

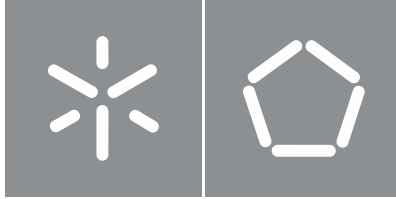
University of Minho
School of Engineering

**Extracting Protein-Protein Interactions from
Biomedical Literature with Large Language Models**

Mariana Pereira Oliveira

Mariana Pereira Oliveira

**Extracting Protein-Protein Interactions from
Biomedical Literature with Large Language
Models**



University of Minho

School of Engineering

Mariana Pereira Oliveira

Extracting Protein-Protein Interactions from Biomedical Literature with Large Language Models

Master's Dissertation

Master's Degree in Bioinformatics

Dissertation supervised by:

**Miguel Francisco Almeida Pereira Rocha, Associate
Professor, Department of Informatics, University of
Minho**

**Yvan Saeys, Associate Professor, Department of
Mathematics, Computer Science and Statistics,
Ghent University**

Copyright and Terms of Use for Third Party Work

This dissertation reports on academic work that can be used by third parties as long as the internationally accepted standards and good practices are respected concerning copyright and related rights.

This work can thereafter be used under the terms established in the license below.

Readers needing authorization conditions not provided for in the indicated licensing should contact the author through the RepositóriUM of the University of Minho.

License granted to users of this work:



CC BY

<https://creativecommons.org/licenses/by/4.0/>

Acknowledgements

Firstly, I would like to thank my supervisor, Miguel Rocha, and my co-supervisor, Yvan Saeys, for helping and allowing me to develop this thesis in a new place and while always offering support and help whenever I needed.

This would not have been the same without Johnathan along the way, your support and insights kept me determined to achieve this, thank you!

I would like to thank my family - my mum, my dad and my brother - for being big supporters of my adventures and for making sure I kept my head held high during hard times.

To my Braga friends, who have been with me along this academic journey. Through all the bullying and all the fun, through sadness and joy, thank you all! I would like to specially thank Sofia and Dudu. Two crackheads that have kept me motivated and alive.

To my hometown friends, who remind me softly of what it means to return to a place of love and care. Especially to Marques, you've been the best.

And of course, my Erasmus friends, without you this journey would have not been as fun and as interesting as it was. Thank you to my girls, Bea, Luísa, Carol and Sofia, you've meant so much to me in such a short time.

Statement of Integrity

I hereby declare having conducted this academic work with integrity.

I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

University of Minho, Braga, november 2025

Mariana Pereira Oliveira

Abstract

Protein–protein interactions (PPIs) are essential to comprehend cellular processes, yet their identification from biomedical literature remains challenging. Large Language Models (LLMs) offer a promising ability for automating relation extraction, but their reliability in specialized biomedical applications is not fully established. This thesis evaluated the potential of open-source LLMs for PPI extraction using a Retrieval-Augmented Generation (RAG) pipeline.

Full text articles from biomedical literature were retrieved, processed and PPIs extracted. These interactions were then benchmarked against curated databases such as BioGRID, STRING, and GENIA. BioMistral consistently produced higher-quality outputs during prompt testing compared to other models, which showed more variable performance. However, benchmarking revealed high false-positive rates and generally low precision.

The findings highlight both the opportunities and limitations of LLMs for biomedical relation extraction. Domain-specific models outperformed general-purpose LLMs during prompt testing, emphasizing the importance of specialized training. Persistent challenges include accurate entity recognition, handling full-text complexity, and the absence of robust benchmarks beyond abstracts.

This work shows that while open-source LLMs are a valuable starting point for automated PPI extraction, their successful application requires careful prompt engineering, domain adaptation, and improved evaluation strategies.

Keywords Large Language Models, Protein-Protein Interactions, Retrieval Augmented Generation, Natural Language Processing

Resumo

As interações proteína-proteína (PPIs) são essenciais para compreender os processos celulares, mas a sua identificação na literatura biomédica continua a ser um desafio. Os Modelos de Linguagem de Grande Escala (LLMs) oferecem uma capacidade promissora para automatizar a extração de relações, mas a sua fiabilidade em aplicações biomédicas especializadas ainda não está totalmente estabelecida. Esta tese avaliou o potencial dos LLMs de código aberto para a extração de PPIs utilizando um pipeline de Geração Aumentada por Recuperação (RAG).

Artigos completos da literatura biomédica foram recuperados, processados e as PPIs extraídas. Estas interações foram então comparadas com bases de dados selecionadas, como BioGRID, STRING e GENIA. O BioMistral produziu consistentemente resultados de maior qualidade durante os testes rápidos em comparação com outros modelos, que mostraram um desempenho mais variável. No entanto, a comparação revelou altas taxas de falsos positivos e precisão geralmente baixa.

As conclusões destacam tanto as oportunidades como as limitações dos LLMs para a extração de relações biomédicas. Os modelos específicos do domínio superaram os LLMs de uso geral durante a fase de testes de prompts, enfatizando a importância do treinamento especializado. Os desafios persistentes incluem o reconhecimento preciso de entidades, o tratamento da complexidade do texto completo e a ausência de referências robustas além dos resumos.

Este trabalho mostra que, embora os LLMs de código aberto sejam um ponto de partida valioso para a extração automatizada de PPI, sua aplicação bem-sucedida requer engenharia cuidadosa, adaptação do domínio e estratégias de avaliação aprimoradas.

Palavras-chave Modelos de Linguagem de Grande Escala, Interações Proteína-Proteína, Geração Aumentada de Recuperação, Processamento de Linguagem Natural

Contents

1	Introduction	1
1.1	Context and motivation	1
1.2	Objectives	2
1.3	Thesis organization	3
2	State of the Art	4
2.1	Natural Language Processing in Biomedical Research	4
2.1.1	NLP Pipeline	6
2.1.2	Large Language Models	7
2.1.3	Prompt Engineering	16
2.2	Mechanism and Insights of Cell-Cell Communication	20
2.2.1	Types of CCC	20
2.2.2	Importance of studying CCC	24
2.2.3	How to predict CCC	24
2.2.4	Computational Approaches to Study CCC	26
2.2.5	Computational minimization of false discoveries	29
2.2.6	Validation of CCC predictions	30
2.2.7	Challenges	30
2.2.8	Future Insights	31
3	Materials and Methods	33
3.1	Benchmark Datasets	33
3.1.1	BioGRID	33
3.1.2	STRING	35
3.1.3	GENIA Corpus	35
3.2	RAG Pipeline	36

3.2.1	Extraction and Preprocessing of Documents	36
3.2.2	Embedding Generation	37
3.2.3	Retrieval and Generation with LLMs	38
4	Results and Discussion	42
4.1	Prompt Testing	42
4.1.1	Protein A Outputs	42
4.1.2	Interaction Type Output	44
4.1.3	Protein B Output	45
4.1.4	Overall performance of LLMs	46
4.2	PPI Extraction	48
4.2.1	BioGRID	49
4.2.2	STRING	50
4.2.3	GENIA	50
4.2.4	Overall performance of LLMs	51
5	Conclusion	53
I	Appendices	68
A	BioGRID Dataset Details	69
B	STRING Dataset Details	71
C	Prompts Selected	73

List of Figures

2.1	NLP Applications	5
2.2	Main steps of a NLP pipeline.	7
2.3	RAG paradigms: Naive, Advanced and Modular.	18
2.4	Ligand-receptor complex formation from left to right: primary recognition, orientation and physical binding. Adapted from (Guryanov et al., 2016).	21
2.5	Exemplification of the different receptors that exist.	23
3.1	Schema of the RAG pipeline applied.	37
3.2	Schema of the RAG pipeline applied for prompt testing.	38
3.3	Schematic representation of how the 10 selected prompts were incorporated into the RAG pipeline for PPI extraction.	40
4.1	Protein A output grouped by quality.	43
4.2	Quality of Protein A output per LLM.	44
4.3	Interaction type output grouped by quality.	44
4.4	Quality of Interaction Type output per LLM.	45
4.5	Protein B output grouped by quality.	46
4.6	Quality of Protein B output per LLM.	46
4.7	Average Quality Score per LLM and Output Type.	47
4.8	LLM Performance Comparison Across Tasks.	48

List of Tables

2.1	Comparison of Open-source LLMs.	12
3.1	Summary of benchmark datasets used in this study.	34
3.2	GENIA final dataset columns and example entries.	36
3.3	Evaluation criteria for extracted Protein and Interaction Type outputs.	41
4.1	Performance of LLMs on BioGRID benchmark.	49
4.2	Performance of LLMs on STRING benchmark.	50
4.3	Performance of LLMs on GENIA benchmark.	51
A.1	BioGRID dataset columns and example entries (Part 1)	69
A.2	BioGRID dataset columns and example entries (Part 2)	70
B.1	STRING dataset columns and example entries (Part 1)	71
B.2	STRING dataset columns and example entries (Part 2)	72
C.1	Prompts selected for evaluation. G = Good, N = Neutral, B = Bad.	73
C.2	Prompts selected for evaluation. G = Good, N = Neutral, B = Bad.	74

Chapter 1

Introduction

1.1 Context and motivation

The transition from single-cell organisms to multi-cell organisms expanded the potential for evolution into new life forms and made them dependent on the establishment of a dynamic communication network between the cells within (Su et al., 2024; Cheng et al., 2023). These cells synthesize a diverse array of molecules and membrane structures, including structural and functional proteins, as well as membrane projections. These components activate signaling pathways in adjacent cells, orchestrating various cellular functions crucial for regulating cellular activities and intercellular interactions, playing an essential role in homeostasis, development, and disease processes. (Armingol et al., 2024; Browaeys et al., 2020; Almet et al., 2021; Su et al., 2024). This process is known as Cell-Cell Communication (CCC) (Almet et al., 2021).

Cells communicate by recognizing, receiving and responding to external signal molecules involving signal generation followed by signal transduction. The latter is the process through which biological information (excitation or inhibition) is transformed and transmitted between cells or within cells. A specific stimulus triggers the transcription and translation of certain genes, leading to the synthesis of signaling molecules, such as proteins or other small molecules that are stored in specific organelles within the cell. Signaling molecules bind to receptors on the cell surface or inside the target cell, creating receptor-ligand complexes, and receptors end up binding to specific signaling molecules. Once binding occurs, it triggers a chain reaction of cellular responses, in which the signal is transmitted and amplified within the cell (downstream signaling), culminating in altered transcription factor activity and gene expression. Understanding how these signaling pathways work has been essential for uncovering the mechanisms of CCC (Su et al., 2024; Armingol et al., 2021).

As mentioned before, the way cells communicate plays a crucial role in various biological processes and if done incorrectly, the development of tissues, organs and immune responses can be affected (Shao

[et al., 2020](#)). For example, the failure to interact or interpret molecular information can typically lead to diseases. Therefore, understanding the complex biological pathways present in CCC is critical to provide deeper insights into the underlying mechanisms of various biological processes, helping elucidate the mechanisms behind the initiation and progression of diseases and aiding in the development of pharmaceuticals ([Azam et al., 2024](#); [Su et al., 2024](#)).

CCC research has been possible through data acquisition using multiple experimental techniques, including imaging methods (such as electron microscopy, optical microscopy, and fluorescence resonance energy transfer), cell surface detection, chemical tagging of interacting cells, and mechanical force analysis([Su et al., 2024](#)). Data generated from these experimental approaches are often integrated into biological knowledge bases, which can then be incorporated into computational tools to infer CCC. These tools - such as NicheNet, CellPhoneDB, and CellChat - combine existing data with newly generated datasets and specific analytical methods to predict intercellular communication events ([Browaeys et al., 2020](#); [Efremova et al., 2020](#); [Jin et al., 2021](#)).

Many of the knowledge resources supporting CCC inference are general biological databases, such as the Kyoto Encyclopedia of Genes and Genomes (KEGG), and they rely primarily on manually curated information from biomedical literature. As the scientific literature keeps being expanded with three thousand new publications a day, keeping with manual curation has become an enormous challenge due to its laborious and time-consuming nature. Consequently, a substantial portion of known molecular interactions described in scientific literature remains unincorporated and undiscovered. To address this gap, automated text mining techniques have emerged as an efficient means to extract valuable biological knowledge from the growing body of biomedical literature ([Azam et al., 2024](#); [Anandaraj and Bali, 2023](#)).

Since biomedical literature comes in the form of unstructured text, curators are struggling to extract structured content from it. Machine Learning (ML) techniques such as Natural Language Processing (NLP) and text mining tools have become valuable assets for curating large biological databases ([Anandaraj and Bali, 2023](#)). NLP and text mining have been revolutionized by the onset of the transformer architecture, which has enabled the use of LLMs to generation and mining of biomedical text ([Park et al., 2022](#); [Rehana et al., 2024](#); [Azam et al., 2024](#)).

1.2 Objectives

In this thesis, we aim to explore the potential of Large Language Models (LLMs) for automating the extraction of protein-protein interactions from biomedical literature. Specifically, it aims to evaluate and com-

pare different open-source LLMs on biomedical relation extraction tasks using curated datasets, assess the quality of their outputs and examine how different prompts influence extraction performance. In addition, this work implements a retrieval-augmented generation (RAG) pipeline to explore its feasibility for improving context awareness in biomedical text mining. Together, these objectives aim to identify the current strengths and limitations of open-source LLMs for relation extraction and provide insights for future model and pipeline development.

1.3 Thesis organization

The remainder of this thesis is organized as follows. Chapter 2, “State of the Art,” provides an overview of current developments in biomedical text mining and CCC. It introduces NLP pipelines, the role of LLMs and prompt engineering, and discusses the biological mechanisms of CCC, its types, computational prediction methods, validation strategies, and existing challenges.

Chapter 3, “Materials and Methods,” provides an overview of the benchmark datasets and methods used. Publicly available databases, including BioGRID and STRING, were used to evaluate the accuracy of the protein-protein interactions (PPIs) extracted. Meanwhile, the GENIA corpus was used as an additional benchmark, containing only interactions explicitly reported in article abstracts. The chapter also details the RAG pipeline applied for text extraction, preprocessing, embedding generation, and LLM-based PPI extraction, including the process of prompt creation and selection to optimize the quality of extracted PPIs.

Chapter 4, “Results and Discussion,” presents the findings from prompt testing and PPI extraction across the benchmark datasets. It first outlines the evaluation and selection of prompts for PPI extraction, and then assesses the performance of various open-source LLMs in retrieving PPIs from BioGRID, STRING, and GENIA using the preselected prompts. The chapter also discusses general trends and challenges observed, including variability between models, differences across benchmarks, and the complexities of unstructured biomedical text.

Finally, Chapter 6, “Conclusion,” summarizes the contributions of this work, highlights its limitations, and outlines potential directions for future research.

Additionally, in the Appendices, methodological details, prompts, and supplementary results are provided.

Chapter 2

State of the Art

2.1 Natural Language Processing in Biomedical Research

Since scientific literature has been rapidly growing, the difficulty for curators and researchers to keep up with the expanding data has been increasing. This makes it difficult for curators and researchers to keep up with expanding volume of data ([Anandaraj and Bali, 2023](#)). Furthermore, manual information extraction has become difficult, challenging and time consuming ([Park et al., 2023](#); [Azam et al., 2024](#)). Therefore, the search for automated text mining strategies in scientific texts without human intervention has surged. By making use of ML models - more precisely the ones used for NLP - these strategies have been successfully employed ([Park et al., 2023](#)).

The term Natural Language Processing, also called 'computational linguistics', first emerged in the 1950s when linguistics and Artificial Intelligence (AI) intersected ([Hirschberg and Manning, 2015](#); [Nadkarni et al., 2011](#)). It covers a broad variety of methods related to a computer-based processing of spoken and written language data ([Wulff et al., 2025](#); [Hirschberg and Manning, 2015](#)). More specifically, NLP refers to a theoretically motivated range of computational techniques for analyzing and representing naturally occurring texts at one or more levels of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications ([Liddy, 2001](#)). The goal is for computers to understand and generate language in a similar manner to human comprehension, mainly driven by advances in ML ([Doneva et al., 2024](#); [Wulff et al., 2025](#)).

NLP can be used for many purposes some of which include: indexing and searching large texts, Information retrieval (IR), Classification of text into categories, Information extraction (IE), Automatic language translation, Automatic summarization of texts, Question-answering (QA) and Text generations/dialogues ([Chowdhary, 2020](#)) (see Figure 2.1). IR is a task with the goal of retrieving relevant information from a large collection of documents in response to a user query. Classification of text into categories happens when a text is categorized into one or more predefined categories based on its content ([Patwardhan et al.,](#)

2023). IE focuses on recognizing, tagging and extracting into a structured representation, certain key elements of information (Liddy, 2001). Automatic language translation or machine translation is an NLP task where a source of text is automatically converted in one language to a desired text in another language. When the goal is to condense a given text into a shorter and more concise version while preserving its essential information we are facing automatic text summarization. QA encompasses automatically answering questions posed in natural language to extract the relevant information from a text corpus and present it as an answer to a user's question. Finally, text generation involves producing new text automatically, usually starting from a given prompt or input. (Patwardhan et al., 2023).

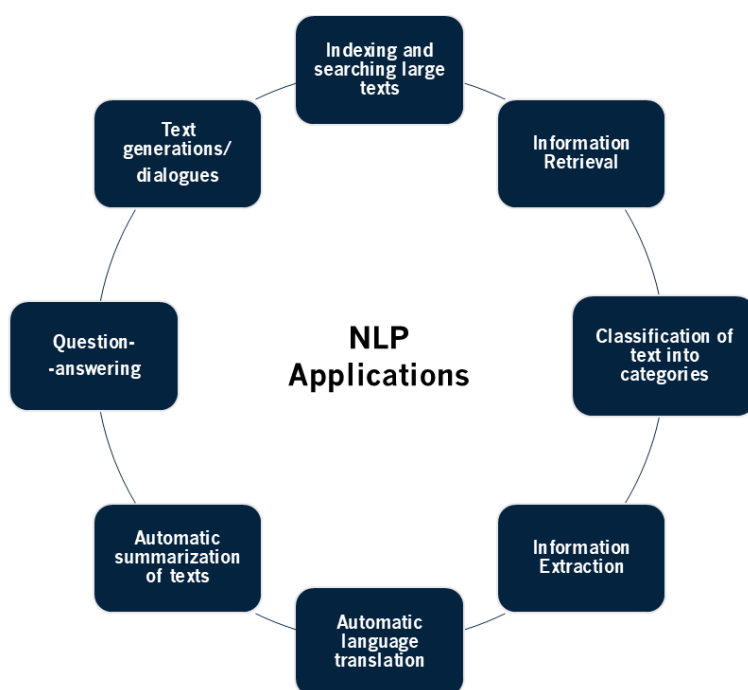


Figure 2.1: NLP Applications

The IE task possesses the ability to uncover new relationships between entities from biomedical literature, such as a drug interacting with another drug or a protein interacting with another protein, this is also known as the relation extraction task (Anandaraj and Bali, 2023). This task aims to extract a relational three tuple $\langle \text{Entity1}, \text{Relation}, \text{Entity 2} \rangle$ from natural language texts (Anandaraj and Bali, 2023; Zhang et al., 2017). It commonly uses approaches such as sentence simplification, link grammar parsing and a combination of vector and tree-based kernels to enhance PPIs relation extraction in simpler sentences. These methods can, however, lead to information loss and overfitting (Gill et al., 2024).

Some of the practical examples of the use of NLP include: multi-domain analysis, pharmacological, clinical applications and social media platforms. While in multi-domain analysis, approaches or frameworks are developed to tackle various NLP tasks using a single and cohesive strategy, in the pharmacological

domain, the identification and extraction of relationships between entities is applied. Meanwhile, the clinical applications domain serves the purpose of predicting patients outcomes and the social media platforms domain deals with, for example, mental health screening, identifying suicidal risk and predicting mental health disorder for user-generated content. Specifically regarding the field of biomedical research, NLP is applied for mostly text classification, IE and relation extraction (Doneva et al., 2024).

A major breakthrough in the field of NLP happened in 2017 with the introduction of the transformer architecture (Vaswani et al., 2023). This architecture, based on the self-attention mechanism, consists of an encoder that transforms raw input data into a structured format and a decoder that generates output sentences using a multi-head attention mechanism (Rehana et al., 2024). By allowing neural networks to access the significance of input elements, ML models developed advanced linguistic capabilities (Wulff et al., 2025; Rehana et al., 2024). This innovation marked the introduction of the LLMs, which later led to the development of models such as BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer) (Hadi et al., 2023). These transformer-based models are often pre-trained on extensive text corpora, such as a wide variety of internet text, allowing them to learn grammar, vocabulary, world knowledge and context, and to identify connections and trends among textual elements. After pre-training, they undergo fine-tuning on specific datasets, tailored to tasks such as text generation or conversation (Hadi et al., 2023; Rehana et al., 2024). The latter allows for an improved zero-shot performance and reduced misaligned behavior (Naveed et al., 2024).

2.1.1 NLP Pipeline

When creating an NLP system, a sequence of steps is followed to achieve text-processing. The main steps are as follows: data collection, text cleaning, initial processing, development of features, modeling, evaluation and, finally, implementation (Elov et al., 2023).

Following the steps shown in Figure 2.2, the first involves collecting data relevant to the given task. The desirable data can be available on multiple places such as in a public dataset, websites, images or even PDF files. Since most of these are rarely “clean”, the next step involves text cleaning. This step involves using some techniques such as Unicode Normalization which passes by removing unwanted characters or converting them to a machine-readable format or spelling corrections which involves replacing common mistakes with the correct word. Afterwards, the text data is preprocessed which mainly involves tokenization. This process encompasses segmenting the text into a list of tokens. Some other steps may occur in the initial processing such as lowercasing or stop word removal. Next, the goal is to represent the text in the numeric vector form, a canonical form. This involves performing feature engineering so that the

ML algorithm can understand the text attribute. Finally, modeling and evaluation are performed. In this final step, one or more LLMs are created and their performance is compared with evaluation indicators (Elov et al., 2023; GeeksforGeeks, 2025).

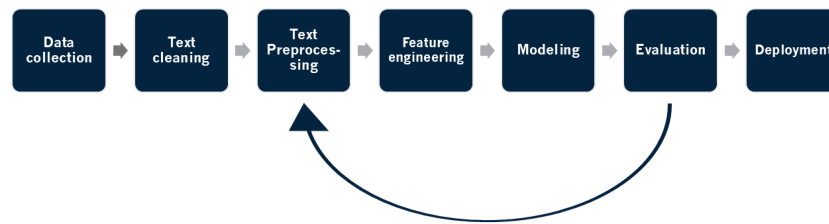


Figure 2.2: Main steps of a NLP pipeline.

2.1.2 Large Language Models

LLMs, sometimes referred to as “transformative” or “next-generation” language models, are characterized by their enormous parameter sizes and training on extensive text corpora (Park et al., 2023; Azam et al., 2024; Hadi et al., 2023). They possess an ability to learn and understand complex patterns and structures present in language data, making them able to capture semantic relationships between words and phrases (Park et al., 2023; Bolton et al., 2024). Moreover, they are also able to process large amounts of other types of data, such as visual, audio, audiovisual and multi-modal information (Hadi et al., 2023).

The history of LLMs can be traced back to the 1980s and 1990s when statistical language models (SLMs) came to address contextually relevant properties of natural language from a probabilistic statistical perspective (Rosenfeld, 2000; Stolcke, 2002) (Wang et al., 2025; Hadi et al., 2023). They employ simple probabilistic distributions to predict word sequences, such as Hidden Markov Models and Maximum Entropy Models (Wang et al., 2025; Hadi et al., 2023). Neural language models emergence followed up in the mid-2010s (Hadi et al., 2023). These models make use of neural networks - deep learning techniques - to grasp complex patterns and representations of language, predicting the subsequent words within sequences. Compared to SLMs, they introduced the ability to handle longer sequences. Afterwards, Pre-trained Language Models surfaced. They undergo initial training using extensive volume of unlabeled text and that makes them able to understand fundamental language structures such as vocabulary, syntax, semantics, and logic. They can be applied to several NLP tasks such as machine translation, text

summarization, and QA systems (Wang et al., 2025).

Finally, the transformer architecture was introduced in 2017 (Vaswani et al., 2023). The introduction of the transformer architecture enabled the models to learn the longer-term dependencies in language and allowed for parallel training on multiple Graphic Processing Units (GPUs) (Doneva et al., 2024). The key unit of this architecture is the transformer block. It consists of two main components: a multi-head self attention mechanism and a fully connected feedforward network. The first one allows the model to focus on distinct parts of the input sequence at each layer and weight the importance of each part in making a prediction and the latter allows the model to learn different attention patterns for different tasks and input sequences. Thus, being essentially a multi-layer perceptron that takes in the self-attention-generated representation as input, applies linear transformations with activation functions, and outputs the final representation (Patwardhan et al., 2023).

The training process for LLMs involves two phases - pre-training and fine-tuning. It starts with pre-training and during the process they are exposed to a wide range of datasets of internet text, allowing them to learn grammar, vocabulary, world knowledge, and context. Next, they undergo fine-tuning on specific datasets meant for specific tasks, like text generation or conversation. This step improves their capabilities for specific tasks and leads to better zero-shot results, minimizing behavior that doesn't align with user expectations (Naveed et al., 2024; Hadi et al., 2023).

LLMs show a wide range of applications throughout many fields such as: medical, education, finance, engineering, media and entertainment industry, legal practice and marketing. Concerning medical applications, ChatGPT has emerged as an interactive tool that helps learning and problem-solving and its explanations displayed high level of concordance and insightful understanding (Kung et al., 2023). When it comes to the educational field, LLMs show promise in areas such as providing student feedback, assisting teacher development, and creating personalized learning experiences (Cohn et al., 2023; Wu et al., 2023; Doveh et al., 2023). LLMs can help students complete their assignments more efficiently and provide personalized learning experiences (Kasneci et al., 2023). However, it may lead to the loss of creativity and critical skills among students as they may rely too much on AI to complete their tasks and end up not developing the necessary skills Kosmyna et al. (2025); Soubhari et al. (2023). In the finance field, risk assessment, algorithmic trading, market prediction and financial reporting are some of the use of LLMs (Zaremba and Demir, 2023; Lopez-Lira and Tang, 2024; Yang et al., 2020). When it comes to engineering, LLMs have been applied in software engineering with models being developed to assist in debugging code, for example (Fraivan and Khasawneh, 2023). The use of these models in the media and entertainment sector has led to companies introducing groundbreaking innovations that allow for one-on-one interactions

to happen ([Lachman and Joffe, 2021](#)). Moreover, they can also be used for generating attractive advertisements and marketing, political speeches, slogans and social media posts ([Eloundou et al., 2023](#); [Abdulhai et al., 2023](#)). Additionally, the future of legal practice is also changing. LLMs show a promising ability to empower lawyers to enhance legal research, drafting tasks and decision-making ([lu and Wong, 2023](#)). The adoption of LLMs in marketing through content generation, creating compelling product descriptions, ad copy, blogs, and social media posts ([Rivas and Zhao, 2023](#)).

After the rise in popularity of the transformers in machine translation, the use of this type of architecture began to gain importance for many NLP tasks. They have stayed at the top of the leaderboard rankings for most NLP tasks due to their ability to capture long-range dependencies, scalability, and versatility ([Patwardhan et al., 2023](#)).

Within the biomedical domain, the NLP task of relation extraction, a type of information extraction, has reached new benchmarks and plays a crucial role in advancing research and drug development ([Rehana et al., 2024](#)). This type of task allows for the identification of relationships between two or more identities in the literature such as drugs, genes or proteins ([Doneva et al., 2024](#)). By fine-tuning these models for specific tasks, some high-performance levels within the biomedical domain have been reached ([Doneva et al., 2024](#)). These aspects, can lead to improvements in healthcare systems by the advancements in biomedical discovery, thus lowering healthcare costs and improving patient costs ([Bolton et al., 2024](#); [Park et al., 2023](#)).

Models

Many of the available LLMs are closed-source and the underlying algorithms and models are not publicly available for scrutiny or modification. Since researchers are not able to access the code, the growth and development of the field can become limited ([Patwardhan et al., 2023](#)). Some of the non-open source LLMs include GPT-3.5, GPT-4 and Claude-2. GPT-4 has shown remarkable abilities when it comes to comprehending and generating language, conversational AI, code generation and language translation and Claude-2 has shown proficiency in context-aware responses, especially in extended interactions ([Azam et al., 2024](#); [Teebagy et al., 2023](#)). In the meantime, Claude-2 was developed by Anthropic and is only accessible in the United States of America and in the United Kingdom. being able to process textual prompts and more complex ones, such as multimodal prompts, including small- and medium-sized files ([Caruccio et al., 2024](#); [Agarwal et al., 2023](#)).

Open-source development allows for greater collaboration and sharing of ideas and knowledge leading to a faster and more effective progress in the field. This, in turn, can lead to a more transparent and

ethical progress, including the public and researchers in developing countries who may in turn not have access to costly software. Furthermore, it can also help build trust and credibility in the technology since the algorithms and models are available for public scrutiny (Patwardhan et al., 2023).

Recently, open-source models (Table 2.1) such as LLaMA 2, MEDITRON and BioMistral have emerged in the biomedical domain, enabling a more transparent and reproducible research environment compared to API-based models (Doneva et al., 2024; Touvron et al., 2023; Chen et al., 2023a; Labrak et al., 2024b). The LLaMA2 was developed by Meta AI and is widely used for language modeling and text generation (Azam et al., 2024; Park et al., 2025). It was trained on massively publicly datasets (Park et al., 2025). Additionally it has shown to outperform other open-source models such as MosaicML Pretrained Transformer (MPT) and Falcon 7B in most benchmarks (Touvron et al., 2023). According to Park et al. (2025), the LLaMA model is not well-suited for answering questions or following instructions. To address this limitation, the Alpaca model was developed as a fine-tuned version of LLaMA, specifically designed to improve performance in instruction-following and question-answering tasks (Park et al., 2025). BioMistral 7B is a specialized LLM tailored for the biomedical domain, derived from Mistral 7B Instruct v0.1. It was pre-trained on PubMed Central (Labrak et al., 2024b). Meditron was built on LLaMA2 and pretrained on a comprehensively curated medical corpus such as PubMed Central and PubMed abstracts. It seeks to make medical literature more accessible through combination of high-level NLP capabilities with direct access to extensive medical databases (Sangeetha et al., 2024). Meditron is specifically tailored for medical reasoning (Chen et al., 2023a).

Some of the models mentioned previously such as MPT, Falcon 7B and Mistral 7B Instruct v0.1 constitute part of the open-source models also available nowadays. The MPT is part of the MosaicML Foundation Series and was designed to match the quality of LLaMA-7B (The Mosaic Research Team, 2023). The Falcon 7B is part of the Falcon series. These models were trained on a large-scale, high-quality English web corpus and their architecture is based on the PaLM model (Park et al., 2025; Almazrouei et al., 2023). According to Azam et al. (2024), Falcon 180B excels at conversational AI (Azam et al., 2024). The Mistral 7B is a LLM engineered for superior performance and efficiency and the fine-tuned version Mistral 7B- Instruct was created to follow instruction. According to Jiang et al. (2023), Mistral 7B outperforms LLaMA2 across many benchmarks 13B and Mistral 7B - Instruct surpasses Llama 2 13B- chat model both on human and automated benchmarks (Jiang et al., 2023).

In addition, Codellama-34 is part of the Codellama model series, a family of LLMs for code based on LLaMA2 and tailored for programming tasks. Some of its variants include Codellama-Python, specialized for python, and Codellama-Instruct, fine-tuned with human instructions and self-instruct code synthesis

data ([Rozière et al., 2024](#)). The WizardLM series is best at knowledge retrieval and processing, while Mistral-7b is performs best in instruction-following tasks. The Vicuna series are widely utilized for language modeling and text generation ([Azam et al., 2024](#)). Claude-2 is another example of an open-source LLM lauched by Anthropic. but it's only freely accessible in the United Kingdom and the United States of America . According to Agarwal et al. (2023), it has surpassed ChatGPT-3.5 when it comes for answering multiple-choice questions ([Agarwal et al., 2023](#)).

Table 2.1: Comparison of Open-source LLMs.

Model Name	Specialization	Advantages	Reference
Biomistral 7B	Biomedical domain	Demonstrates state-of-the-art performance on the multilingual medical evaluation benchmark	Labrak et al. (2024b)
Codellama series	Programming tasks	Showed fewer critical logic errors	Abbassi et al. (2025)
Falcon series	General-purpose	Falcon-180B has advanced conversational abilities; performance compared to PaLM 2 (Bard)	Azam et al. (2024) ; Almazrouei et al. (2023)
Meditron	Medical reasoning	Outperforms all open-source generalist and medical LLMs on all medical benchmarks	Chen et al. (2023a)
Mistral 7B	Instruction following	Efficient processing with emphasis on privacy-preserving techniques	Azam et al. (2024)
MPT	General-purpose	Able to handle extremely long input sequences	The Mosaic Research Team (2023)
LLaMA2 series	Language modeling	Efficient in text generation, user-friendly	Azam et al. (2024) ; Hadi et al. (2023)
WizardLM series	Knowledge retrieval	Comprehends and follows complex instructions, adapted to the health domain	Azam et al. (2024) ; Chen et al. (2023a)

Evaluation of LLMs

An essential step to assessing the models performance is through evaluation. To evaluate LLMs, it's important to take use key statistics indicators such as precision, recall and F1 scores. Another critical aspect of this evaluation is how well the model's answers align with real-world truth and verifiable facts, which falls under the broader task of information extraction. This is of great importance to trust and efficiently use these models (Chang et al., 2023; Azam et al., 2024).

Two common evaluation methods used are: automatic evaluation and human evaluation. If the evaluation criterion can be automatically computed it is categorized as automatic evaluation. Otherwise, it is categorized into human evaluation (Chang et al., 2023).

Automatic Evaluation Automatic evaluation involves using standard metrics and evaluation tools to evaluate model performance (Chang et al., 2023). The metrics used to evaluate LLMs performance can be categorized into three types: Multiple-Classification (MC), Token-Similarity (TS) and QA. The MC metrics can be applied to many tasks with developed benchmark datasets, the TS metrics evaluates the if the texts generated aligns with the expected texts and the QA metrics is specifically tailored for the task of Question-Answering (Hu and Zhou, 2024).

Multiple-Classification metrics The effectiveness of a LLM in classifying text into multiple groups, each serving as a distinct label, covers MC metrics. Some of the MC metrics available are: accuracy, recall, precision, F1 scores, calibrations, fairness and robustness metrics (Hu and Zhou, 2024; Johansson et al., 2021; Chang et al., 2023).

Accuracy is the measure of how correct a model is on a given task. This concept can vary dependent on different scenarios and tasks (Chang et al., 2023). For example, when considering a two-label problem, distinguishing "positive" and "negative" cases, the gold standard label for each case is provided by benchmark datasets and the LLM performs classification and generates predictions. This process generates a 2x2 chart where accuracy can be expressed as (Hu and Zhou, 2024):

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{True Positives} + \text{False Negatives} + \text{False Positives} + \text{True Negatives}} \quad (2.1)$$

Other ways to measure accuracy is through metrics like Exact Match (EM), F1 or the ROUGE metric. The EM evaluates if the model's output in text generation tasks matches the reference answer. If it's an exact match then the EM is equal to 1, if not, then it's 0. Some authors claim that the F1 score is part of evaluating the accuracy of a binary classification model, combining the model's precision and recall

(Chang et al., 2023).

Recall, which can also be referred as Sensitivity or True Positive Rate, can be used to measure the proportion of positive detections among the actual "positive" instances. It measures how well an LLM can detect positive instances. The formula is given by (Hu and Zhou, 2024):

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad (2.2)$$

Precision indicates the proportion of actual positive instances among all instances identified as positive, measuring the ability of an LLM to filter out negative instances that are falsely labeled as positive (Hu and Zhou, 2024). It indicates the model's ability to avoid false positives Chang et al. (2025).

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \quad (2.3)$$

But, often high Recall matches with low precision, and vice versa. Therefore, the F1 score balances these two metrics, reflecting an overall performance of the model, using the following formula (Hu and Zhou, 2024; Chang et al., 2025):

$$\text{F1} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \quad (2.4)$$

The previous metric ranges from 0 to 1 and the closer to 1, the better the Recall and Precision (Hu and Zhou, 2024).

On multiple classification tasks, calibrations metrics may be also used (Johansson et al., 2021). These pertain to the degree of agreement between the confidence level of the model output and the actual prediction accuracy. These metrics include Expected Calibration Error and Area Under the Curve. Fairness metrics evaluate whether the model's performance is equal across different groups. Such metrics can be: Demographic Parity Difference, that evaluates if the model's predictions are distributed equally across different population groups and Equalized Odds Difference that aims to ensure that the model provides equal error rates across different populations. When faced with challenging inputs—such as adversarial attacks or shifts in data distribution—robustness becomes an important metric for evaluating model performance. Such metrics include Attack Success Rate and Performance Drop Rate (Chang et al., 2023).

Prior to understanding the interpretation of these metrics, it's important to take into consideration the sample and population. The population refers to the global corpus on which the models are evaluated, while the sample comprises the test datasets (Hu and Zhou, 2024).

Token-Similarity Metrics These metric encompass the similarity between the texts generated by LLMs and the references, assessing the quality of the produced desired sequence of words given contextual

information. These metrics are frequently used to assess the behavior of the LLM in tasks such as machine translation and text summarization. Some of the metrics used for this are: Perplexity and BertScore ([Hu and Zhou, 2024](#)).

Question-Answering Metrics They're used when the LLM is required to identify an answer to a specific question given a contextual passage. The model finds the answer to a question by predicting the start and end positions of the answer within a given passage. It ensures the start comes before the end, filters out invalid predictions, and selects the most likely answer span based on prediction probabilities. The three most popular metrics of this type are: Strict Accuracy, Lenient Accuracy and Mean Reciprocal Rank ([Tsatsaronis et al., 2015](#)).

Human Evaluation In some cases, automatic evaluation is not suitable and human evaluation becomes the natural choice. For example, in cases that involve open generation tasks like summarization where there might be multiple valid answers. It involves evaluating the model's quality and accuracy of the model-generated results through human participation ([Chang et al., 2023](#)).

Some of the essential aspects of the human evaluation methods of LLMs encompass the number of evaluators, the evaluation rubrics and the evaluator's expertise level. The number of evaluators is essential to ensure adequate representation and statistical inference. The evaluation rubrics ensure the LLMs performance in syntax, semantics and context is achieved. Such rubrics include: accuracy, relevance, fluency, transparency, safety and human alignment. Finally, the evaluator's expertise level is of critical consideration and it's fundamental to ensure they have the necessary background knowledge to comprehend and assess the domain-specific text generated by LLMs ([Chang et al., 2023](#)).

Challenges associated with the models

Nevertheless, there are some drawbacks associated with these models. For example, since the models require vast amounts of data for pre-training, collecting and curating these datasets can be very challenging. Also, reading or assessing the quality of these datasets is impossible and duplicates are often incorporated ([Hadi et al., 2023](#)). This duplication occurrence can make the model biased and make its answer degraded ([Hadi et al., 2023](#); [Raiaan et al., 2024](#)). If the datasets contain biases related to factors such as race, gender, or socioeconomic status, these biases may be internalized and reproduced by the model. Due to the vast amount of data they need, significant computational costs with thousands of compute hours and energy consumption are demanded - making it financially and environmentally heavy ([Hadi et al., 2023](#)). The lack of transparency poses risks, such as potential legal issues if the model's training data infringes

copyright laws, potentially leading to the model becoming unusable (Bolton et al., 2024).

Moreover, huge quantities of energy are needed to train LLMs at scale. This can lead to a significant role in carbon footprint. Also, the fact they are trained on historical data from the Internet that might become outdated, may generate untrue answers regarding recent events (Raiaan et al., 2024).

Other problems such as hallucination and the lack of explainability are also common with LLMs. Hallucination happens when nonsensical or unfaithful text is generated, due to the model's attempt to fill in gaps in knowledge or context, assuming on the patterns it learning during training. Lack of explainability happens because these models have billions of parameters, their scale and complexity makes it challenging for humans to understand and interpret (Hadi et al., 2023; Hu and Zhou, 2024).

In short, these costs and challenges can make language model technology difficult to access or unsuitable for biomedical and healthcare organizations that have limited budgets and strict privacy requirements.

2.1.3 Prompt Engineering

LLMs rely on the user providing an input “prompt” to which the model produces an output in response. These can be textual, videos, audio or a combination thereof. The better the prompts, the better the results across the desired tasks. Therefore, the use of prompts has been studied and reviewed in order to improve results (Schulhoff et al., 2025).

According to Schulhoff et al. (2025), there are 6 major categories of text-based prompt techniques: zero-shot, few-shot, thought generation, ensembling, self-criticism and decomposition. Zero-shot prompting makes use of zero exemplars and some of the techniques include assigning a specific role to the LLM in the prompt or instructing the LLM to rephrase and expand the question before generating the final answer. Few-shot prompting involves using examples to instruct the output of the LLM. Some aspects like the order and quantity of the exemplar can affect the model's output and some of the techniques include Self-Generated In-Context Learning where a model is leveraged to automatically generate exemplars and prompt mining which is the process of discovering optimal “middle words” in prompts through large corpus analysis. Thought Generation involves prompting the LLM to articulate its reasoning while solving a problem. Ensembling is the process of using multiple prompts to solve the same problem and afterwards, aggregating these responses into a final output. Self-criticism involves using the LLMs criticisms of its own outputs, either by judgment (e.g., is this output correct) or prompting the LLM to provide feedback. Finally, decomposition involves decomposing complex problems into simpler sub-questions (Schulhoff et al., 2025).

Some of the prompting engineering techniques which are used to automatically optimize prompts

include Meta Prompting and Prompt Optimization with Textual Gradients (ProTeGi). Meta Prompting is the process of instructing an LLM to generate or improve a prompt or prompt template and ProTeGi is the multi-step process which includes the following steps: passing a batch of inputs through the template, then the output, ground truth, and prompt into another prompt that criticizes the original prompt. New prompts are generated from these criticisms and then a bandit algorithm is used to select one ([Schulhoff et al., 2025](#)).

Retrieval Augmented Generation

LLMs are prone to fabricating facts, lack knowledge when dealing with specific domains or highly specialized queries and lack transparency in answers. By retrieving important information from external sources, accurate, real-time and domain-specific context is provided to LLMs showing to be effective in enhancing answer accuracy, reducing the fabrication of facts and increasing the trust in model outputs. Furthermore, it also facilitates knowledge updates and the introduction of domain-specific knowledge ([Gao et al., 2024](#)). RAG, in this context, is a paradigm in which information is extracted from an external source and inserted into the prompt, increasing the performance of the LLM in knowledge retrieval ([Schulhoff et al., 2025](#)).

An RAG system has the following workflow: it utilizes an external knowledge base to enhance the generation system, such as external documents. Each of these documents contains a set of chunks, and afterwards they are transformed into vector embeddings using an embedding model. Then, an input in the form of a query is made, which is also embedded as a vector, and the retriever in the RAG system retrieves top- k chunks, the ones most relevant to the query ([Wu et al., 2025](#)). The previous workflow can be usually divided into multiple processing steps: query classification (determining whether retrieval is necessary for a given input query), retrieval (efficiently obtaining relevant documents for the query), reranking (refining the order of retrieved documents based on their relevance to the query), repacking (organizing the retrieved documents into a structured one for better generation), summarization (extracting key information for response generation from the repacked document and eliminating redundancies) modules. Each processing step requires making decisions on which ways to properly handle them, therefore the workflow of RAG can become complex and challenging ([Wang et al., 2024b](#)).

When it comes to the retrieval step, chunking, embedding and having a vector database is necessary beforehand. Chunking the documents into smaller segments (paragraphs or sentence blocks) is crucial to enhance the retrieval precision and avoid length issues in LLMs. Different levels of chunking can be applied - token-level chunking, semantic-level chunking and sentence-level chunking. Additionally, chunk size is important to take into consideration since it can significantly impact performance and choosing the right

model to conduct the embeddings is crucial to secure an effective semantic matching of queries and chunk blocks. Moreover, the right embedding model is important for an effective semantic match of queries and chunk blocks. Then, the embedded vectors generated from the embedding model on the chunks must be stored in an appropriate vector database. The top- k chunks end up being selected from a corpus that was built based on the similarity between the query and the documents, as mentioned before. Afterwards the LLMs use these chunks to formulate an appropriate response to the query. The reranking phase comes next to enhance the relevance of the retrieved documents, making the most relevant information appear at the top of the list. This ensures that the documents are reordered effectively with increased similarity between the query and the top-ranked documents. Next, the repacking occurs. This allows for the ranked chunks to be reordered depending on the relevancy scores. The results that end up being generated may contain redundant or unnecessary information. Therefore, the summarization of the retrieved documents is important for the RAG pipeline. The summarization process can either be extractive or abstractive. The first ones segment text into sentences and ends up scoring and ranking them base on their importance and the latter synthesizes information from multiples documents to rephrase and generate a cohesive summary (Wang et al., 2024b; Polan, 2025).

RAG Paradigms There are three distinct RAG paradigms as presented in Figure 2.3: Naive, Advanced and Modular RAG (Neha et al., 2025).

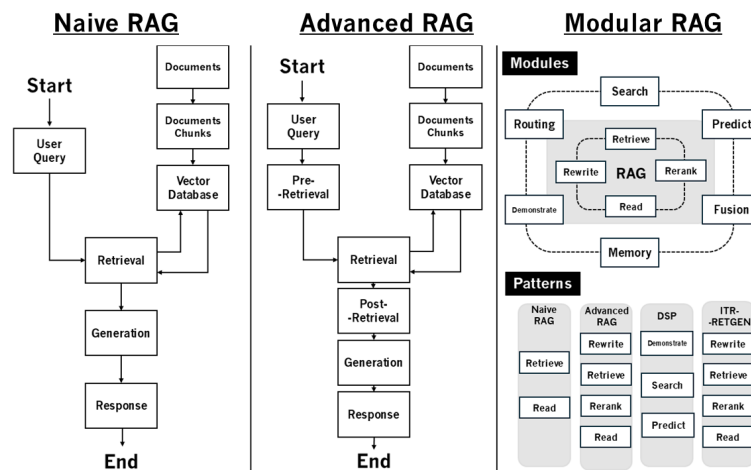


Figure 2.3: RAG paradigms: Naive, Advanced and Modular.

Naive RAG This type of RAG paradigm is the simplest and earliest methodology, and it's a foundational framework for integrating retrieving systems with LLMs. With this type of architecture there is no need to handle complex queries. It first indexes relevant data, then retrieves information based on user

queries and, in the end, generates responses using the retrieved content, which is also characterized as a "Retrieve-Read" framework (Gao et al., 2024; Neha et al., 2025).

There are primarily three areas of challenges associated with this type of RAG paradigm: retrieval quality, response generational quality and the augmentation process. Firstly, there are many issues when it comes to the retrieval quality, such as low precision and low recall. Low precision happens when irrelevant chunk selection is conducted, thus inducing hallucination and mid-air drop issues and low recall happens due to the prevention from the LLMs in obtaining sufficient context to synthesize an answer; Moreover, issues with response generation quality can also arise: hallucination because of the fabrication of answers that don't exist in the context; irrelevance is another issue where the model generates an answer that doesn't address the query; toxicity or bias, happens when a harmful or offensive response is generated by the model. Finally, the augmentation process faces several key challenges: effectively integrating retrieved context with the generation task, avoiding incoherence and redundancy, assessing the relevance of multiple passages, maintaining consistency across different writing styles, and preventing the model from over-relying on retrieved content instead of producing original, synthesized output (Neha et al., 2025; Gao et al., 2024).

Advanced RAG The limitations of the Naive RAG paradigm led to the creation of an improved paradigm, the Advanced RAG. This type of paradigm focuses on enhancing retrieval quality by employing pre-retrieval and post-retrieval strategies (Gao et al., 2024). With the use of sophisticated algorithms that can combine a series of ideas, such as rerankers, fine-tuned LLMs and feedback loops, enhancements in terms of accuracy, adaptability and performance were possible (Harkar, 2025).

When a user submits a query, it is first converted into a high-dimensional vector using an embedding model that captures its semantic meaning. This encoded query is then used to search a large knowledge database through hybrid retrieval, combining dense vector (semantic) and sparse (keyword-based) methods to identify relevant documents. The retrieved documents are reranked based on their contextual relevance to the query. Next, the model fuses the encoded contexts from these documents to ensure a coherent and contextually aligned response. Finally, a LLM generates the answer based on the retrieved information. Over time, a feedback loop involving techniques like active learning, reinforcement learning, and retriever-generator co-training refines both retrieval and generation processes, using implicit signals (e.g., user clicks) and explicit feedback (e.g., ratings or corrections) to continually improve accuracy and relevance (Harkar, 2025).

Modular RAG As described by Gao et al. (2023), the Modular RAG paradigm turns the RAG process from a linear pipeline into a flexible structure with modules that have the ability to branch and merge as needed, offering adaptability and versatility (Gao et al., 2024). The goal is to enhance the older paradigms from a single rigid design, turning them into a framework with various subcomponents. For example, if a query requires visual data an image retrieval module can be activated or, instead, if a query requires multi-step reasoning, a logic chain module can be used (Polan, 2025).

The introduction of new modules, specialized components, enhances retrieval and processing capabilities. The Search module enables direct searches across diverse data sources—such as search engines, databases, and knowledge graphs—using LLM-generated queries. RAGFusion expands user queries into multiple perspectives through parallel searches and intelligent reranking to reveal deeper insights. The Memory module uses the LLM’s memory for iterative self-enhancement, aligning retrieval more closely with data patterns. Routing optimizes how queries navigate different data sources, choosing the best path for summarization or data merging. The Predict module minimizes noise and redundancy by letting the LLM generate context directly. Finally, the Task Adapter customizes RAG for specific downstream applications, automating prompt retrieval and generating task-specific retrievers. Together, these modules streamline retrieval, improve relevance, and boost flexibility across a wide range of tasks (Gao et al., 2024).

Module rearrangement such as Demonstrate-Search-Predict framework and the iterative Retrieve-Read-Retrieve-Read flow of ITERRETGEN are good examples of the dynamic use of module outputs to improve performance.

2.2 Mechanism and Insights of Cell-Cell Communication

2.2.1 Types of CCC

CCC is often mediated by interactions of various types of protein, including ligand-receptor, receptor-receptor, and extracellular matrix-receptor interactions (Armingol et al., 2021).

Ligand-Receptor Interaction

The most predominant mechanism in CCC is Ligand-Receptor Interaction (LRI), in which extracellular protein signals (ligands) are released by a “sender cell” and, afterward, become attached to receptor proteins in the membrane of the “receiver cell”, which in turn allows activation of downstream gene expression responses (Almet et al., 2021; Armingol et al., 2024). Ligands responsible for mediating CCC include hormones, growth factors, chemokines, cytokines and neurotransmitters (Armingol et al., 2021).

This type of interactions is at the base of all biological events occurring in living cells (Guryanov et al., 2016). The way cells communicate can be studied by knowing which ligands bind to receptors, following an analysis of its effects on target gene expression in the receiving cells (Browaeys et al., 2020). Thus, it is essential to understand CCC at the single-cell level (Su et al., 2024).

Some LRIs include PPIs which are essential for many biological functions such as DNA replication and transcription, signaling pathways, and cell metabolism (Rehana et al., 2024). The way PPI occur involves firstly the *primary recognition* of the receptor by a corresponding ligand at a large distance by electrostatic forces, followed by *orientation* and change of structural conformation in order to achieve the proper interface contact and finally the *physical binding* of the two molecules (Guryanov et al., 2016) (Figure 2.4).

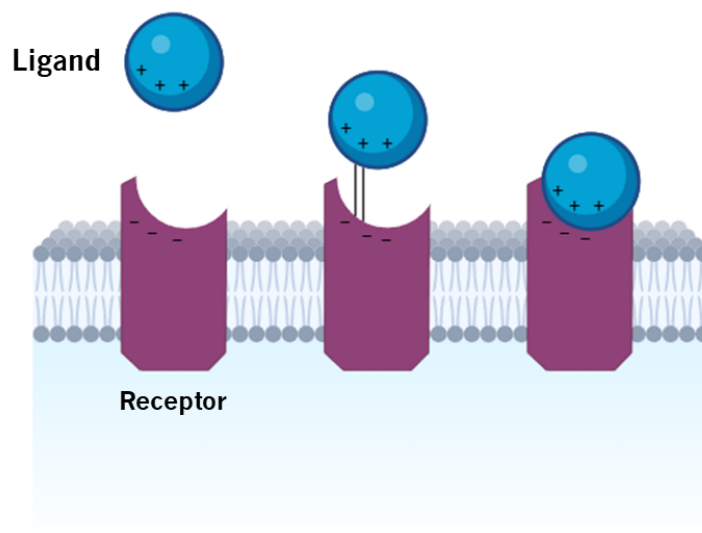


Figure 2.4: Ligand-receptor complex formation from left to right: primary recognition, orientation and physical binding. Adapted from (Guryanov et al., 2016).

Receptor-Receptor Interaction

The concept of Receptor-Receptor Interaction (RRI) refers to the direct physical contacts between receptor proteins that can result in the formation of dimers or higher-order oligomers at the cell membrane. RRI occur in neurons, astrocytes and in cells outside the central nervous system, such as in cells of the cardiovascular and endocrine systems and in cancer cells (Guidolin et al., 2019).

Cell receptors belong to different families such as: matrix (e.g., integrins), ligand-gated ion channels, voltage-gated ion channels, enzyme-linked, intracellular and G protein-coupled receptors (GPCRs) (see Figure 2.5) ([Guidolin et al., 2019](#)). The most common being the family of GPCRs which, in mammals, are very important to all physiological processes and are very common targets for drugs ([Guidolin et al., 2019](#)).

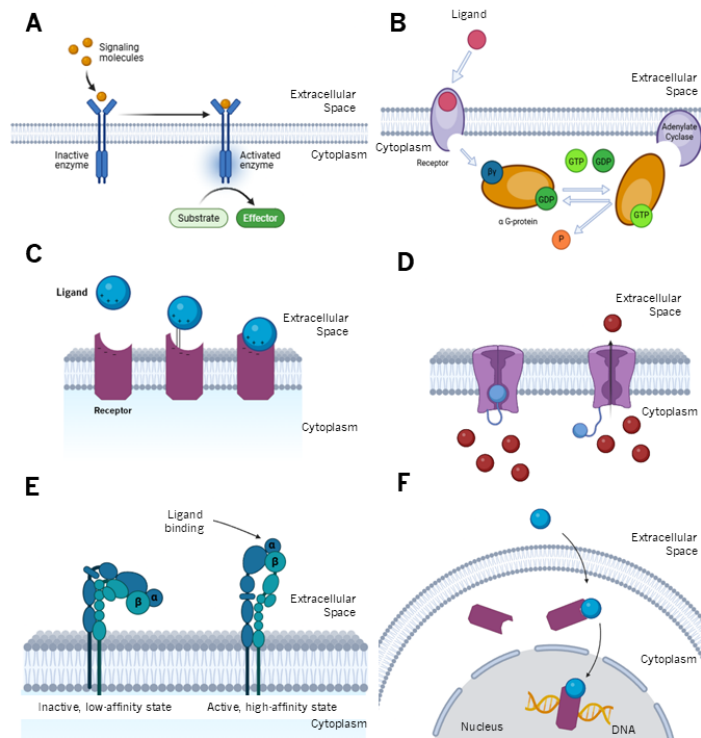


Figure 2.5: **A**- Enzyme-linked receptors are transmembrane proteins with their ligand-binding domain on the outer surface of the plasma membrane where their cytosolic domain either has an intrinsic enzyme activity or associates directly with an enzyme (Alberts et al., 2002). Adapted from (Meriney and Fanselow, 2019a); **B** - GPCRs may assemble as homo- or hetero/oligomeric complexes, either constitutively or by ligand promotion. Ligand binding induces conformational change in the transmembrane receptor, enabling interaction with the β -subunit of a G protein, through exchange of guanine nucleotides. GTP-bound α -subunit then activates adenylate cyclase. Adapted from (Giorgi et al., 2010). **C** - Ligand-receptor complex formation. Adapted from (Guryanov et al., 2016). **D** - Voltage-gated ion channels. The difference in membrane potential caused by the ion migration causes a structural change in the channel leading to its opening. Adapted from (Jacob, 2017) **E** - Integrins are represented. They are molecules that mediate interactions between cells and and of these cells with the extracellular matrix (ECM). Adapted from (Mezu-Ndubuisi and Maheshwari, 2021). **F** - Intracellular receptors. Hydrophobic molecules can diffuse across the plasma membrane and bind to intracellular receptors. After binding, most cytoplasmatic receptors become active transcription factors and diffuse into the cell nucleus and bind to DNA to regulate gene expression. Adapted from (Meriney and Fanselow, 2019b)

The complexes formed by RRI are considered of great importance in neurophysiology, neuropsychopharmacology and have become appealing for the development od therapies in serious diseases of the central nervous system, such as depression and schizophrenia (Cheng et al., 2023).

Extracellular Matrix-Receptor Interaction

The extracellular matrix is a complex network of structural and functional macromolecules that plays a key role in shaping tissues and organs, as well as maintaining cellular and tissue integrity. Cells interact with it through transmembrane proteins or other surface-associated components, which in turn regulate essential processes such as adhesion, migration, differentiation, proliferation, and apoptosis, either directly or indirectly ([Analysis, 2025](#); [Nersisyan et al., 2021](#)). The Extracellular Matrix-Receptor Interactions are important type of these interactions and they happen between cellular receptors, such as, integrins or cadherins, and the noncellular component of tissue, which biochemically and structurally supports cells ([Nersisyan et al., 2021](#)).

2.2.2 Importance of studying CCC

If cells communicate incorrectly, the development of tissues, organs and immune responses can be affected ([Su et al., 2024](#)). Therefore, understanding the complex biological pathways present in CCC is critical to exploring the nature of biology and the mechanisms of disease and drug development ([Azam et al., 2024](#); [Su et al., 2024](#)). Reprogramming of cellular signaling pathways can potentially lead to improvement of cell-based therapies ([Cheng et al., 2023](#)).

2.2.3 How to predict CCC

Prior to recent advancements, CCC could only be studied through *in vitro* experiments consisting of one or two cell types and a selection of a few genes ([Almet et al., 2021](#)). Since CCC involves multiple cells and a considerable amount of genes, the old approach misses key aspects ([Almet et al., 2021](#)).

The standard procedure to study CCC involves the following steps: first, a gene expression matrix is filtered to include only ligands and receptors, secondly, the average expression of each gene is calculated across single cells of a specific cell type ([Armingol et al., 2024](#)). After, for each cell type pair, LRIs are assessed based on ligand expression in sender cells and receptor expression in receiver cells ([Armingol et al., 2024](#)). A communication score is then calculated for each LRI, often using measures like the product, mean, or geometric mean of ligand and receptor expression levels ([Armingol et al., 2024](#)). Finally, statistical analyses can be further performed to identify significant interactions.

Single-Cell Omics

The study of CCC has been significantly advanced by transcriptomics, especially from bulk and single-cell omics technologies, such as single-cell RNA sequencing (scRNA-seq), providing deeper insights into cellular interactions by their ability to analyze tissue composition in high-resolution and offering researchers fresh perspectives on cell communication mechanisms, cell functions, and the organization of cell populations ([Almet et al., 2021](#); [Browaeys et al., 2020](#); [Su et al., 2024](#); [Armingol et al., 2021](#); [Cheng et al., 2023](#)). Both these approaches make use of transcriptional profiling to decipher CCCs in any multicellular community relying in gene co-expression, where for each interacting pair, one gene originates from one cell, while the other comes from the interacting counterpart ([Armingol et al., 2021](#)).

There are certain challenges associated with using scRNA-seq to predict CCC such as the process of clustering, which is essential to characterize cellular heterogeneity, identifying rare cell types and analyzing cell-cell interactions ([Cheng et al., 2023](#)). Currently, the clustering methods struggle to accurately determine the number of cell types or clusters and they often fail to incorporate intrinsic hierarchical structure among cells. Therefore, the manual classification of cells rely heavily on domain expertise and the current clustering algorithms may lead to potential inaccuracies ([Cheng et al., 2023](#)).

Moreover, these type of technologies are still costly, and single-cell datasets, especially those linked to clinical patient cohorts, remain more limited compared to, for example, bulk data ([Villemin et al., 2023](#)). Also, scRNA-seq data processing involves stricter protocols for library preparation and low starting material and there is a lack of analytical approaches required to accommodate large datasets generated during scRNA-seq experiments ([Adil et al., 2021](#)).

Bulk Data

In addition to the existing single cell omics technologies, there has been a need to free cellular networks from bulk data, as it provides a more sensitive analysis to low-abundant molecules ([Villemin et al., 2023](#)). Compared to scRNA-seq, bulk analysis has limitations, particularly in detecting gene expression in rare cell types and identifying the cellular origin of proteins involved in cell-cell interactions ([Armingol et al., 2021](#)). Overall, single-cell technology was found to be more conclusive compared to sequencing bulk data ([Adil et al., 2021](#)).

Spatial Transcriptomics

CCC analysis, in a spatial context, can become empowered with the inclusion of spatial transcriptomics (ST), a form of bulk data, to help users visualize gene expression across spatial spots (specific locations

in tissues) and perform tasks such as clustering cells or detecting cellular neighborhoods (Almet et al., 2021; Armingol et al., 2024; Villemin et al., 2023). This allows for the reduction of the false-positive links that come up when using scRNA-seq data to infer CCC, since cells only communicate directly over a limited spatial distance-critical information lost in the scRNA-seq data (Almet et al., 2021). Usually, at the cost of compromised cellular resolution, coverage, or sequencing depth (Almet et al., 2021).

2.2.4 Computational Approaches to Study CCC

Standard PPI databases

The use of transcriptomics to decipher CCCs relies mainly on the use of literature-curated lists of interacting proteins, facilitating the interpretation of biological results, with the predominant class being the LRI. These databases have been essential to decipher CCC in all strategies. Few predicted PPIs in these databases have been validated which raises concern for the false positives (Armingol et al., 2021).

Some of the most used ones are BioGRID, IntAct and STRING. While BioGRID and IntAct encompass a group of databases of experimental interaction evidence curated from literature, STRING belongs to the group of composite databases (Szkarczyk et al., 2024).

However, many of the reported ligand-receptor pairs lack information regarding protein complexes involving multiple units, greatly increasing false-positive CCC predictions. Since some proteins require multisubunit assembly to function, if a subunit blocks a ligand-receptor interaction then the consequent communication can become affected. This has led to the creation of recent computation tools (Armingol et al., 2021).

Recent computational tools

Some of the tools that have been created recently include multimeric proteins and interactions between complexes of both ligands and receptors, better representing functional LRIs (Armingol et al., 2021). Some of these computational approaches available to detect the main type of CCCs (LRIs) can be grouped into four types depending on the mathematical framework: (1) expression permutation-based tools, (2) difference-assembly-based tools, (3) network-based tools, and (4) tensor-based tools (Su et al., 2024; Armingol et al., 2021). Most of the tools use gene expression information obtained by scRNA-Seq (Dimitrov et al., 2022).

Expression permutation-based tools Expression permutation-based tools are the most widely used among the four previously listed (Armingol et al., 2021). They make use of different techniques to calcu-

late communication scores for each ligand-receptor pair and assess their significance by comparing them to a null model (Su et al., 2024). This is done through methods such as clustering label arrangement, non-parametric testing, or empirical approaches (Su et al., 2024). Examples of these tools include CellPhoneDB, CellChat, ICELLNET (Su et al., 2024). These tools consider multi-subunit protein complexes, one of the major limitations of many CCC methods (Su et al., 2024; Armingol et al., 2021).

CellPhoneDB recognizes that numerous receptors and ligands operate exclusively as heteromers and is recognized for its comprehensive cellular communication ligand-receptor database. In the meantime, it neglects some vital signaling cofactors which CellChat integrates, such as soluble agonists, antagonists and both stimulatory and inhibitory membrane-bound co-receptors (Su et al., 2024). CellChat provides cell interaction network diagrams and communication pathway analysis facilitating the analysis of inter-cellular interactions and communication networks. While CellPhoneDB identifies enriched ligand-receptor interactions between two cell groups by focusing on interactions that are highly specific to those groups. In contrast, CellChat highlights interactions based on ligands and receptors that are strongly overexpressed, using the mass action principle to estimate the strength of their association (Su et al., 2024).

Moreover, ICELLNET sums all LRI production scores across two clusters and that's how it calculates an overall CCC score and to determine how strong the interactions are, it multiplies the geometric mean of the ligand expression and the receptor expression. Besides that, it also groups the LRI into biological families, therefore adding biological context to the predictions.

Difference-assembly-based tools Difference-assembly-based tools, also called differential combination-based tools, such as PyMINER, iTALK, and CellTalker, identify LRIs by analyzing differentially expressed genes between cell clusters in scRNA seq (Su et al., 2024; Armingol et al., 2021). While these methods are effective in capturing specific LRIs within a dataset, they may overlook common interactions shared across all cell groups (Su et al., 2024; Armingol et al., 2021).

PyMINER builds gene co-expression networks using Spearman correlations, and combines these with known PPI networks to strengthen the biological relevance of the predicted interactions. While this tool identifies cell types based on the enrichment of gene pathways that are active in specific subgroups and detects changes in signaling by looking at differentially expressed LR pairs and the pathways they're involved, iTalk focuses specifically on communication between tumor and normal cells, and organizes ligand-receptor pairs into categories like cytokines, growth factors, immune checkpoints, and others (Su et al., 2024).

CellTalker identifies unique interactions between clusters by using differentially expressed ligands and receptors within individual clusters and assuming that both the ligand and receptor must be expressed uniformly across communicating cells (Su et al., 2024).

Network-based tools The Network-based tools approach is used by tools that leverage gene connectivity properties (Su et al., 2024; Armingol et al., 2021). This process involves intercellular signaling, intracellular signal transmission, and amplification through specific pathways, ultimately influencing downstream transcription factors and gene regulatory networks (Su et al., 2024). To account for these complexities, various computational methods have been developed, including CCCExplorer, NicheNet, scMLnet, SoptSC, Scriabin, CytoTalk, RNA-Magnet, and ContactTracing (Su et al., 2024). Even if their strength is that they also consider the expression levels of downstream signaling targets, they also overlook gene regulation rules, making them unable to detect signaling crosstalk, where one receptor's signal affects another's (Armingol et al., 2021).

CCCExplorer uses a statistical approach similar to Fisher's method to create a detailed network showing different signalling pathways. It analyses the expression levels of ligands, receptors, and downstream transcription factors and after it identifies the most important interactions (Su et al., 2024).

Tensor-based tools Tensor-based tools represent the most mathematically complex approach and they are based on tensor analysis. scTensor, for example, through an innovative approach, identifies key LRIs by modeling relationships between ligand expression, receptor expression, and ligand-receptor pairs using a tensor (Armingol et al., 2021; Su et al., 2024). It uses tucker decomposition to identify important LRIs specific to cell types and it also uses a potential LRP database automatically generated by interactions from STRING and annotations from Swissprot. While these tools are capable of analyzing communication across all cell pairs at once and uncovering interactions between various cellular environments, the results from tensor decomposition can be harder to interpret compared to simpler methods.

NicheNet NicheNet can be included into the network-based tools (Su et al., 2024). It's a method that predicts ligand-target links between interacting cells by integrating their expression data with existing information on signaling and gene regulatory networks. The gene expression data comes from human or mouse and it's merged with a preexisting model based on previously acquired knowledge about ligand-to-target signaling pathways. NicheNet can also incorporate intracellular signaling, predicting which ligands affect the expression of other cells, which target genes are affected by each ligand and which signaling mediators may be involved (Browaeys et al., 2020).

The ligand-target regulatory potential is calculated by first gathering multiple complementary data sources that include ligand-receptor interactions, signal transduction, and gene regulatory interactions. Next, these data sources become integrated into weighted networks that represent signaling pathways (protein-protein interactions) from ligands to transcriptional regulators and regulatory interactions

between transcriptional regulators and target genes. Following this, a model-based parameter optimization is used to assign weights to each data source, prioritizing those that contributed most to improving the accuracy of ligand-target predictions. Finally, the regulatory potential score between ligands and target genes is calculated ([Browaeys et al., 2020](#)).

NicheNet works by first predicting which ligands, produced by the sender cells, are most likely to affect gene expression in the receiver cells. Such prediction is achieved by comparing predicted gene-ligand interaction with the actual changes in gene expression observed in the receiver cells, resulting in a list of ligands, ranked by their likeness of responsibility for the gene expression changes observed in the received cells. Next, it searches for specific genes (target genes) in the receiver genes that might be regulated by the ligands obtained. For each target gene, a “regulatory potential score” is generated, indicating how strongly the model suggests that the ligand controls that particular gene. Finally, NicheNet helps visualize the pathways that connect the two elements by showing which molecules and interactions are involved in transmitting the signal from the ligand to the target gene ([Browaeys et al., 2020](#)).

The validation and optimization of the NicheNet prior model of ligand-target regulatory potential is done by an evaluation process using public transcriptome data from many cell type, collected before and after stimulation by one or two ligands. Moreover, the model’s performance was assessed by two evaluation methods: target gene prediction and ligand activity prediction. The first, evaluates if the model is good at predicting which genes are directly regulated by the ligand and the second tests how well NicheNet can determine which ligand caused the observed effects on the genes, based on how well the predicted ligand-target relationships match the real data ([Browaeys et al., 2020](#)).

2.2.5 Computational minimization of false discoveries

Applying statistical modeling can help discard artefactual or non-specific CCC inferences ([Armingol et al., 2021](#)). They include: permutation-based analyses, subsampling analyses and enrichment analyses ([Armingol et al., 2021](#)). Permutation-based analyses prioritizes cell type-enrichment ligand-receptor pairs, allowing to discard results originating from random noise ([Armingol et al., 2021](#)). In the meantime, sub sampling has been applied to evaluate the robustness of the CCC results ([Armingol et al., 2021](#)). Using random sub-sample of the original data, one can rerun the CCC analysis and compute true-positive and false-positive rates with respect to the original dataset ([Armingol et al., 2021](#)). By making use of enrichment analyses, one can reduce false discoveries. There are also many other strategies that can be used to validate CCC mechanism computationally such as Wilcoxon’s test and cross-validation ([Armingol et al., 2021](#)).

2.2.6 Validation of CCC predictions

Inferring CCC mechanisms can generate different results, depending on the approach used (Armingol et al., 2021). Therefore, the use of validation mechanisms is crucial for confirming associations with phenotypes and behaviors of cell communities by establishing appropriate benchmarks (Almet et al., 2021). One way to do so is by using prior biological knowledge and to evaluate it with indicators such as robustness and enrichment analysis (Almet et al., 2021). However, most times, the biological truth is unknown (Almet et al., 2021).

Experimental Methods

CCC mechanisms inferred computationally can be validated using experimental methods (Armingol et al., 2021; Almet et al., 2021). There are three types of experimental methods that can be used for this purpose: confirmation of the expression of proteins thought to be involved in CCC, visualization of interactions between gene products expressed in neighboring cells and by assessing *in vivo* or *in vitro* experiments using activators or inhibitors of the interacting molecules or genetic manipulation of cells to observe effects consistent with the predicted communication (Armingol et al., 2021). Some conventional experiments involve techniques such as fluorescence *in situ* hybridization, fluorescence resonance energy transfer and immunostaining (Armingol et al., 2024). However, it's difficult to validate CCC coming from large-scale scRNA-seq studies since they are limited to examining only a few cells and LRIs in a given experiment (Armingol et al., 2024).

Moreover, validation of the CCC methods by prior biological knowledge and using indicators like robustness and enrichment analysis is crucial. However, it can be a difficult task to do so since the biological truth is often unknown (Almet et al., 2021).

2.2.7 Challenges

There are several challenges associated with inferring the interaction between single cells in the context of scRNA-seq analysis, such as data sparsity and scalability. Data sparsity occurs because many genes have zero expression in individual cells and it can further complicate the analysis and interpretation of CCC. Meanwhile, scalability is a big issue since datasets are commonly large ones, presenting computational challenges. These challenges can be tackled using several approaches: with denoising algorithms, implementing zero-preserving communication scores, using subsamples, prioritizing pairs of cells based on specific properties of interest (Armingol et al., 2024).

The scRNA-seq datasets also come frequently with a considerable amount of noise due to too little or poor-quality data or samples, which often leads to incorrect conclusions. To address these issues, normalization is usually applied. However, bulk-based strategies such as fragments per kilo base of transcript counts of per million mapped fragments (FPKM) or transcripts per millions (TPM) have been reported as unsuitable due to the high zero-inflated and higher technical noise nature of the datasets. Some methods have been developed for normalizing the scRNA-seq data such as $O(n \log n)$ ([Adil et al., 2021](#)).

Moreover, since the data comes from cells that show high dimensions, such as a large number of genes, it's necessary to make use of dimensionality reduction. For this, two common methods employed are principal component analysis (PCA) and T-distribution stochastic neighbor embedding (t-SNE) ([van der Maaten and Hinton, 2008](#); [Adil et al., 2021](#)). Both methods have their own limitations since PCA doesn't effectively amount to complexities of the structure of the scRNA-seq data and t-SNE takes a long time to reach the final result because of the magnitude of the scRNA-seq data. Other algorithms such as the UMAP or methods such as linear discriminant analysis have been applied instead ([Adil et al., 2021](#); [McInnes et al., 2020](#)).

2.2.8 Future Insights

Multi-omics integration

Furthermore, transcriptomics may not represent a fully accurate view of intercellular communication, as transcript and protein abundances can be uncoupled by post-transcriptional and post-transcriptional processes ([Armingol et al., 2021](#)). Therefore, borrowing information from other omics technologies can improve CCC inference, validation, and benchmarking ([Armingol et al., 2021](#); [Almet et al., 2021](#)).

Linking CCC to downstream response and gene regulatory networks

According to Almet et al. (2021), a limited number of methods consider the connection between CCC and its downstream effects, where these downstream responses represent the cellular processes influenced by the identified signaling pathways. Moreover, the within-cell regulatory networks (GRNs) play a key role in determining what the cell does. This type of inference is usually separated from CCC inference. Hence, incorporating CCC with its downstream response and GRNs is a promising advancement to understand cell type-specific responses and cell signaling ([Almet et al., 2021](#)).

Incorporating trajectory inference with CCC

The way cells communicate with one another can influence how they develop into different types. These changes can be mapped through scRNA-seq with a technique named trajectory inference. Just like GRNs this is usually done independently from CCC inference. Therefore, incorporating both these techniques together could give a better understanding of how signals between cells influence their development over time ([Almet et al., 2021](#)).

Generation of gold-standard datasets

The lack of benchmarks to validate the CCC methodologies is still a significant issue nowadays ([Almet et al., 2021](#)). Tool performance can become greatly improved with the generation of gold-standard datasets or ST, in order to diminish the high false positives rates associated with CCC tools ([Armingol et al., 2024](#)). These datasets are usually of synthetic origin but it's not clear how to best generate synthetic scRNA-seq with a well defined CCC network, as the ground truth is unknown. Moreover, there are already methods available to stimulate synthetic scRNA-seq data but no methods to generate CCC networks with realistic spatial constraints ([Almet et al., 2021](#)).

Data imputation

When studying CCC across different conditions or samples requires matching or aligning single cells from one sample with cells from another which is technically hard since samples do not contain the same amount of cells. This may make some algorithms incompatible and one solution is data imputation - a method to fill in missing or estimated values to make comparisons possible ([Armingol et al., 2024](#)).

Building LRI datasets

LRI databases are essential for studying CCC. These should include ligands, receptors and other elements like protein subunits, activators, inhibitors and competitors that affect signalling. Some of the current databases are incomplete and the different variants of ligands and receptors are not incorporated into them. Thus, the performance of the CCI tools can become negatively impacted ([Armingol et al., 2024](#)).

Chapter 3

Materials and Methods

3.1 Benchmark Datasets

There have been several databases developed thorough the years, ranging from carefully curated pathway databases such as KEGG to databases of experimental interaction evidence curated from literature such as BioGRID or IntAct ([Oughtred et al., 2019](#); [Kanehisa et al., 2023](#); [Del Toro et al., 2022](#)). Moreover, other databases like STRING or GeneMANIA do not only incorporate data from the the previous sources mentioned, as well as specific modes of interactions and an array of computational methods to predict additional associations, providing a more comprehensive overview of the interactome ([Szklarczyk et al., 2024](#); [Warde-Farley et al., 2010](#)). These databases can be used to gain insights into the functional context of proteins or to understand the biological organization behind a whole protein dataset. Therefore, two of the previously mentioned databases were selected as benchmark datasets against the data produced with the use of LLMs, BioGRID and STRING (see Table [3.1](#)).

Annotated corpora such as the event annotation, or so called biological annotation, of GENIA corpus, which has been successfully used by researchers to develop NLP tools was also used as a benchmark dataset to evaluate the quality of data obtained. This type of annotation was performed by biologists with the purpose to identify what kind of biological contexts appears in the text, such as proteins, DNA and RNA ([Kim et al., 2008](#)).

3.1.1 BioGRID

BioGRID is a freely accessible database dedicated to the curation and archival storage of protein, genetic and chemical interactions for humans and all major model organism species including *Escherichia coli* and *Saccharomyces cerevisiae*. All the data is directly downloadable in standard formats and has almost 1.6 millions biological interactions that are backed up by published experimental data. It also includes the

Table 3.1: Summary of benchmark datasets used in this study.

Dataset	Number of Samples	Number of Features	Source / Origin
BioGRID	1 298 433	37	Literature-curated, high-throughput and targeted studies
STRING	13 715 404	3	Genomic context predictions, high-throughput lab experiments, co-expression (conserved), automated text mining and curated interactions from other databases
Genia Corpus	48	3	Annotated PubMed abstracts

curation of post-translational modification and the annotation of chemical interactions between genes/proteins and bioactive small molecules. The data content is updated and freely distributed to the biomedical community with monthly releases, as well as through partnerships with model organism databases such as Saccharomyces Genome Database or WormBase, various meta-databases for interaction data, and general data portals, such as NCBI or UniProt. Thus, this database provides research communities with rigorously annotated resources ([Oughtred et al., 2019](#)).

PPIs were extracted from the BioGRID dataset (version 4.4.247) which was compiled on the 25th of June of 2025 - the most recent date available. The zip archive containing all the interaction and associated annotation data from *Homo sapiens* was downloaded in a tab-delimited format. For detailed column descriptions and sample entries, see Appendix A. Pre-processing and filtering were carried out to retain only Interactor A and Interactor B columns with SWISS-PROT, REFSEQ and TREMBL accessions included as they are provided by BioGRID.

First, non-relevant columns were removed - such as publication metadata, organism identifiers and ontology terms. Swiss-Prot accessions for Interactor A and B were extracted and deduplication was executed. Batches of up to 300 accessions were queried against UniProt REST API to retrieve both recommended and alternative names. Next, RefSeq accessions were extracted through submitted batches of up to 200 entries to the NCBI Entrez E-utilities API. Finally, TrEMBL accession codes were extracted through batches of up to 300 accession codes via UniProt REST API, similarly to Swiss-Prot. The final dataset was saved in comma-separated values (CSV) format for downstream analysis.

3.1.2 STRING

The STRING database scores, compiles and integrates protein-protein association information from experimental assays, prior knowledge and computational predictions. It aims to create comprehensive and objective global networks with physical and functional interactions among proteins for the whole proteomes of thousands of organisms. Moreover, it comes with network clustering and pathway enrichment analysis. This database is known for having an user-friendly interface, a robust scoring system, many sources of evidence and a comprehensive suite of enrichment features (Szklarczyk et al., 2024). For detailed column descriptions and sample entries, see Appendix B.

The combined score is computed by combining the probabilities from the different evidence channels and corrected for the probability of randomly observing an interaction (von Mering et al., 2005).

The PPI data from the STRING database was processed to generate a cleaned and annotated network. First, the interaction dataset was loaded and the columns representing the interacting proteins were renamed to `source` and `target` for clarity, while the confidence score column was removed. All protein identifiers were verified to belong to *Homo sapiens*. Protein metadata was then used to map the original STRING identifiers to human-readable preferred names. Additionally, all known protein aliases were retrieved from the STRING aliases file and appended to each preferred name, ensuring that multiple identifiers for the same protein were captured. Finally, the resulting dataset, with proteins represented by preferred names and associated aliases, was saved as a CSV file for downstream analyses.

3.1.3 GENIA Corpus

The GENIA corpus contains XML files where regulatory relationships and interaction types between biological entities are annotated. The corpus consists of 2,000 MEDLINE abstracts, collected using the three Medical Subject Headings terms *human*, *blood cells* and *transcription factors*. Moreover, it also possesses annotated event data from the GENIA Event Annotation Corpus (version 0.9) with 1,000 Medline abstracts, 9,372 sentences in which 36,114 events were identified (Kim et al., 2008).

The XML files from the Genia Event Annotation Corpus (version 0.9) were used as an additional benchmark dataset. It provides detailed annotations of several entities (e.g., proteins, protein families, protein complexes) and events (e.g., binding, positive regulation, negative regulation). However, the XML files are highly complex and lack a standardized structure for directly extracting PPIs. This made the automatic generation of a PPI benchmark challenging, and a manual approach was required. In this process, only clearly identifiable PPIs were retained, focused on binding and regulatory events.

The final dataset was saved in a CSV format, with each row containing the filename and the two interacting proteins (see Table 3.2). In total, 48 curated interactions were obtained. It's important to note that this benchmark has an important limitation: it only covers PPIs appearing in abstracts rather than full texts; This should be kept in mind when interpreting the model evaluation results.

Table 3.2: GENIA final dataset columns and example entries.

Column Name	Example Entries
filename	1675604, 1765275, 9094628
protein_A	KBF1 factor, OTF-1, Ets
protein_B	"enhancer A, MHC class I gene H-2kb, enhancer of MHC class I gene H-2kb", "Sp1-binding site, Sp1", NFAT

3.2 RAG Pipeline

The RAG pipeline employed in this study was designed to extract PPIs from biomedical literature ¹. As input, the system receives a scientific article together with an user query/prompt. The text is first extracted from the document, divided into smaller chunks and converted into vector embeddings that capture their semantic meaning. The query is embedded in the same vector space, allowing the system to retrieve the most relevant chunks based on cosine similarity. These retrieved passages, combined with the original query, are then passed to the LLM, which generates a textual response describing the PPIs mentioned in the selected literature (see Figure 3.1).

3.2.1 Extraction and Preprocessing of Documents

A random selection of 20 articles from the GENIA Even Annotation Corpus (version 0.9) was made for downstream analysis using LLMs. These articles were downloaded in PDF format and had their text extracted and preprocessed using a custom pipeline implemented in Python.

¹ <https://github.com/mariana-olivertree/master-thesis>

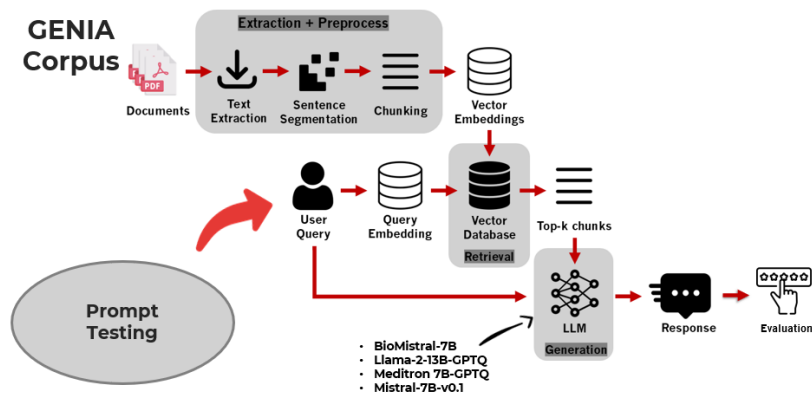


Figure 3.1: Schema of the RAG pipeline applied.

Text Extraction

For the text extraction of the selected articles, the PyMuPDF library was used. This is a high-performance Python library for data extraction, analysis, conversion and manipulation of PDF and other documents (Artifex, 2025). For each article, line breaks were removed and whitespace was normalized to clean the beginning and end of the text.

Pre-Processing

Sentence Segmentation Next, the text was segmented into sentences using the `en_core_sci_sm` model from `scispaCy`, a specialized NLP library for processing biomedical texts which builds on the robust `spaCy` library (Neumann et al., 2019).

Chunking The parsed text was then grouped into fixed-size sentence chunks and consecutive groups of 10 sentences were grouped to form context passages. Additionally, noise reduction through filtering chunk with fewer than 30 tokens and punctuation reduction was executed.

The processing described above was done for all of the 20 PDF files and each resulting chunks were saved in a separate CSV file for each PDF. The final structured output included metadata such as the page number, sentence chunk content, character and word counts and an approximate token count.

3.2.2 Embedding Generation

Each sentence chunk generated was converted into a vector using the Multilingual E5 Large Instruct sentence transformer model, a multilingual instruction-tuned model optimized for semantic similarity tasks across languages. This transformation enables efficient semantic comparison and retrieval operations for

downstream tasks involving LLMs (Wang et al., 2024a). The final embeddings were stored in CSV files.

3.2.3 Retrieval and Generation with LLMs

Prompt Testing

The overall workflow adopted for this procedure is illustrated in Figure 3.2. To identify the most effective prompts for extracting PPIs, a set of 41 candidate prompts was tested using five LLMs available through the Hugging Face Hub². For this evaluation, a single article from the GENIA corpus, entitled "*The Peri-kappa B Site Mediates Human Immunodeficiency Virus Type 2 Enhancer Activation in Monocytes but Not in T Cells*", which described PPIs in its full text, was selected as an input (Clark et al., 1995). This document served as a consistent test case to compare how different prompts influenced the quality of the extracted interactions. The choice of a single representative article ensured that the evaluation focused on prompt design rather than on variability across texts.

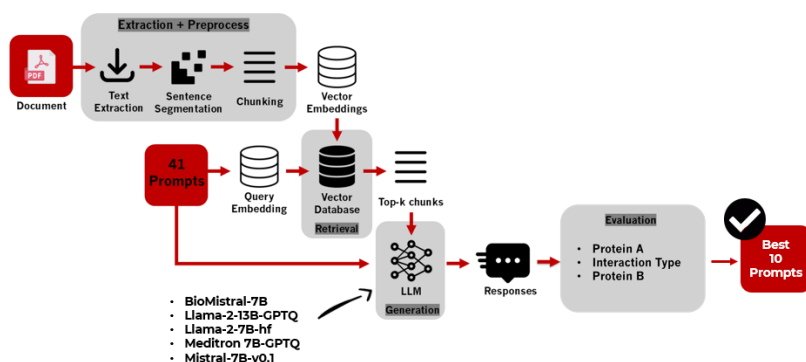


Figure 3.2: Schema of the RAG pipeline applied for prompt testing.

The candidate prompts were evaluated based on their ability to correctly extract three elements: protein A, the interaction type, and protein B. Each element was assessed according to the criteria presented in Table 3.3 and assigned a rating of Good (G), Neutral (N), or Bad (B).

For proteins, a Good (G) rating was assigned when the extraction included either only protein(s) or protein(s) together with the associated article information, reflecting correct and meaningful extraction from the document. A Neutral (N) rating was given when the extraction included either protein(s) together with template information from the prompt or protein(s) together with article information and template information from the prompt, indicating partial correctness but reliance on the prompt rather than the article. A Bad (B) rating was assigned when no protein was extracted or only irrelevant information (e.g., article text or template proteins) was returned.

² <https://huggingface.co/>

For interaction types, a Good (G) rating corresponded to the extraction of only interaction types, which is the most precise and useful output. A Neutral (N) rating reflected extra information included in the extraction which could reduce specificity. A Bad (B) rating was assigned when no interaction type was extracted, rendering the result incomplete.

Prompts were then ranked based on the number of G, N, and B ratings across these three elements. Prompts with the highest number of G's and minimal N's were considered the best-performing. Using this selection criterion, the initial set of 41 prompts was narrowed down to 10 top prompts, which were subsequently used to extract PPIs from the 20 preselected articles. The final set of selected prompts is listed in Appendix C.

Some of the models used for prompt testing have been previously mentioned in Section 2.1.3 - BioMistral-7B and Mistral-7B-v0.1 - as well as quantized versions of earlier introduced models (Section 2.1.3), namely Meditron 7B-GPTQ and LLaMA-2-13B-GPTQ (Labrak et al., 2024a; Jiang et al., 2023; Chen et al., 2023b; Touvron et al., 2023). Additionally, LLaMA-2-7B-hf, another variant from the LLaMA 2 family available in Hugging Face format, was also tested (Touvron et al., 2023). These specific versions were chosen based on availability and hardware compatibility.

All model training and inference tasks were performed on the UGent High-Performance Computing (HPC) cluster,³ which features the following specifications: 2x 24-core AMD EPYC 7413 processors (Milan @ 2.2 GHz) and 4x NVIDIA Ampere A100 GPUs, each with 80GB of GPU memory.

PPI extraction

The computed chunks mentioned in Section 3.2.2 were grouped by the PubMed ID, i.e., by research articles. To extract protein-protein interactions (PPIs), the set of 10 top-performing prompts, selected during prompt testing (see Appendix C for the full list), was used. These 10 prompts were chosen rather than only the single best prompt to increase coverage and robustness, as different prompts can extract complementary interactions from the same text. Figure 3.3 illustrates how the set of prompts is integrated into the RAG pipeline. Each prompt was embedded using the same multilingual sentence transformer model (Multilingual E5 Large Instruct) to ensure consistency with the chunk embedding space.

For each article, cosine similarity was computed between the embedded prompt and the stored chunk embeddings. The top 10 most semantically similar chunks were selected and they provided contextual input to every LLM alongside the original prompt, forming the input to the generation stage. The generated outputs were parsed using regular expressions to extract structured PPI triples in the format:

³ Documentation available at <https://docs.hpc.ugent.be>.

Table 3.3: Evaluation criteria for extracted Protein and Interaction Type outputs.

Category	Description	Evaluation
Protein		
Extracted protein only (P)	Extraction includes only protein(s)	G
Extracted article information + protein (RP)	Extraction includes article information and protein(s)	G
Extracted protein + template (PT)	Extraction includes protein(s) and information from the prompt	N
Extracted article information + protein + template (RPT)	Extraction includes article information, protein(s), and information from the prompt	N
Extracted article information only (R)	Extraction includes only article information	B
Extracted article information + template (RT)	Extraction includes article information and information from the prompt	B
Extracted template only (T)	Extraction includes only information from the prompt	B
Extracted nothing	No extraction	B
Interaction Type		
Extracted interaction types only (IT)	Only interaction types were extracted	G
Extracted more than just interaction types (NIT)	Extraction includes more than just interaction types	N
Extracted no interaction type (N)	No interaction type was extracted	B

Chapter 4

Results and Discussion

4.1 Prompt Testing

The results of prompt testing were evaluated based on three main components: Protein A, Interaction Type and Protein B. This evaluation followed the criteria observed in Table 3.3.

4.1.1 Protein A Outputs

Figure 4.1 shows the distribution of prompt outputs for the Protein A component. Each bar on the x-axis represents one of the possible output categories from Table 3.3, while the color of each bar indicates its quality classification: green for Good (G), yellow for Neutral (N) and red for Bad (B).

The most frequent outputs were classified as **Bad**, followed by **Neutral**, with only a small fraction labeled as **Good**. A large proportion of the predictions consisted only of the examples embedded in the prompts -represented by the tall red bar labeled **T** (Template) - indicating that models often reproduced template content rather than extracting relevant proteins from the article. In contrast, the **P** category (only proteins extracted) corresponds to the most desirable behavior and it appears least frequently.

For clarity, the abbreviations on the x-axis represent combinations of extracted content:

- **P**: only proteins extracted (Good)
- **RP**: article information and proteins (Good)
- **PT**: proteins and template/prompt proteins (Neutral)
- **RPT**: article information, proteins, and template/prompt proteins (Neutral)
- **R**: article information only (Bad)
- **RT**: article and template information (Bad)

- **T**: template/prompt information only (Bad)
- **N**: nothing extracted (Bad)

A big chunk of the outputs generated by the models contained information from the template/prompts proteins. According to Chang et al. (2025) (Chang et al., 2025), the addition of contextual information, while it may enhance the model's ability to infer interactions, it also adds noise which may affect the model's predictions. These resonates with the results obtained from the prompt testing since a big proportion of the output were affected by the noise of the examples added to the prompts.

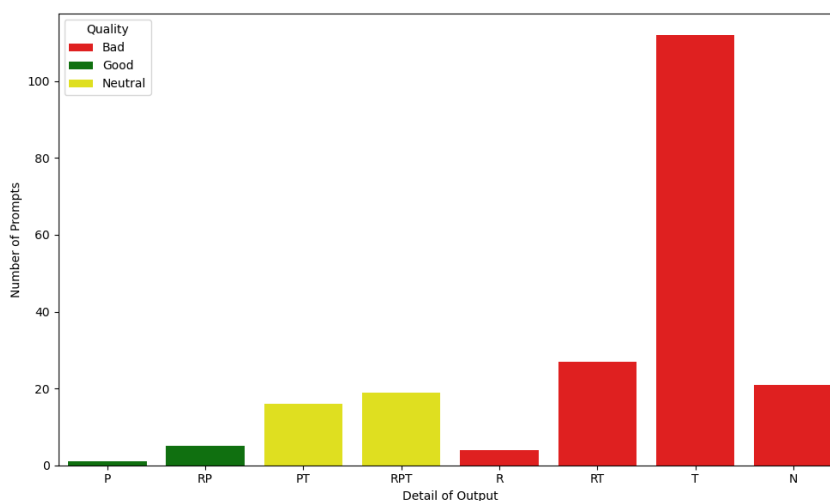


Figure 4.1: Protein A output grouped by quality.

The distribution of the quality of the prompt outputs for the Protein A component by each model is shown in Figure 4.2. All models generated a higher amount of "Bad" outputs compared to "Neutral" and "Good" categories. Furthermore, Meditron generated no "Good" outputs, while BioMistral generated almost equal numbers of "Neutral" and "Bad" outputs. This may suggest that BioMistral generates more desirable results compared to the other models; yet, this observations can only be further validated by evaluating the model performance across the remaining set of PDFs.

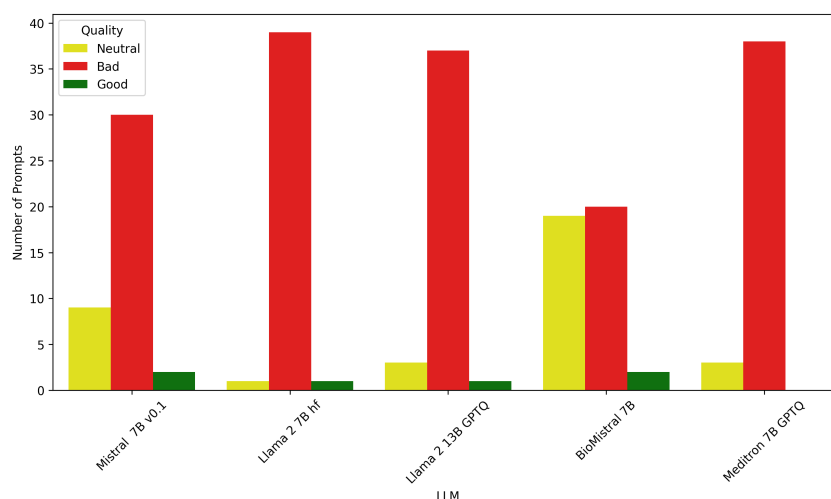


Figure 4.2: Quality of Protein A output per LLM.

4.1.2 Interaction Type Output

The figure below shows the distribution of prompt outputs for the Interaction Type component according to the criteria in Table 3.3. Most outputs were classified as "Good" (green), followed by "Neutral" (yellow) and "Bad" (red) categories (see Figure 4.3).

The labels used in the figure are defined as follows:

- **IT**: extracted interaction types only (Good, G).
- **NIT**: extracted other information in addition to interaction types (Neutral, N).
- **N**: did not extract any interaction type (Bad, B).

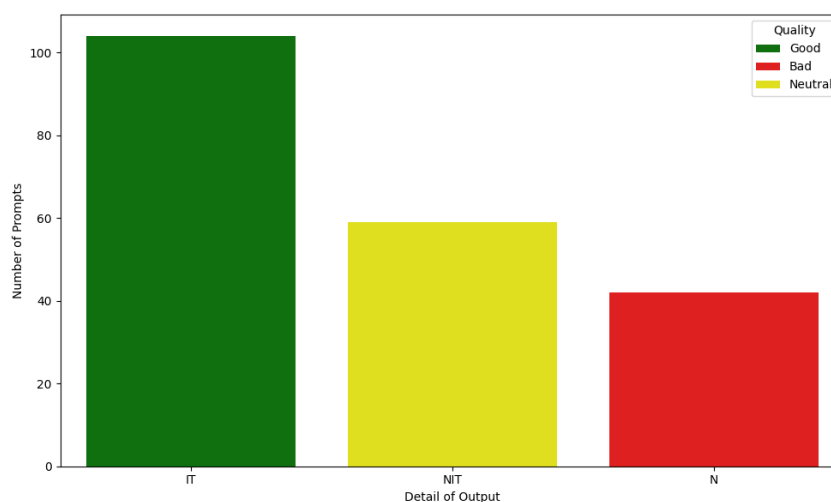


Figure 4.3: Interaction type output grouped by quality.

The distribution of the quality of the prompt outputs for the Interaction Type component by each model is shown in Figure 4.4. All models generated a higher proportion of "Good" outputs. Additionally, BioMistral produced no "Bad" outputs, while the other models produced more "Bad" outputs than "Neutral" ones, except for Meditron. The latter generated more "Neutral" outputs than "Bad". These results suggest, once again, that BioMistral has the ability to generate more desirable outputs.

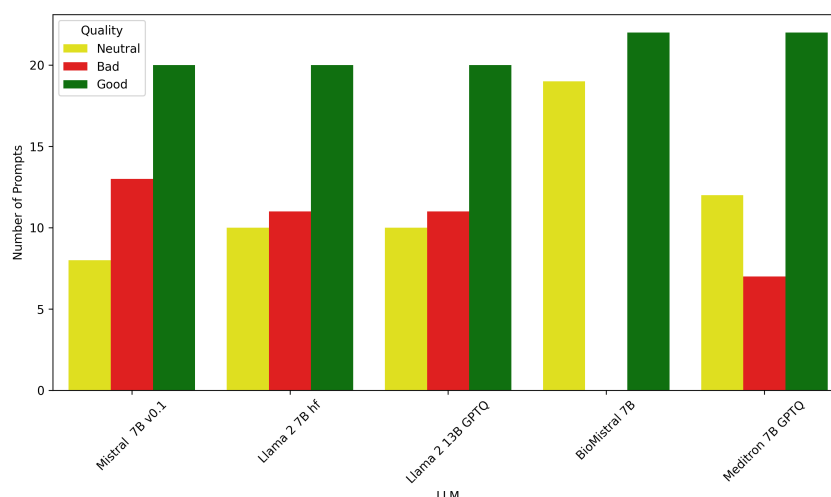


Figure 4.4: Quality of Interaction Type output per LLM.

4.1.3 Protein B Output

Figure 4.5 shows the distribution of prompt outputs for the Protein B component, using the evaluation criteria in Table 3.3. Similar to Protein A (Figure 4.1), most outputs were classified as **Bad**, followed by **Neutral**, with very few labeled as **Good**. The large "T" (Template) bar indicates that many outputs consisted only of the examples embedded in the prompts.

One notable difference from Protein A is that no outputs containing only proteins ("P") were observed for Protein B, suggesting that the models had more difficulty extracting this component independently of the prompt templates.

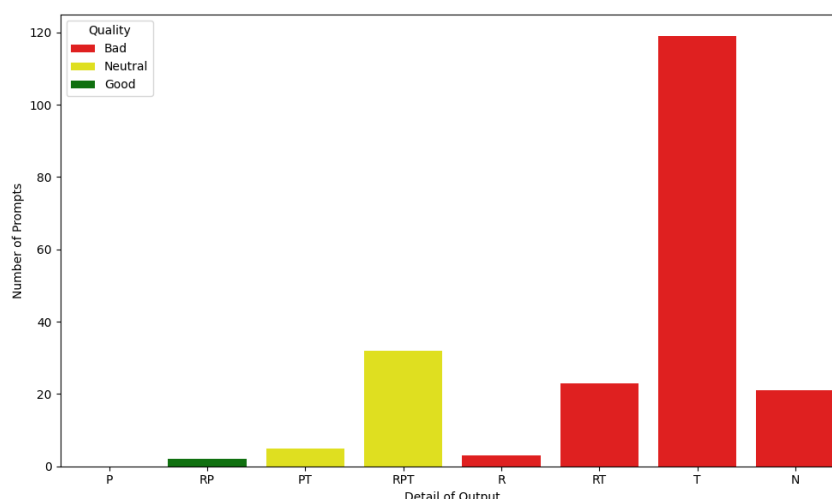


Figure 4.5: Protein B output grouped by quality.

The distribution of the quality of the prompt output for the Protein B component by each model can be seen on Figure 4.6. All models generated a higher proportion of "Bad" outputs, except for Mistral, which generated more "Neutral" outputs. No "Good" outputs were obtained with Llama 2 7B, Llama 2 13B and Meditron. This may suggest that Mistral generates more desirable outputs for the Protein B component.

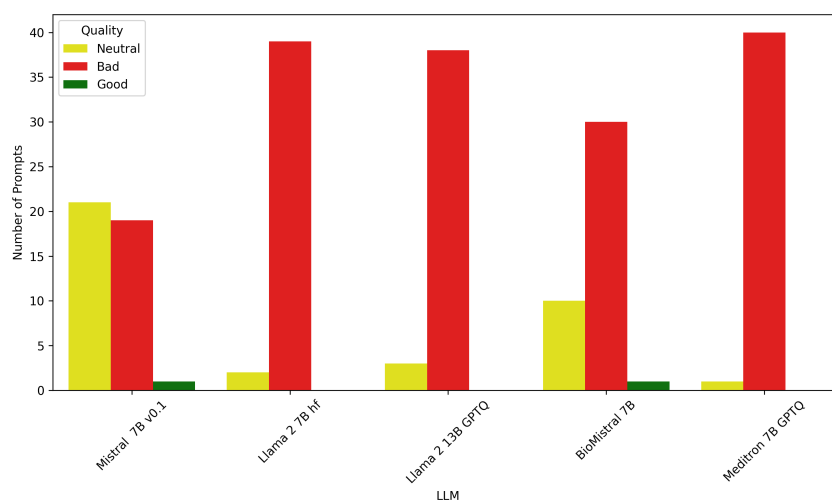


Figure 4.6: Quality of Protein B output per LLM.

4.1.4 Overall performance of LLMs

The overall quality of LLM outputs was quantified using a simple scoring system. For each output type (Protein A, Protein B, and Interaction Type), the model predictions were first categorized as "Good," "Neutral," or "Bad" according to the criteria in Table 3.3. These categories were then mapped to numeric scores: 2 for "Good," 1 for "Neutral," and 0 for "Bad." The average score for each LLM across all evaluated outputs

was computed separately for each component, resulting in the values shown in Figure 4.7.

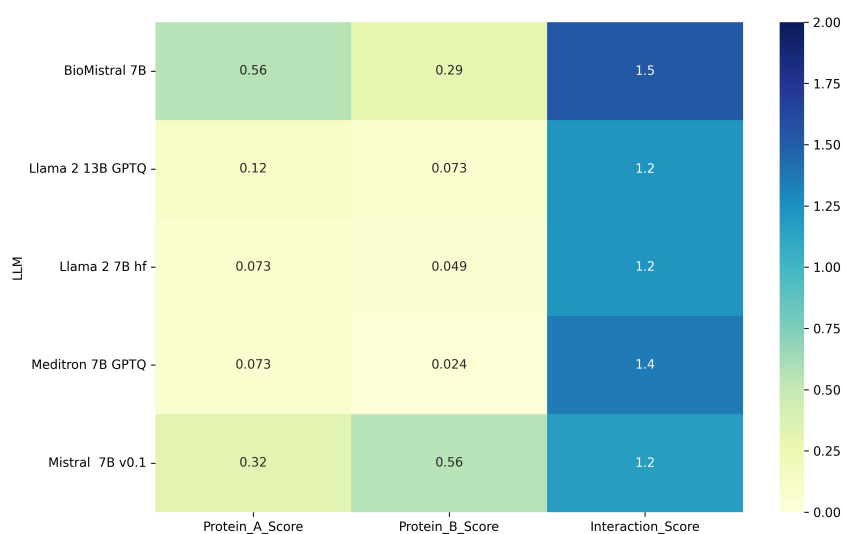


Figure 4.7: Average Quality Score per LLM and Output Type.

According to the figure, BioMistral achieved the highest average score for the Protein A component (0.56), followed by Mistral (0.32). The Llama 2 models and Meditron obtained substantially lower scores (≤ 0.12) for the same component. For Protein B, Mistral achieved the highest score (0.56), with BioMistral second (0.29), while the Llama models and Meditron scored ≤ 0.073 . For the Interaction Type component, BioMistral again had the highest score (1.5), followed by Meditron (1.4), whereas the Llama 2 models and Mistral tied at 1.2 and 1.4, respectively. These results indicate that BioMistral consistently produces higher quality outputs across different components, while performance for the other models varies depending on the output type.

The next figure (Figure 4.8), compares the average performance of the LLMs used for prompt testing across the three components (Protein A, Protein B and Interaction Type).

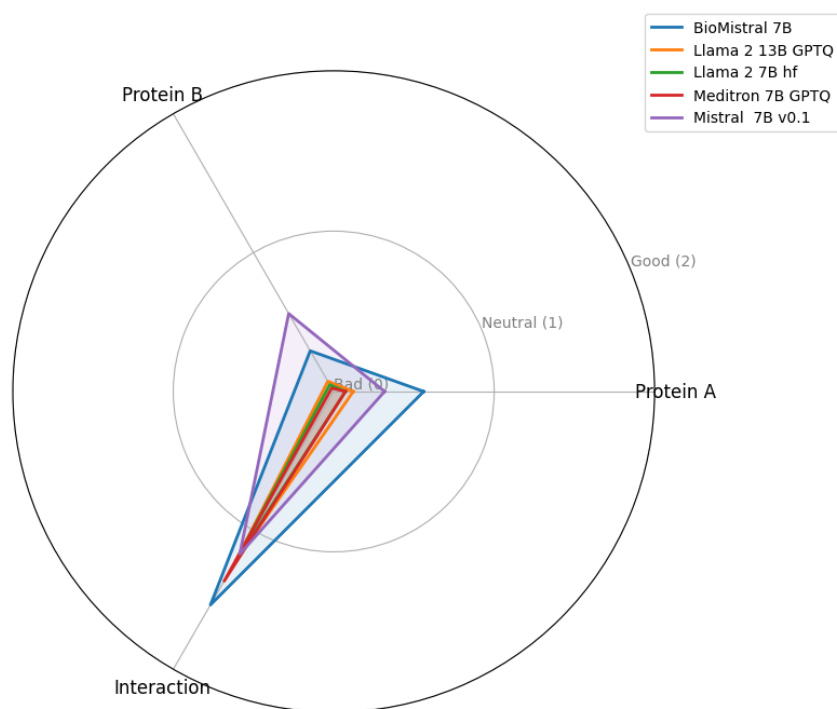


Figure 4.8: LLM Performance Comparison Across Tasks.

BioMistral shows the strongest overall performance: the highest score for Protein A and Interaction Type and the second highest score amongst the rest of the models for Protein B. Mistral performs best at identifying Protein B component but presents a mid-range when it comes to the other components. Meditron and the two Llama 2 variants share a common pattern of relatively low scores on the entity-identification tasks (Protein A and Protein B; values ≤ 0.12) but moderate-to-high Interaction scores (Meditron 1.40; Llama 2 13B and 7B both 1.20). Overall, interaction extraction yields the highest absolute values for all models, whereas entity-level extraction (Protein A and Protein B) shows much larger variability between models.

4.2 PPI Extraction

The 20 preselected GENIA articles were used as input for PPI extraction. Each article was processed through the RAG pipeline described in Section 3.2, which includes chunking, embedding, retrieval of relevant chunks, and generation of PPI predictions by the LLMs. The 10 selected prompts were applied within this pipeline for each tested model. For every model and prompt, the resulting predicted PPIs (protein A, interaction type, protein B) were collected and normalized (lowercased and sorted) to form candidate interaction pairs for evaluation. These predictions were then compared against benchmark datasets to calculate precision, as described in the corresponding Methods subsections for each database.

4.2.1 BioGRID

Precision was computed by comparing each predicted protein–protein pair against the BioGRID benchmark described in Section 3.1.1. As detailed there, the benchmark includes all known interactions and considers all synonym combinations for each protein, such that a predicted pair is counted as a true positive if any of its synonym pairs appear in BioGRID. Pairs not found in the benchmark were counted as false positives. Because it is unknown whether all interactions reported in the 20 GENIA articles are included in BioGRID, the total number of true PPIs is not available, and thus recall and F1-scores could not be calculated. Precision therefore provides an estimate of prediction accuracy relative to this benchmark, reflecting the fraction of predicted interactions that are present in BioGRID.

In terms of precision, Llama 2 13B GPTQ achieved the highest score (0.50), followed by BioMistral 7B (0.45), Meditron 7B GPTQ (0.41), and Mistral 7B v0.1 (0.39) (Table 4.1). These results differ from the outcomes of prompt testing, where Llama 2 13B GPTQ was never the best-performing model. However, the relatively strong performance of BioMistral 7B in both settings is consistent with the trends observed earlier as well as the weaker performance of both Meditron 7B GPTQ and Mistral 7B v0.1.

It was not possible to calculate recall or F1-scores for this evaluation because it is unknown whether all PPIs reported in the 20 GENIA articles are included in BioGRID. Therefore, the total number of true interactions in this subset cannot be determined. Precision was used as the evaluation metric, reflecting the fraction of predicted interactions that are present in BioGRID, and providing an estimate of prediction accuracy relative to this benchmark.

Overall, these findings may suggest that BioMistral demonstrates stable performance across evaluation methods, the performance of Llama 2 13B GPTQ appears to be more task-dependent.

Table 4.1: Performance of LLMs on BioGRID benchmark.

Model	True Positives		False Positives	Total Predictions	Precision
BioMistral 7B	1789	2183		3972	0.4504
Llama2 13B GPTQ	5409	5403		10812	0.5003
Meditron 7B GPTQ	2845	4139		6984	0.4074
Mistral 7B v0.1	6136	9682		15818	0.3879

4.2.2 STRING

Precision was computed against the STRING benchmark in the same manner as for BioGRID (see Section 3.1.2 for benchmark creation). The evaluation showed that LLaMA 2-13B GPTQ achieved the highest precision (0.47), followed by Meditron 7B GPTQ (0.38), BioMistral 7B (0.35), and Mistral 7B v0.1 (0.27). LLaMA 2-13B GPTQ again obtained the leading score, contrasting with the results from prompt testing. BioMistral 7B performed worse compared to both prompt testing and the BioGRID benchmark. Mistral 7B v0.1 showed the lowest precision score. Overall, models performed worse on STRING than on BioGRID: the number of true positives decreased slightly, while false positives increased, leading to a general drop in precision.

As with BioGRID, recall and F1-scores were not calculated due to unknown coverage of true PPIs in the 20 GENIA articles.

Table 4.2: Performance of LLMs on STRING benchmark.

Model	True Positives	False Positives	Total predictions	Precision
BioMistral 7B	1377	2595	3972	0.3467
Llama2 13B GPTQ	5046	5766	10812	0.4667
Meditron 7B GPTQ	2646	4338	6984	0.3789
Mistral 7B v0.1	4273	11545	15818	0.2701

4.2.3 GENIA

The results, shown in Table 4.3, highlight an important contrast with the BioGRID and STRING evaluations. Precision values for all models were very low, with Mistral 7B v0.1 achieving a comparatively higher precision (0.0072), although this score is still near zero. However, when considering recall, Mistral 7B v0.1 continued to show an advantage (0.62), while the other models remained below 0.20. Consequently, Mistral 7B v0.1 also achieved the highest F1 score (0.0142), despite the overall low values across all models.

These findings seem to emphasize the difficulty of PPI extraction at the sentence and document level when evaluated against a curated benchmark such as GENIA. The large number of false positives suggest that the models frequently predicted interactions that were not present in the benchmark, while at the same time missing many true PPIs. However, it's important to note that the GENIA benchmark only

includes PPIs that appear in the abstracts of the articles, while the models were applied to the full text. As a result, many of the interactions perceived as false positives may in fact be true positives that occur elsewhere in the article but fall outside the annotated benchmark. This mismatch between the scope of the benchmark and the model output probably explains the extremely low precision values observed, as the evaluation penalizes valid extractions that are not represented in the abstract-based benchmark.

Table 4.3: Performance of LLMs on GENIA benchmark.

Model	True Positives	False Positives	False Negatives	Total predictions	Precision	Recall	F1 score
BioMistral 7B	14	3958	79	3972	0.0035	0.1505	0.0069
Llama2 13B GPTQ	20	10792	81	10812	0.0018	0.198	0.0037
Meditron 7B GPTQ	8	6976	82	6984	0.0011	0.0889	0.0023
Mistral 7B v0.1	114	15704	71	15818	0.0072	0.6162	0.0142

4.2.4 Overall performance of LLMs

The results obtained across the three benchmarks shows a consistent tendency of LLMs to generate false positive predictions in PPI extraction. Chang et al. (2025) have also reported a consistent bias to false positive predictions in LLM-based approaches, highlighting a more general challenge associated with using current LLM-based approaches (Chang et al., 2025).

A central limitation also lies in the corpora used for benchmarking. For example, the use of the GENIA benchmark has played an important role in biomedical text mining research but it is no longer publicly available and even when access was possible, structural challenges were presented. In particular, the organization of the XML files makes it hard to extract PPI interactions and, therefore making it hard to evaluate results. Alternative corpora such as BioInfer, HPRD50, IEPA and LLL are commonly used but they are all made from abstracts or single sentences and not from full-text articles (Gill et al., 2024; Huang et al., 2024; Doneva et al., 2024).

While LLMs have shown encouraging performance while detecting protein events, its application has been restricted to abstracts and single sentences (Huang et al., 2024). While these demonstrate promising results, it also highlights their dependency on abstract-level datasets which limits their applicability to a broader scope of literature.

Furthermore, the complexity of unstructured biomedical text further enhances the challenging task of extracting PPIs. Unlike structured data with a predefined structure, such as tables and datasets, unstructured data contains abbreviations, it has context-dependent meaning and lacks explicit markers for

relationships ([Gill et al., 2024](#)). Additionally, the lack of a standard annotation across the corpora makes benchmarking of the models difficult and reduces comparability between studies ([Park et al., 2022](#))

Fine-tuning approaches can also help enhance models performance across PPI detections tasks. By transferring general language knowledge into domain-specific contexts, their task performance and accuracy can be improved. Gill et al. (2024) have shown promising results with fine-tuning BioBERT and BERN2 with previously mentioned commonly used PPI databases such as HPRD50 and LLL through a systematic and robust relation extraction framework the researchers built ([Gill et al., 2024](#)). Moreover, the use of alternative methodological approaches might have led to different outcomes. For instance, preprocessing pipelines such as Stanford CoreNLP or Apache OpenNLP for feature extraction, combined with specialized tools for entity recognition like Gimli or NERsuite, could have improved the detection of PPIs. Despite this, event extraction continues to be challenging. While simple events achieve high F1 scores, complex interactions, such as binding or regulatory events, achieve lower levels of accuracy ([Anandaraj and Bali, 2023](#); [Patwardhan et al., 2023](#)).

Chapter 5

Conclusion

This thesis investigated the capabilities of open-source LLMs in extracting PPIs from biomedical literature, employing a RAG pipeline. With prompt testing, BioMistral generally produced the highest quality outputs compared to the other models tested, which exhibited a more variable performance. The benchmarking process against BioGRID, STRING, and GENIA indicated a greater number of false positives and generally, a low precision; implicating that the more favorable results shown through prompt testing do not always transfer to benchmark tasks.

The results of this work also indicated significant limitations of LLMs in performing extraction of a biomedical relationship. Accurate entity recognition and reduced noise in template-based prompts were noted to be challenging, as well as performing extractions to complex interactions, particularly from the broad scope of abstracts to the full-text articles. The absence of full-text benchmarks further complicates the evaluation process and limits validation of the output to be appropriately validated.

This work indicates that models built specifically for a particular domain may outperform models that are produced with more generic purposes and that domain-specific training of LLMs seems critical for successful applications in biomedical applications. The work also shows the need to use prompt engineering and benchmark conscious evaluations in developing the highest performance of LLMs for PPI extractions.

Despite these challenges, LLMs remain a promising tool for biomedical literature mining. Future work should focus on fine-tuning models using full-text domain-specific corpora, the development of context-aware entity recognition strategies and the development of full-text benchmarks that go beyond the abstract and that capture entities and their relationships.

Overall, this thesis demonstrates that LLMs represent a valuable starting point for automated PPI extraction, while also clarifying the methodological challenges that must be overcome for their reliable application in biomedical research. By systematically evaluating open-source models within a RAG pipeline, it highlights the gap between prompt-based performance and benchmark robustness, and underscores the

importance of domain-specific adaptation and rigorous evaluation. More broadly, these findings contribute to the growing evidence that LLMs, when carefully fine-tuned and validated, may have the ability to show promise to accelerate biomedical literature mining, support the expansion of curated interaction databases, and ultimately enhance our understanding of complex molecular interactions.

Bibliography

- Altat Allah Abbassi, Leuson Da Silva, Amin Nikanjam, and Foutse Khomh. Unveiling inefficiencies in llm-generated code: Toward a comprehensive taxonomy, 2025. URL <https://arxiv.org/abs/2503.06327>.
- Marwa Abdulhai, Gregory Serapio-Garcia, Clément Crepy, Daria Valter, John Canny, and Natasha Jaques. Moral foundations of large language models, 2023. URL <https://arxiv.org/abs/2310.15337>.
- Asif Adil, Vijay Kumar, Arif Tasleem Jan, and Mohammed Asger. Single-cell transcriptomics: Current methods and challenges in data acquisition and analysis. *Frontiers in Neuroscience*, 15, 2021. ISSN 1662-453X. doi: 10.3389/fnins.2021.591122. URL <https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2021.591122>.
- Mayank Agarwal, Ayan Goswami, and Priyanka Sharma. Evaluating chatgpt-3.5 and claude-2 in answering and explaining conceptual medical physiology multiple-choice questions. *Cureus*, Sep 2023. doi: 10.7759/cureus.46222.
- Bruce Alberts, Alexander Johnson, Julian Lewis, et al. *Molecular Biology of the Cell*. Garland Science, New York, 4 edition, 2002. Available from: <https://www.ncbi.nlm.nih.gov/books/NBK26822/>.
- Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. The falcon series of open language models, 2023. URL <https://arxiv.org/abs/2311.16867>.
- Axel A Almet, Zixuan Cang, Suoqin Jin, and Qing Nie. The landscape of cell-cell communication through single-cell transcriptomics. *Curr. Opin. Syst. Biol.*, 26:12–23, June 2021.
- Gene Set Enrichment Analysis. Human gene set: Kegg ecm receptor interaction, 2025. URL https://www.gsea-msigdb.org/gsea/msigdb/cards/KEGG_ECM_RECEPTOR_INTERACTION. Accessed: 2025-02-26.

- S. Anandaraj and Manish Bali. Biomolecular event extraction using natural language processing. *International journal of electrical and computer engineering systems*, 14:601–612, 06 2023. doi: 10.32985/ijeces.14.5.12.
- Erick Armingol, Adam Officer, Olivier Harismendy, and Nathan E. Lewis. Deciphering cell-cell interactions and communication from gene expression. *Nature Reviews Genetics*, 22(2):71–88, February 2021. doi: 10.1038/S41576-020-00292-X. URL <https://escholarship.org/content/qt0826r6v0/qt0826r6v0.pdf?t=qmgrp7s>.
- Erick Armingol, Hratch M Baghdassarian, and Nathan E Lewis. The diversification of methods for studying cell-cell interactions and communication. *Nat. Rev. Genet.*, 25(6):381–400, June 2024.
- Artifex. Welcome to pymupdf, 2025. URL <https://pymupdf.readthedocs.io/en/latest/#welcome-to-pymupdf>. Accessed: 2025-07-29.
- Muhammad Azam, Yibo Chen, Micheal Olaolu Arowolo, Haowang Liu, Mihail Popescu, and Dong Xu. A comprehensive evaluation of large language models in mining gene relations and pathway knowledge. *Quant. Biol.*, 12(4):360–374, December 2024.
- Elliot Bolton, Abhinav Venigalla, Michihiro Yasunaga, David Hall, Betty Xiong, Tony Lee, Roxana Daneshjou, Jonathan Frankle, Percy Liang, Michael Carbin, and Christopher D. Manning. Biomedlm: A 2.7b parameter language model trained on biomedical text, 2024. URL <https://arxiv.org/abs/2403.18421>.
- Robin Browaeys, Wouter Saelens, and Yvan Saeys. NicheNet: modeling intercellular communication by linking ligands to target genes. *Nat. Methods*, 17(2):159–162, February 2020.
- Loredana Caruccio, Stefano Cirillo, Giuseppe Polese, Giandomenico Solimando, Shanmugam Sundaramurthy, and Genoveffa Tortora. Claude 2.0 large language model: Tackling a real-world classification problem with a new iterative prompt engineering approach. *Intelligent Systems with Applications*, 21:200336, 2024. ISSN 2667-3053. doi: <https://doi.org/10.1016/j.iswa.2024.200336>. URL <https://www.sciencedirect.com/science/article/pii/S2667305324000127>.
- Yung-Chun Chang, Ming-Siang Huang, Yi-Hsuan Huang, and Yi-Hsuan Lin. The influence of prompt engineering on large language models for protein–protein interaction identification in biomedical literature. *Scientific Reports*, 15(1):15493, 2025. ISSN 2045-2322. doi: 10.1038/s41598-025-99290-4. URL <https://doi.org/10.1038/s41598-025-99290-4>.

- Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. A survey on evaluation of large language models, 2023. URL <https://arxiv.org/abs/2307.03109>.
- Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, Alexandre Sallinen, Alireza Sakhaeirad, Vinitra Swamy, Igor Krawczuk, Deniz Bayazit, Axel Marmet, Syrielle Montariol, Mary-Anne Hartley, Martin Jaggi, and Antoine Bosselut. Meditron-70b: Scaling medical pretraining for large language models, 2023a. URL <https://arxiv.org/abs/2311.16079>.
- Zeming Chen, Alejandro Hernández-Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, Alexandre Sallinen, Alireza Sakhaeirad, Vinitra Swamy, Igor Krawczuk, Deniz Bayazit, Axel Marmet, Syrielle Montariol, Mary-Anne Hartley, Martin Jaggi, and Antoine Bosselut. Meditron-70b: Scaling medical pretraining for large language models, 2023b.
- Changde Cheng, Wenan Chen, Hongjian Jin, and Xiang Chen. A review of single-cell rna-seq annotation, integration, and cell-cell communication. *Cells*, 12(15), 2023. ISSN 2073-4409. doi: 10.3390/cells12151970. URL <https://www.mdpi.com/2073-4409/12/15/1970>.
- K. R. Chowdhary. *Natural Language Processing*, pages 603–649. Springer India, New Delhi, 2020. ISBN 978-81-322-3972-7. doi: 10.1007/978-81-322-3972-7_19. URL https://doi.org/10.1007/978-81-322-3972-7_19.
- N. M. Clark, M. C. Hannibal, and D. M. Markovitz. The peri-kappa b site mediates human immunodeficiency virus type 2 enhancer activation in monocytes but not in t cells. *Journal of Virology*, 69(8):4854–4862, August 1995. ISSN 0022-538X. doi: 10.1128/JVI.69.8.4854-4862.1995. URL <https://doi.org/10.1128/JVI.69.8.4854-4862.1995>.
- Clayton Cohn, Nicole Hutchins, and Gautam Biswas. Towards a formative feedback generation agent: Leveraging a human-in-the-loop, chain-of-thought prompting approach with llms to evaluate formative assessment responses in k-12 science. 2023. URL <https://par.nsf.gov/biblio/10468997>.
- Noemi Del Toro, Anjali Shrivastava, Eliot Ragueneau, Birgit Meldal, Colin Combe, Elisabet Barrera, Livia Perfetto, Karyn How, Prashansa Ratan, Gautam Shirodkar, Odilia Lu, Bálint Mészáros, Xavier

- Watkins, Sangya Pundir, Luana Licata, Marta Iannuccelli, Matteo Pellegrini, Maria Jesus Martin, Simona Panni, Margaret Duesbury, Sylvain D Vallet, Juri Rappsilber, Sylvie Ricard-Blum, Gianni Cesareni, Lukasz Salwinski, Sandra Orchard, Pablo Porras, Kalpana Panneerselvam, and Henning Hermjakob. The intact database: efficient access to fine-grained molecular interaction data. *Nucleic Acids Research*, 50(D1):D648–D653, 2022. ISSN 1362-4962. doi: 10.1093/nar/gkab1006. URL <https://doi.org/10.1093/nar/gkab1006>. PMCID: PMC8728211.
- Daniel Dimitrov, Dénes Türei, Martin Garrido-Rodriguez, Paul L. Burmedi, James S. Nagai, Charlotte Boys, Ricardo O. Ramirez Flores, Hyojin Kim, Bence Szalai, Ivan G. Costa, Alberto Valdeolivas, Aurélien Dugourd, and Julio Saez-Rodriguez. Comparison of methods and resources for cell-cell communication inference from single-cell rna-seq data. *Nature Communications*, 13(1):3224, 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-30755-0. URL <https://doi.org/10.1038/s41467-022-30755-0>.
- Simona Emilova Doneva, Sijing Qin, Beate Sick, Tilia Ellendorff, Jean-Philippe Goldman, Gerold Schneider, and Benjamin Victor Ineichen. Large language models to process, analyze, and synthesize biomedical texts: A scoping review. *Discover Artificial Intelligence*, 4(1), Dec 2024. doi: 10.1007/s44163-024-00197-2.
- Sivan Doveh, Assaf Arbelle, Sivan Harary, Rameswar Panda, Roei Herzig, Eli Schwartz, Donghyun Kim, Raja Giryes, Rogerio Feris, Shimon Ullman, and Leonid Karlinsky. Teaching structured vision & language concepts to vision & language models, 2023. URL <https://arxiv.org/abs/2211.11733>.
- Mirjana Efremova, Miquel Vento-Tormo, Sarah A. Teichmann, and Roser Vento-Tormo. Cellphonedb: Inferring cell–cell communication from combined expression of multi-subunit ligand–receptor complexes. *Nature Protocols*, 15(4):1484–1506, Feb 2020. doi: 10.1038/s41596-020-0292-x.
- Tyna Eloundou, Sam Manning, Pamela Mishkin, and Daniel Rock. Gpts are gpts: An early look at the labor market impact potential of large language models, 2023. URL <https://arxiv.org/abs/2303.10130>.
- Botir Elov, Shahlo Khamroeva, and Zilola Khusainova. The pipeline processing of nlp. *E3S Web of Conferences*, 413, 08 2023. doi: 10.1051/e3sconf/202341303011.
- Mohammad Fraiwan and Natheer Khasawneh. A review of chatgpt applications in education, marketing, software engineering, and healthcare: Benefits, drawbacks, and research directions, 2023. URL <https://arxiv.org/abs/2305.00237>.

- Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey, 2024. URL <https://arxiv.org/abs/2312.10997>.
- GeeksforGeeks. Natural language processing (nlp) pipeline, Apr 2025. URL <https://www.geeksforgeeks.org/natural-language-processing-nlp-pipeline/>.
- Jaskaran Kaur Gill, Madhu Chetty, Suryani Lim, and Jennifer Hallinan. Large language model based framework for automated extraction of genetic interactions from unstructured data. *PLOS ONE*, 19(5), May 2024. doi: 10.1371/journal.pone.0303231.
- Franco Giorgi, Luis Bruni, and Roberto Maggio. Receptor oligomerization as a process modulating cellular semiotics. *Biosemiotics*, 3:157–176, 08 2010. doi: 10.1007/s12304-010-9085-x.
- Diego Guidolin, Manuela Marcoli, Cinzia Tortorella, Guido Maura, and Luigi F. Agnati. Receptor-receptor interactions as a widespread phenomenon: Novel targets for drug development? *Frontiers in Endocrinology*, 10, 2019. ISSN 1664-2392. doi: 10.3389/fendo.2019.00053. URL <https://www.frontiersin.org/journals/endocrinology/articles/10.3389/fendo.2019.00053>.
- Ivan Guryanov, Stefano Fiorucci, and Tatiana Tennikova. Receptor-ligand interactions: Advanced biomedical applications. *Materials Science and Engineering: C*, 68:890–903, 2016. ISSN 0928-4931. doi: <https://doi.org/10.1016/j.msec.2016.07.072>. URL <https://www.sciencedirect.com/science/article/pii/S0928493116307469>.
- Muhammad Usman Hadi, Qasem Al-Tashi, Rizwan Qureshi, Abbas Shah, Amgad Muneer, Muhammad Irfan, Anas Zafar, Muhammad Shaikh, Naveed Akhtar, Jia Wu, and Seyedali Mirjalili. Large language models: A comprehensive survey of its applications, challenges, limitations, and future prospects. *TechRxiv*, 07 2023. doi: 10.36227/techrxiv.23589741.v1.
- Shalini Harkar. Rag, Feb 2025. URL <https://www.ibm.com/think/topics/rag-techniques>.
- Julia Hirschberg and Christopher Manning. Advances in natural language processing. *Science (New York, N.Y.)*, 349:261–266, 07 2015. doi: 10.1126/science.aaa8685.
- Taojun Hu and Xiao-Hua Zhou. Unveiling llm evaluation focused on metrics: Challenges and solutions, 2024. URL <https://arxiv.org/abs/2404.09135>.

- Ming-Siang Huang, Jen-Chieh Han, Pei-Yen Lin, Yu-Ting You, Richard Tzong-Han Tsai, and Wen-Lian Hsu. Surveying biomedical relation extraction: a critical examination of current datasets and the proposal of a new resource. *Briefings in Bioinformatics*, 25(3):bbae132, March 2024. ISSN 1477-4054. doi: 10.1093/bib/bbae132.
- Kwan Yuen lu and Vanessa Wong. Chatgpt by openai: The end of litigation lawyers? *SSRN Electronic Journal*, 01 2023. doi: 10.2139/ssrn.4339839.
- Nilan Jacob. Drug targets: ligand and voltage gated ion channels. *International Journal of Basic & Clinical Pharmacology*, 6:235, 01 2017. doi: 10.18203/2319-2003.ijbcp20170314.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, L  lio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timoth  e Lacroix, and William El Sayed. Mistral 7b, 2023. URL <https://arxiv.org/abs/2310.06825>.
- Suoqin Jin, Christian F. Guerrero-Juarez, Lihua Zhang, Ivan Chang, Raul Ramos, Chen-Hsiang Kuan, Peggy Myung, Maksim V. Plikus, and Qing Nie. Inference and analysis of cell-cell communication using cellchat. *Nature Communications*, 12(1), Feb 2021. doi: 10.1038/s41467-021-21246-9.
- Ulf Johansson, Tuwe L  fstr  m, and Henrik Bostr  m. Calibrating multi-class models. In Lars Carlsson, Zhiyuan Luo, Giovanni Cherubin, and Khuong An Nguyen, editors, *Proceedings of the Tenth Symposium on Conformal and Probabilistic Prediction and Applications*, volume 152 of *Proceedings of Machine Learning Research*, pages 111–130. PMLR, 08–10 Sep 2021. URL <https://proceedings.mlr.press/v152/johansson21a.html>.
- Minoru Kanehisa, Miho Furumichi, Yoko Sato, Masayuki Kawashima, and Mari Ishiguro-Watanabe. Kegg for taxonomy-based analysis of pathways and genomes. *Nucleic Acids Research*, 51(D1):D587–D592, 2023. ISSN 1362-4962. doi: 10.1093/nar/gkac963. URL <https://doi.org/10.1093/nar/gkac963>. PMCID: PMC9825424.
- Enkelejda Kasneci, Kathrin Sessler, Stefan K  chemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan G  nnemann, Eyke H  llermeier, Stephan Krusche, Gitta Kutyiniok, Tilman Michaeli, Claudia Nerdel, J  rgen Pfeffer, Oleksandra Poquet, Michael Sailer, Albrecht Schmidt, Tina Seidel, Matthias Stadler, Jochen Weller, Jochen Kuhn, and Gjergji Kasneci. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103:102274, 2023. ISSN 1041-6080. doi: <https://doi.org/10.1016/j.lindif.2023.102274>.

1016/j.lindif.2023.102274. URL <https://www.sciencedirect.com/science/article/pii/S1041608023000195>.

Jin-Dong Kim, Tomoko Ohta, and Jun'ichi Tsujii. Corpus annotation for mining biomedical events from literature. *BMC Bioinformatics*, 9(1):10, 2008. ISSN 1471-2105. doi: 10.1186/1471-2105-9-10. URL <https://doi.org/10.1186/1471-2105-9-10>.

Nataliya Kosmyna, Eugene Hauptmann, Ye Tong Yuan, Jessica Situ, Xian-Hao Liao, Ashly Vivian Berensnitzky, Iris Braunstein, and Pattie Maes. Your brain on chatgpt: Accumulation of cognitive debt when using an ai assistant for essay writing task, 2025. URL <https://arxiv.org/abs/2506.08872>.

Tiffany H Kung, Morgan Cheatham, Arielle Medenilla, Czarina Sillos, Lorie De Leon, Camille Elepaño, Maria Madriaga, Rimel Aggabao, Giezel Diaz-Candido, James Maningo, and Victor Tseng. Performance of ChatGPT on USMLE: Potential for AI-assisted medical education using large language models. *PLOS Digital Health*, 2(2):e0000198, February 2023. doi: 10.1371/journal.pdig.0000198. URL <https://doi.org/10.1371/journal.pdig.0000198>. eCollection 2023 Feb.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. Biomistral: A collection of open-source pretrained large language models for medical domains, 2024a.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. Biomistral: A collection of open-source pretrained large language models for medical domains, 2024b. URL <https://arxiv.org/abs/2402.10373>.

Richard Lachman and Michael Joffe. *Applications of Artificial Intelligence in Media and Entertainment*, pages 201–220. 01 2021. ISBN 9781799835011. doi: 10.4018/978-1-7998-3499-1.ch012.

Elizabeth D. Liddy. Natural language processing. In *Encyclopedia of Library and Information Science*. Marcel Dekker, Inc., New York, 2nd edition, 2001.

Alejandro Lopez-Lira and Yuehua Tang. Can chatgpt forecast stock price movements? return predictability and large language models, 2024. URL <https://arxiv.org/abs/2304.07619>.

Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2020. URL <https://arxiv.org/abs/1802.03426>.

- Stephen D. Meriney and Erika E. Faselow. Chapter 10 - introduction to receptors. In Stephen D. Meriney and Erika E. Faselow, editors, *Synaptic Transmission*, pages 209–214. Academic Press, 2019a. ISBN 978-0-12-815320-8. doi: <https://doi.org/10.1016/B978-0-12-815320-8.00010-7>. URL <https://www.sciencedirect.com/science/article/pii/B9780128153208000107>.
- Stephen D. Meriney and Erika E. Faselow. Chapter 10 - introduction to receptors. In Stephen D. Meriney and Erika E. Faselow, editors, *Synaptic Transmission*, pages 209–214. Academic Press, 2019b. ISBN 978-0-12-815320-8. doi: <https://doi.org/10.1016/B978-0-12-815320-8.00010-7>. URL <https://www.sciencedirect.com/science/article/pii/B9780128153208000107>.
- Olachi J. Mezu-Ndubuisi and Akhil Maheshwari. The role of integrins in inflammation and angiogenesis. *Pediatric Research*, 89(7):1619–1626, 2021. ISSN 1530-0447. doi: [10.1038/s41390-020-01177-9](https://doi.org/10.1038/s41390-020-01177-9). URL <https://doi.org/10.1038/s41390-020-01177-9>.
- Prakash M. Nadkarni, Lucila Ohno-Machado, and Wendy W. Chapman. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5):544–551, Sep-Oct 2011. ISSN 1067-5027. doi: [10.1136/amiajnl-2011-000464](https://doi.org/10.1136/amiajnl-2011-000464). Research supported by NIH grants U54 HL108460, UL1 RR031980, R01 LM009520.
- Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models, 2024. URL <https://arxiv.org/abs/2307.06435>.
- Fnu Neha, Deepshikha Bhati, and Deepak Kumar Shukla. Retrieval-augmented generation (rag) in health-care: A comprehensive review. *AI*, 6(9), 2025. ISSN 2673-2688. doi: [10.3390/ai6090226](https://doi.org/10.3390/ai6090226). URL <https://www.mdpi.com/2673-2688/6/9/226>.
- Stepan Nersisyan, Victor Novosad, Narek Engibaryan, Yuri Ushkaryov, Sergey Nikulin, and Alexander Tonevitsky. Ecm–receptor regulatory network and its prognostic role in colorectal cancer. *Frontiers in Genetics*, 12, 2021. ISSN 1664-8021. doi: [10.3389/fgene.2021.782699](https://doi.org/10.3389/fgene.2021.782699). URL <https://www.frontiersin.org/journals/genetics/articles/10.3389/fgene.2021.782699>.
- Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. Scispacy: Fast and robust models for biomedical natural language processing. In *Proceedings of the 18th BioNLP Workshop and Shared Task*. Association for Computational Linguistics, 2019. doi: [10.18653/v1/w19-5034](https://doi.org/10.18653/v1/w19-5034). URL <http://dx.doi.org/10.18653/v1/W19-5034>.

Rose Oughtred, Chris Stark, Bobby-Joe Breitkreutz, Jennifer Rust, Lorrie Boucher, Christie Chang, Nadine Kolas, Lara O'Donnell, Genie Leung, Rochelle McAdam, Frederick Zhang, Sonam Dolma, Andrew Willems, Jasmin Coulombe-Huntington, Andrew Chatr-Aryamontri, Kara Dolinski, and Mike Tyers. The biogrid interaction database: 2019 update. *Nucleic Acids Research*, 47(D1):D529–D541, 2019. ISSN 1362-4962. doi: 10.1093/nar/gky1079. URL <https://thebiogrid.org>. Open access database for protein, genetic, and chemical interactions; includes CRISPR screen data via ORCS.

Gilchan Park, Sean McCorkle, Carlos Soto, Ian Blaby, and Shinjae Yoo. Extracting protein-protein interactions (ppis) from biomedical literature using attention-based relational context information. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 2052–2061, 2022. doi: 10.1109/BigData55660.2022.10021099.

Gilchan Park, Byung-Jun Yoon, Xihai Luo, Vanessa Lpez-Marrero, Patrick Johnstone, Shinjae Yoo, and Francis Alexander. Automated extraction of molecular interactions and pathway knowledge using large language model, galactica: Opportunities and challenges. In Dina Demner-fushman, Sophia Ananiadou, and Kevin Cohen, editors, *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 255–264, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.bionlp-1.22. URL <https://aclanthology.org/2023.bionlp-1.22>.

Gilchan Park, Byung-Jun Yoon, Xihai Luo, Vanessa López-Marrero, Shinjae Yoo, and Shantenu Jha. Comparative performance evaluation of large language models for extracting molecular interactions and pathway knowledge. *Journal of Computational Biology*, 2025. doi: 10.1089/cmb.2025.0078. URL <https://doi.org/10.1089/cmb.2025.0078>. Ahead of print.

Nikhil Patwardhan, Stefano Marrone, and Carlo Sansone. Transformers in the real world: A survey on nlp applications. *Information*, 14(4):242, 2023. doi: 10.3390/info14040242. URL <https://doi.org/10.3390/info14040242>.

Ş. Polan. Retrieval-augmented generation: Architecture, techniques, and evaluations. *Journal of Modern Technology and Engineering*, 10(1):42–56, 2025. doi: 10.62476/jmte.10142. URL <https://doi.org/10.62476/jmte.10142>.

Mohaimenul Azam Khan Raiaan, Md. Saddam Hossain Mukta, Kaniz Fatema, Nur Mohammad Fahad, Sadman Sakib, Most Marufatul Jannat Mim, Jubaer Ahmad, Mohammed Eunus Ali, and Sami Azam. A

- review on large language models: Architectures, applications, taxonomies, open issues and challenges. *IEEE Access*, 12:26839–26874, 2024. doi: 10.1109/ACCESS.2024.3365742.
- Hasin Rehana, Nur Bengisu Çam, Mert Basmaci, Jie Zheng, Christianah Jemiyo, Yongqun He, Arzucan Özgür, and Junguk Hur. Evaluating GPT and BERT models for protein-protein interaction identification in biomedical text. *Bioinform. Adv.*, 4(1):vbae133, September 2024.
- Pablo Rivas and Liang Zhao. Marketing with chatgpt: Navigating the ethical terrain of gpt-based chatbot technology. *AI*, 4(2):375–384, 2023. ISSN 2673-2688. doi: 10.3390/ai4020019. URL <https://www.mdpi.com/2673-2688/4/2/19>.
- R. Rosenfeld. Two decades of statistical language modeling: where do we go from here? *Proceedings of the IEEE*, 88(8):1270–1278, 2000. doi: 10.1109/5.880083.
- Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. Code llama: Open foundation models for code, 2024. URL <https://arxiv.org/abs/2308.12950>.
- R. Sangeetha, Karishma S M, Kiran Kumar A, Lordson Gnanadurai A, and Prabanjan R. Intelligent medical literature access for healthcare professionals using nlp and pubmed. In *2024 International Conference on Emerging Research in Computational Science (ICERCS)*, pages 1–7, 2024. doi: 10.1109/ICERCS63125.2024.10895030.
- Sander Schulhoff, Michael Ilie, Nishant Balepur, Konstantine Kahadze, Amanda Liu, Chenglei Si, Yinheng Li, Aayush Gupta, HyoJung Han, Sevien Schulhoff, Pranav Sandeep Dulepet, Saurav Vidyadhara, Dayeon Ki, Sweta Agrawal, Chau Pham, Gerson Kroiz, Feileen Li, Hudson Tao, Ashay Srivastava, Hevander Da Costa, Saloni Gupta, Megan L. Rogers, Inna Goncareenco, Giuseppe Sarli, Igor Galynker, Denis Peskoff, Marine Carpuat, Jules White, Shyamal Anadkat, Alexander Hoyle, and Philip Resnik. The prompt report: A systematic survey of prompt engineering techniques, 2025. URL <https://arxiv.org/abs/2406.06608>.
- Xin Shao, Xiaoyan Lu, Jie Liao, Huajun Chen, and Xiaohui Fan. New avenues for systematically inferring cell-cell communication: through single-cell transcriptomics data. *Protein & Cell*, 11(12):866–880, 05

2020. ISSN 1674-800X. doi: 10.1007/s13238-020-00727-5. URL <https://doi.org/10.1007/s13238-020-00727-5>.

Tushar Soubhari, Sudhansu Nanda, Tariq Ahmad Lone, and P. Beegam. *Digital Hacks, Creativity Shacks, and Academic Menace: The AI Effect*, pages 208–232. 10 2023. ISBN 9798369306505. doi: 10.4018/979-8-3693-0650-5.ch014.

Andreas Stolcke. Srilm - an extensible language modeling toolkit. In *Interspeech*, 2002. URL <https://api.semanticscholar.org/CorpusID:1988103>.

Jimeng Su, Ying Song, Zhipeng Zhu, Xinyue Huang, Jibiao Fan, Jie Qiao, and Fengbiao Mao. Cell–cell communication: new insights and clinical implications. *Signal Transduction and Targeted Therapy*, 9, 08 2024. doi: 10.1038/s41392-024-01888-z. URL <https://www.nature.com/articles/s41392-024-01888-z>.

Damian Szklarczyk, Katerina Nastou, Mikaela Koutrouli, Rebecca Kirsch, Farrokh Mehryary, Radja Hachilif, Dewei Hu, Matteo E Peluso, Qingyao Huang, Tao Fang, Nadezhda T Doncheva, Sampo Pyysalo, Peer Bork, Lars J Jensen, and Christian von Mering. The string database in 2025: protein networks with directionality of regulation. *Nucleic Acids Research*, 53(D1):D730–D737, 11 2024. ISSN 1362-4962. doi: 10.1093/nar/gkae1113. URL <https://doi.org/10.1093/nar/gkae1113>.

Sean Teebagy, Lauren Colwell, Emma Wood, Antonio Yaghy, and Misha Faustina. Improved performance of chatgpt-4 on the okap examination: A comparative study with chatgpt-3.5. *Journal of Academic Ophthalmology*, 15(02), Jul 2023. doi: 10.1055/s-0043-1774399.

The Mosaic Research Team. Introducing mpt-7b: A new standard for open-source, commercially usable llms, 2023. URL <https://www.databricks.com/blog/mpt-7b>.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rishi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan

- Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023. URL <https://arxiv.org/abs/2307.09288>.
- George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R. Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, Yannis Almirantis, John Pavlopoulos, Nicolas Baskiotis, Patrick Gallinari, Thierry Artières, Axel-Cyrille Ngonga Ngomo, Norman Heino, Eric Gaussier, Liliana Barrio-Alvers, Michael Schroeder, Ion Androutsopoulos, and Georgios Paliouras. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC Bioinformatics*, 16(1):138, 2015. ISSN 1471-2105. doi: 10.1186/s12859-015-0564-6. URL <https://doi.org/10.1186/s12859-015-0564-6>.
- Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL <http://jmlr.org/papers/v9/vandermaaten08a.html>.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. URL <https://arxiv.org/abs/1706.03762>.
- Jean-Philippe Villemin, Laia Bassaganyas, Didier Pourquier, Florence Boissière, Simon Cabello-Aguilar, Evelyne Crapez, Rita Tanos, Emmanuel Cornillot, Andrei Turtoi, and Jacques Colinge. Inferring ligand-receptor cellular networks from bulk and spatial transcriptomic datasets with bulksignalr. *Nucleic Acids Research*, 51(10):4726–4744, 05 2023. ISSN 0305-1048. doi: 10.1093/nar/gkad352. URL <https://doi.org/10.1093/nar/gkad352>.
- Christian von Mering, Lars J Jensen, Berend Snel, Sean D Hooper, Markus Krupp, Mathilde Foglierini, Nelly Jouffre, Martijn A Huynen, and Peer Bork. String: known and predicted protein-protein associations, integrated and transferred across organisms. *Nucleic Acids Research*, 33(Database issue):D433–D437, 2005. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gki005. URL <http://string.embl.de/>.
- Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Multilingual e5 text embeddings: A technical report. *arXiv preprint arXiv:2402.05672*, 2024a.
- Xiaohua Wang, Zhenghua Wang, Xuan Gao, Feiran Zhang, Yixin Wu, Zhibo Xu, Tianyuan Shi, Zhengyuan Wang, Shizheng Li, Qi Qian, Ruicheng Yin, Changze Lv, Xiaoqing Zheng, and Xuanjing Huang. Searching for best practices in retrieval-augmented generation, 2024b. URL <https://arxiv.org/abs/2407.01219>.

- Zichong Wang, Zhibo Chu, Thang Viet Doan, Shiwen Ni, Min Yang, and Wenbin Zhang. History, development, and principles of large language models: an introductory survey. *AI and Ethics*, 5(3):1955–1971, 2025. ISSN 2730-5961. doi: 10.1007/s43681-024-00583-7. URL <https://doi.org/10.1007/s43681-024-00583-7>.
- David Warde-Farley, Sylva L Donaldson, Ovi Comes, Khalid Zuberi, Rashad Badrawi, Pauline Chao, Max Franz, Chris Grouios, Farzana Kazi, Christian Tannus Lopes, Anson Maitland, Sara Mostafavi, Jason Montojo, Quentin Shao, George Wright, Gary D Bader, and Quaid Morris. The genemania prediction server: biological network integration for gene prioritization and predicting gene function. *Nucleic Acids Research*, 38(Web Server issue):W214–W220, 2010. ISSN 1362-4962. doi: 10.1093/nar/gkq537. URL <https://doi.org/10.1093/nar/gkq537>. PMCID: PMC2896186.
- Shangyu Wu, Ying Xiong, Yufei Cui, Haolun Wu, Can Chen, Ye Yuan, Lianming Huang, Xue Liu, Tei-Wei Kuo, Nan Guan, and Chun Jason Xue. Retrieval-augmented generation for natural language processing: A survey, 2025. URL <https://arxiv.org/abs/2407.13193>.
- Yue Wu, So Yeon Min, Yonatan Bisk, Ruslan Salakhutdinov, Amos Azaria, Yuanzhi Li, Tom Mitchell, and Shrimai Prabhumoye. Plan, eliminate, and track – language models are good teachers for embodied agents, 2023. URL <https://arxiv.org/abs/2305.02412>.
- Peter Wulff, Marcus Kubsch, and Christina Krist. *Natural Language Processing and Large Language Models*, pages 117–142. Springer Nature Switzerland, Cham, 2025. ISBN 978-3-031-74227-9. doi: 10.1007/978-3-031-74227-9_7. URL https://doi.org/10.1007/978-3-031-74227-9_7.
- Yi Yang, Mark Christopher Siy UY, and Allen Huang. Finbert: A pretrained language model for financial communications, 2020. URL <https://arxiv.org/abs/2006.08097>.
- Adam Zaremba and Ender Demir. Chatgpt: Unlocking the future of nlp in finance. *Modern Finance*, 1(1):93–98, Nov. 2023. doi: 10.61351/mf.v1i1.43. URL <https://mf-journal.com/article/view/43>.
- Qianqian Zhang, Mengdong Chen, and Lianzhong Liu. A review on entity relation extraction. In *2017 Second International Conference on Mechanical, Control and Computer Engineering (ICMCCE)*, pages 178–183, 2017. doi: 10.1109/ICMCCE.2017.14.

Part I

Appendices

Appendix A

BioGRID Dataset Details

Table A.1: BioGRID dataset columns and example entries (Part 1)

Column Name	Example Entries
#BioGRID Interaction ID	103, 107, 183
Entrez Gene Interactor A	6416, 84665, 90
Entrez Gene Interactor B	2318, 88, 2339
BioGRID ID Interactor A	112315, 124185, 106605
BioGRID ID Interactor B	108607, 106603, 108625
Systematic Name Interactor A	RP-547C9.3, GU280_gp01_nsp3 , RP11-531F24.1
Systematic Name Interactor B	RP6-239D12.2 , RP1-290I10.1 , RP3-462O23.2
Official Symbol Interactor A	MAP2K4 , MYPN , ACVR1
Official Symbol Interactor B	FLNC, ACTN2, FNTA
Synonyms Interactor A	CMD1DD CMH22 ..., DCML IMD21 ..., REPA2 RP-A p32 ...
Synonyms Interactor B	ABP-280 ABP280A ..., CMD1AA, FPTA ..., CMD1AA
Experimental System	Two-hybrid, Synthetic Rescue, Negative Genetic
Experimental System Type	physical, genetic, genetic
Author	Marti A (1997), Bang ML (2001), Wang T (1996)
Publication Source	PUBMED:9006895, PUBMED:11309420, PUBMED:8599089
Throughput	Low Throughput, High Throughput, High Throughput Low Throughput
Score	0.952, 0.46675, 0.55625
Modification	-, Phosphorylation, Desumoylation
Qualifications	#LPPI Likely protein-protein interaction, figure 1, Source of MAFK not clear
Tags	-, -, -
Source Database	BIOGRID, BIOGRID, BIOGRID
Organism ID Interactor A	9606, 559292, 10090
Organism ID Interactor B	9606, 10090, 559292
Organism Name Interactor A	Homo sapiens, Saccharomyces cerevisiae (S288c), Mus musculus
Organism Name Interactor B	Rattus norvegicus, Gallus gallus, Drosophila melanogaster
Ontology Term IDs	CHEBI:34159, APO:0000087, BTO:0000007
Ontology Term Names	resistance to chemicals, HEK-293 cell, cellular morphology
Ontology Term Categories	chemical, cell line, tissue
Ontology Term Qualifier IDs	DOID:1749, APO:0000172, CHEBI:15422
Ontology Term Qualifier Names	hepatocellular carcinoma, blood, ATP
Ontology Term Types	partial rescue, wild type, undetermined

Table A.2: BioGRID dataset columns and example entries (Part 2)

Column Name	Example Entries
SWISS-PROT Accessions Interactor A	P45985, Q86TC9, Q04771
TREMBL Accessions Interactor A	-, A0A087WX60, D3DPA4
REFSEQ Accessions Interactor A	NP_003001 NP_001268364 ..., NP_001243197 NP_001243196 ..., NP_001104537 NP_001096
SWISS-PROT Accessions Interactor B	Q14315, P35609, P49354
TREMBL Accessions Interactor B	Q59H94, Q59FD9 F6THM6, -
REFSEQ Accessions Interactor B	NP_001120959 NP_001449 ..., NP_001094 NP_001265272 ..., NP_002018

Appendix B

STRING Dataset Details

Table B.1: STRING dataset columns and example entries (Part 1)

Column Name	Example Entries
#Complex ac	CPX-1, CPX-8, CPX-9
Recommended name	SMAD2-SMAD3-SMAD4 complex, SCF E3 ubiquitin ligase complex, FBXL3 variant
Aliases for complex	SMAD2/SMAD3/SMAD4 transcription factor complex, -, CRL1-FBXL3 complex SCF-FBXL3 complex
Taxonomy identifier	9606
Identifiers (and stoichiometry) of molecules in complex	P84022(1) Q13485(1) Q15796(1), P18846(1) P18848(1), P62877(1) P63208(1) Q13309(1) Q13616(1)
Evidence Code	ECO:0005547(biological system...), ECO:0005544(...), ECO:0005546(...)
Experimental evidence	-, intact:EBI-25772094, intact:EBI-8789396
Go Annotations	GO:0071144(heteromeric SMAD protein complex), GO:0000977 (...), GO:0042752 (...)
Cross references	reactome:R-HSA-9736938(identity), ww-pdb:1LDJ(subset), pubmed:23542741(see-also)
Description	A transcription factor complex which binds (...), E3 ubiquitin ligase which (...), Transcription factor complex that binds the cAMP (...)
Complex properties	The GIRK1-GIRK4, or I(KACh) channel complex (...), The cardiac Troponin complex (...), CDC37 binds its target proteins (...)
Complex assembly	Heterotrimer, Heterodimer, Hetero 25-mer
Ligand	Nppa (ENSG00000175206) promoter, von Willebrand factor (P04275), Fatty acid (CHEBI:35366)

Table B.2: STRING dataset columns and example entries (Part 2)

Column Name	Example Entries
Disease	Alzheimer's disease diabetes, Overexpressed in many solid tumors, Marker of certain types of cancer
Agonist	-, calcium atom (CHEBI:22984) magnesium atom (CHEBI:25107), Gamma-aminobutyric acid (CHEBI:16865)
Antagonist	-, 7SK snRNA (RNAcentral:URS000055E00D), heparin (CHEBI:28304)
Comment	PRPF3 and SNU13 missing from U4, PSI-XML 2.5 interaction id 2593, Interaction not observed in ASCC3 knockout cells
Source	psi-mi:"MI:0469"(IntAct), psi-mi:"MI:2228"(ceitec), psi-mi:"MI:1332"(bhf-ucf)
Expanded participant list	P84022(1) Q13485(1) Q15796(1), P16220(1) P18848(1), P18846(1) P18848(1)

Appendix C

Prompts Selected

Table C.1: Prompts selected for evaluation. G = Good, N = Neutral, B = Bad.

Prompt	Text	Protein A	Interaction Type	Protein B
6	From the context below, extract all protein-protein interactions in the format: Protein A -> [binding/modulation/activation/etc.] -> Protein B. Only include clear interactions from the main body of the document.	G	G	N
7	You are a biomedical research assistant specializing in extracting protein-protein interactions from scientific articles. Only extract information directly stated in the text. Do not invent interactions. Output in the format: Protein [name] -> [Interaction type] -> Protein [name] Evidence: [direct quotation]	G	N	N
10	Extract protein-protein interactions from biomedical literature. Assign a confidence score (High, Medium, Low) for each interaction, and include supporting evidence.	G	N	G
22	Extract all protein-protein interactions from the context below in the format 'Protein A -> [Interaction Type (e.g., 'binds to', 'activates', 'forms complex with')) -> Protein B'. Only include clear interactions from the main body of the document. Example input: "AKT1 activates mTOR and binds to GSK3B." Example output: AKT1 -> activates -> mTOR AKT1 -> binds to -> GSK3B Process the following context:	N	N	G
25	From the context below, extract all protein-protein interactions in the format: 'Protein A -> [Interaction Type (e.g., 'binds to', 'activates', 'forms complex with')) -> Protein B'. Only include clear interactions from the main body of the document.	N	N	G
26	From the context below, extract all protein-protein interactions in the format: 'Protein A -> [binding/modulation/activation/etc.] -> Protein B'. Only include clear interactions from the main body of the document.	G	G	G

Table C.2: Prompts selected for evaluation. G = Good, N = Neutral, B = Bad.

Prompt	Text	Protein A	Interaction Type	Protein B
29	List all protein-protein relations in the following form: 'Protein A -> [Interaction Type (e.g., 'binds to', 'activates', 'forms complex with')) -> Protein B'. Only include clear interactions from the main body of the document.	N	N	G
30	From the context below, extract all protein-protein interactions in the format: 'Protein A -> [Interaction Type (e.g., 'binds to', 'activates', 'forms complex with')) -> Protein B'. Only include clear interactions from the main body of the document.	N	N	G
32	From the context below, extract all protein-protein interactions in the format: 'Protein A -> [binding/modulation/activation/etc.] -> Protein B'. Only include clear interactions from the main body of the document. Example: Input: "AKT1 activates mTOR and binds to GSK3B." Output: AKT1 -> activation -> mTOR AKT1 -> binding -> GSK3B – Now process the following context:	N	N	G
36	You are an expert biomedical information extractor. Task: Extract all protein-protein interactions from the text below in the format: 'Protein A -> [binding/modulation/activation/etc.] -> Protein B' Only include clear interactions from the main body of the document. Note: The following example is just for your understanding. Do not include this example or any references to it in your output. Example (DO NOT INCLUDE THIS IN YOUR OUTPUT): Input: "AKT1 activates mTOR and binds to GSK3B." Expected Output: AKT1 -> activation -> mTOR AKT1 -> binding -> GSK3B – Now, extract interactions from this context:	N	N	G