.

[219] Hamed Zamani, Gord Lueck, Everest Chen, Rodolfo Quispe, Flint Luu, and Nick Craswell. MIMICS: A Large-scale data collection for search clarification. In *Proceedings of the 29th ACM International Con-*

ference on Information and Knowledge Management, CIKM 2020, pages 3189–3196. ACM, 2020.

[220] Hamed Zamani, Bhaskar Mitra, Everest Chen, Gord Lueck, Fernando Diaz, Paul N. Bennett, Nick Craswell, and Susan T. Dumais. Analyzing and learning from user interactions for search clarification. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2020*, pages 1181–1190. ACM, 2020.

[221] Natalia Zevakhina and Svetlana Dzhakupova. Russian metalinguistic comparatives: A functional perspective. HSE Working papers WP BRP 39/LNG/2015, National Research University Higher School of Economics, 2015.

[222] Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *ACM SIGIR Forum*, volume 51, pages 268–276. ACM, 2017.

[223] Aston Zhang, Lluis Garcia Pueyo, James Bradley Wendt, Marc Najork, and Andrei Z. Broder. Email category prediction. In *Proceedings of the 26th International Conference on World Wide Web Companion, WWW 2017*, pages 495–503. ACM, 2017.

[224] Dell Zhang and Wee Sun Lee. Question classification using support vector machines. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2003*, pages 26–32. ACM, 2003.

[225] Zhiling Zhang and Kenny Q. Zhu. Diverse and specific clarification question generation with keywords. In *Proceedings of the Web Conference, WWW 2021*, pages 3501–3511. ACM / IW3C2, 2021.