Iman Khamis

20 Fundamentals of Artificial Intelligence for Libraries

Abstract: Artificial intelligence (AI) developments are impacting on business and libraries in their operations and service delivery. There are many facets of AI including machine learning (ML), recommender systems, natural language processing (NLP), speech recognition and text analytics. Speech recognition technology is relevant to many businesses and to libraries. It provides customer service benefits and enriches the customer experience. The chapter explores ML, recommender systems, and NLP. Chatbots have emerged from NLP developments and allow uninterrupted interactions between organisations and their clients and ideally enhance the overall user experience. Chatbots offer libraries the opportunity to improve user engagement and operational efficiency, and potentially to reduce costs. This chapter discusses the fundamentals of AI technologies using the development of a chatbot as the context.

Keywords: Chatbots; Artificial intelligence - Library applications; Machine learning

Introduction

Machine learning (ML) refers to the development of algorithms that enable machines to mimic human intelligence and has led to the development of many artificial intelligence (AI) applications. Natural language processing (NLP) helps humans to communicate with computers that can read, hear and understand. Chatbots are software applications or web interfaces which simulate human conversation and facilitate communication between a person and a machine through text or voice interactions. Chatbots are growing in importance as tools in business and in libraries because of their capacity to offer effective and individualised support to customers. Chatbots are likely to become a popular method for addressing the demands of library users as a result of the development of digital technologies and the rising demand for remote services. Chatbots can assist with a variety of requests and tasks by leveraging AI and NLP applications. Chatbots can help in many areas ranging from responding to straightforward questions about library hours and services to helping with more complicated research requests for detailed information on specific topics. Additionally, chatbots can run continuously, allowing libraries to offer users round-the-clock support. In general, chatbots have proven to be a price-

https://doi.org/10.1515/9783111336435-021

less tool for libraries looking to improve user experience and their digital products and offerings.

Building a chatbot can seem to be a difficult task. However, with the right tools and appropriate knowledge, the construction can be simplified. Chatbots have the capacity to improve users' experiences and have emerged as a popular solution for organisations looking for seamless customer service. Libraries are no exception to the trend. In this chapter, the steps needed for building a chatbot for a library are outlined, from understanding ML to selecting the appropriate tools and programming languages. Overall, this chapter provides insight into the fundamental knowledge and resources needed to create a successful chatbot for a library.

Basic background information on the concepts involved in understanding ML and NLP is provided. Guidance is then given on how to construct a chatbot using an intent JavaScript Object Notation (JSON) file. Intent is a messaging object for requesting an action from another app component and facilitates communication. JSON is an open standard file and data interchange format. The intent file will document and answer the questions users have in mind. It ensures that the chatbot will work properly and that the chatbot has the ability to analyse intent and provide a successful interaction.

Machine Learning

Machine learning (ML) refers to the development of algorithms that enable machines to mimic human intelligence. Machine learning technology has recently been used in many fields such as image recognition, biomedical applications, natural language processing, and prediction. In 1958, Rosenblatt developed the first neural network that mimicked neural cells in the human brain (Rosenblatt 1958). In 1975 another breakthrough was made by Werbos when he came up with back-propagation making neural networks and ML more efficient (IBM Developer 2023). He developed the multilayer perceptron (MLP). MLP is a neural network that consists of fully connected layers that can produce a set of outputs from inputs (Werbos 1994). In 1986, Quinlan developed another ML technology known as decision trees (Quinlan 1986), followed by Cortes and Vapnik's invention of support vector machines (SVMs) (Cortes and Vapnik 1995). ML technology has grown exponentially through the development of many accompanying algorithms, inventions, and models such as Adaboost, random forests, and multilayer perceptron.

Arthur Samuel was the first to refer to machine learning in 1959 as a computer's capacity to be programmed and learn on its own:

> The studies reported here have been concerned with the programming of a digital computer to behave in a way which. if done by human beings or animals, would be described as involving the process of learning. While this is not the place to dwell on the importance of machine-learning procedures. or to discourse on the philosophical aspects, there is obviously a very large amount of work. now done by people. which is quite trivial in its demands on the intellect but does, nevertheless, involve some learning. We have at our command computers with adequate data-handling ability and with sufficient, computational spend to make use of machine-learning techniques. hut our knowledge of the basic principles of these techniques is still rudimentary. Lacking such knowledge, it is necessary to specify methods of problem solution in minutes and exact detail a time-consuming and costly procedure. Programming computers to learn from experience should eventually eliminate the need for much of this detailed programming effort. (Samuel 1959, 535).

Bernard in writing for the World Economic Forum on ML referred to the work of Samuel: "In 1959, MIT engineer Arthur Samuel described machine learning as a "Field of study that gives computers the ability to learn without being explicitly programmed", and added "Machine learning is all about sorting through those troves of collected information to discern patterns and predict new ones" (Bernard 2017). ML can also be explained as:

> **Definition**: A computer program is said to learn from experience *E* with respect to some class of tasks *T* and performance measure *P*, if its performance at tasks in *T*, as measured by *P*, improves with experience *E*. (Mitchell 1997, 2)

Another definition is "Programming computers to optimize a performance criterion using example data or experience" (Eick 2024). All the various definitions have a commonality; machine learning is about how computers are taught to perform advanced tasks by inputting sufficient data for effective learning.

Recommender Systems

Machine learning techniques are classified into supervised and machine unsupervised learning techniques. Supervised ML refers to ML algorithms that perform accurate predictions based on input-out pairs. Recommender systems work on the same concept. The most important feature of supervised learning methods is that they are adaptable to various regression tasks. Regression analysis is the most used technique in recommender systems because it aims to organise the content and order it according to a ranking system. In recent years, the need for recommender systems has increased. With vast amounts of information available on the internet, selection of appropriate and accurate information has become more difficult. Making the right choice from the plethora available has led to confusion and

an inability to decide on what is best. There is an increasing need for tools that can effectively prioritise the available options according to specific criteria that will match user needs. Recommender systems were born to solve the problem by searching the vast majority of options available for users, filtering the content, and dynamically generating personalised options for each individual (Isinkaye, Folajimi, and Ojokoh 2015). Recommender systems have been around for a long time, and it is no exaggeration to suggest that the ideas about targeted responses to inquiries emerged initially with the invention of the computer and even perhaps with Alan Turing's question in 1950 which has become known as the Turing Test: "Can machines think?"

In 1979, Elaine Rich addressed the problems related to computers treating their users in a personalised way and described a system called Grundy which simulated the behaviour of a successful librarian and dealt with enquiries from stereotyped models of users constructed on the basis of a small amount of knowledge about them. An analysis of Grundy's performance recommending novels to users demonstrated the effectiveness of the user models in guiding its performance (Rich 1979). Grundy might well be a first in the field of recommender systems (Rich 1979). In the 1990s, GroupLens Research, a human-computer interaction laboratory at the University of Minnesota created the GroupLens recommender, which set up an automatic collaborative filtering system collating user ratings of articles from *Usenet News* (Konstan et al. 1998). A recommender system, MovieLens was set up to collate ratings from users on movies and to provide recommendations to users of the system based on user profiles (Konstan et al. 998).

Perhaps the best-known recommender system is the one used by Amazon who created its famous collaborative filtering method initially for book selection. The filter system created by Amazon engineers was a user-based collaborative system that analysed users based on their purchase history and browsing behaviour, established groups, and developed a recommender system which suggested purchases based on the search histories of the users themselves and other similar users (Ekstrand, Riedl, and Konstan 2011). Collaborative filtering has gained prominence since the success of Amazon's approach changing customer behaviour with personalised experiences particularly in the e-commerce field. User-based collaborative filtering to identify users with similar tastes and preferences based on their historical behaviour has become a standard approach and interest in recommender systems has increased driven by the demands of e-commerce and online shopping.

Schafer, Konstan, and Riedl (1999) highlighted the importance of recommender systems in e-commerce as they made the user experience easier, increased sales of products across the store, and enhanced customer loyalty. Browsers became buyers and purchased additional products. In e-commerce, the essential role of a recommender system is to automatically generate the right suggestions for the

right users based on users' interests and search history to make the experience of shopping more pleasant. However, recommender systems require a large amount of data and learning time to train the models to understand user interests. Recommender systems are considered very powerful tools as they navigate millions of items to select only the item/s which match individual needs to suggest for each user. However, each system is evaluated based on the user's interaction with the suggested recommendation and feedback is provided to the system to retrain the model for more accurate suggestions in the future where matching has been unsatisfactory (Singh 2022).

Research papers have been published on the advantages and difficulties of recommender systems. Some of the issues mentioned that need to be addressed are:

- With the rapid increase in the amount of data produced by users on the Internet, scalability has become a problem and systems that can process large-scale datasets with high accuracy and steady performance need to be built (Ricci, Rokach, and Shapira 2011)
- Computers sometimes find it difficult to comprehend Internet users. A user might choose an item that is out of her/his interest. Many users are not willing to fill out questionnaires. These factors make it difficult for computers to classify user choices correctly. The models must learn to understand the implicit and explicit preferences of users to be able to recommend items that might be of interest
- Privacy is an extremely sensitive matter when gathering user data, and maintaining confidentiality and security of data has become a problematic yet essential feature of recommender systems, and
- Systems might fail to make suggestions and limit or restrict users' choices, preventing them from exploring the variety of items available. Diversity of offerings must be part of the system to ensure the recommender system remains a help rather than a hindrance (Qomariyah 2018).

Natural Language Processing

Natural language processing is one of the core pillars of the AI and data science field.

> Natural language processing, or NLP, combines computational linguistics—rule-based modeling of human language—with statistical and machine learning models to enable computers and digital devices to recognize, understand and generate text and speech (IBM n.d.a)

NLP uses different methods and algorithms to enable successful communication between computers and humans. NLP is where computer science, linguistics, and mathematics come together to convert the natural language of humans to commands that can be understood, executed and regenerated by a machine.

NLP comprises two primary research areas natural language understanding (NLU) and natural language generation (NLG). NLU emphasises making human natural language comprehensible and understandable to a computer by extracting valuable information from text that is said or written. NLG is the opposite where the computer generates a natural language that is understandable by humans from some underlying representation of information or unstructured data (Kang et al. 2020; McDonald 2010; Schank1972). NLP focuses primarily on classification problems rather than regression and uses classification algorithms, such as random forest classification or support vector machines (SVMs). The first step in this process is to represent the data numerically through techniques such as vectorisation (Science Direct 2024), embedding (Barnard 2023), bagging, (IBM n.d.b), or the bag-of-words model. Some of these approaches are explored further in the next section of this chapter.

The purpose of language models is to understand natural language and to predict the probability of the occurrence of a sentence or the next word. Language models can also be used for text generation, which is particularly useful in translation applications. Various techniques, such as counting the frequency of words, and recurrent neural networks (RNN) including long short-term memory (LSTM), can be employed for this purpose. To understand natural language, knowledge graphs are extracted, and inductive and deductive reasoning applied. Deductive reasoning helps the model understand grammar and language rules. It is crucial to comprehend the differences between taxonomy, ontology, and graphs. Understanding the differences between the three will help in data organisation, and information retrieval. Taxonomy provides a hierarchical classification system, while ontology adds meanings and relationships between concepts; finally graphs represent complex relationships between entities. This knowledge is essential for structured data management, accurate data analysis, and facilitating interoperability.

Taxonomy involves naming concepts or entities and organising them based on shared characteristics and can take various structural forms, including hierarchies, and is primarily focused on classification. Ontology involves defining the properties, relationships, and classes used to describe concepts or entities in a specific domain and captures the underlying structure, semantics, and complex relationships within the domain. Graphs are a data structure used to represent relationships between entities or concepts. While they can be used to represent hierarchical structures, they are versatile and can depict various types of connections and structures beyond hierarchies.

Before constructing knowledge graphs, the information must first be extracted from unstructured data. This involves three steps: named entity recognition (NER), named entity linking (NEL), and relation extraction. NER involves labelling words into predefined categories, such as places or objects. NEL links entities to real-world identities, while relation extraction involves extracting knowledge from unstructured data. A knowledge graph is a knowledge base or set of sentences that uses a structured model to store descriptions of entities and their relationships or underlying semantics. A triple is a sequence of three entities, subject, predicate, and object, that codifies a statement. The triples can be stored in a relation description format (RDF), extensible markup language (XML), or graph format. More detail is now provided on some of the topics mentioned in relation to NLP: vectorisation, clustering and support vector machines.

Vectorisation

Vectorisation is the process of converting an algorithm from operating on a single value at a time to operating on a set of values at one time. Term Frequency-Inverse Document Frequency (TF-IDF) assigns a value to each term in a document based on its frequency and inverse document frequency, indicating the importance of the term in the document and its relationship to the collection of documents, bearing in mind that some words appear more frequently in general. The TF-IDF matrix is a numerical measure that indicates the significance of a word in each corpus. Compared to simply counting the frequency of a word, TF-IDF focuses on how often a term is mentioned in the document.

To conduct a more comprehensive analysis and reduce the input dimensions, less significant words can be eliminated in future steps by plotting the components of the TF-IDF matrix and k-means clusters on a two-dimensional plane. Doc2vec is used to create a model that converts groups of words into a single unit for vectorisation. Numeric representations are necessary for machine analysis and TF-IDF or Doc2vec matrices can be used to represent any text document numerically. To avoid issues with word similarity based on frequency, latent semantic analysis (LSA) can be used to analyse the corpus of the document and simulate the meaning of words and passages based on topic similarity rather than frequency. Additionally, latent Dirichlet allocation (LDA) can be used to extract topics from the documents and map them to their respective documents as a probabilistic distribution.

As already mentioned, K-means clustering can be used to reduce the input dimensions. K-means clustering can also be used to discover patterns within the data, to group data points together and to identify underlying structures in the dataset. K-means establishes the centroids for each cluster and assigns the data

points to the nearest cluster. By iterating and recalculating the centroids, the algorithm accurately clusters the points in a specific group. K-means clustering is capable of accurately clustering 250 documents in a thousand-dimensional space within minutes.

Support Vector Machines (SVM)

The support vector machines (SVM) classification technique is highly robust due to its use of the kernel method to map data from lower to higher dimensions, enabling the identification of a decision boundary. The approach maximises the width of the decision boundary, ensuring effective discrimination between positive and negative instances. SVM is not only a linear classification model but can also be used for regression problems. To separate data into classes, SVM creates a hyperplane, and the SVC (Support Vector Classifier) function, which is the classification variant of the SVM algorithm, is utilised to fit the data in the TF-IDF matrix and train the corpus of data with TF-IDF-weighted word frequencies. The function returns the best-fit hyperplane that divides or organises the data.

Building a Chatbot

To build a chatbot, there needs to be a bank of answers which can be provided to the questions expected. The chatbot will classify and map questions into a group of appropriate responses. The intent JSON file contains a variety of questions that library users might ask in a typical customer service situation and will contribute to developing appropriate answers to the questions users have in mind. The questions can be mapped against a group of appropriate responses. The tag on each dictionary in the file indicates the group to which a customer's message would belong. The neural network will be trained to analyse the words in a sentence and classify it to one of the tags in the JSON file. The chatbot will be able to take a response from the groups and display the correct answer to the customer.

The greater the number of tags, responses, and patterns that can be provided to the neural network, the better the chatbot answers will be. Neural networks can be edited, layers added, hyperparameters changed and the network edited to achieve better performance. Convolutional neural networks (CNN) (IBM n.d.c) or recurrent neural networks (RNN) (IBM n.d.d) are recommended for building chatbots locally. And there needs to be lots of data. Larger quantities of data will improve the learning and the outputs. There are various frameworks that can be used: Keras,

PyTorch, TensorFlow, and Apache Spark. Particular frameworks will suit particular situations.

Conclusion

No doubt there will be improvements in NLP approaches, frameworks and techniques that can be applied to the development of chatbot services. Developments in areas like named entity recognition will add more features to the capacity of chatbots. Sentiment analysis could be used to determine whether data is positive, negative or neutral which would help in understanding client needs and could also be used to encourage emotional engagement with a chatbot (Amazon Web Services n.d.). Much more data must be added to the neural networks underpinning the chatbots. Advances in ML, recommender systems, NLP, and associated AI applications have led to significant growth in the use of chatbots in libraries which are capable of providing personalised services to users in multiple languages. "Chatbots are increasingly replacing some of the traditional library services provided by humans, and their role in libraries is expanding rapidly with the evolution of artificial intelligence" (Aboelmaged et al. 2024). Libraries must be openly innovative to capitalise on the opportunities available and ensure that staff are led from an awareness of the technologies to adoption of them and the development of personalised online services for the benefit of library users.

References

Aboelmaged, Mohamed, Shaker Bani-Melhem, Mohd Ahmad Al-Hawari, and Ifzal Ahmad. 2024. "Conversational AI Chatbots in Library Research: An Integrative Review and Future Research Agenda." *Journal of Librarianship and Information Science*. First published online February 6, 2024. https://doi.org/10.1177/09610006231224440

Amazon Web Services. n.d. "What is Sentiment Analysis?" https://aws.amazon.com/what-is/sentiment-analysis/#:~:text=Sentiment%20analysis%20is%20the%20process,social%20media%20comments%2C%20and%20reviews.

Barnard, Joel. 2023. "What is Embedding?" *IBM [Blog]* December 22, 2023. https://www.ibm.com/topics/embedding.

Bernard, Zoë. 2017." Here's What Machine Learning Actually Is." *World Economic Forum*. *Artificial Intelligence*, November 28, 2017. https://www.weforum.org/agenda/2017/11/heres-what-machine-learning-actually-is/.

Cortes, Corinna, and Vladimir Vapnik. 1995. "Support-vector Networks." *Machine Learning* 20: 273-297. http://dx.doi.org/10.1007/BF00994018.

Eick, Christoph F. 2024. "A Gentle Introduction to Machine Learning." [PowerPoint Presentation]. University of Houston. https://www2.cs.uh.edu/~ceick/ai/4368-ML-Intro.pptx.

Ekstrand, Michael D., John T. Riedl, and Joseph A. Konstan. 2011. "Collaborative Filtering Recommender Systems." *Foundations and Trends® in Human–Computer Interaction* 4: 81-173. http://dx.doi.org/10.1561/1100000009.

IBM. n.d.a. "What Is NLP?" https://www.ibm.com/topics/natural-language-processing.

IBM. n.d.b. "What is Bagging?" https://www.ibm.com/topics/bagging.

IBM. n.d.c. "What Are Convolutional Neural Networks?" https://www.ibm.com/topics/convolutional-neural-networks.

IBM. n.d.d. "What are Recurrent Neural Networks?" https://www.ibm.com/topics/recurrent-neural-networks.

Isinkaye, Folasade Olubusola, Yetunde Oluwatoyin Folajimi, and Bolanle Adefowoke Ojokoh. 2015." Recommendation Systems: Principles, Methods, and Evaluation." *Egyptian Informatics Journal* 16: 261-273. https://doi.org/10.1016/j.eij.2015.06.005.

Jones, M. Tim. 2017. "Cognitive Neural Networks: A Deep Dive." *IBM Developer*. July 24, 2017. https://developer.ibm.com/articles/cc-cognitive-neural-networks-deep-dive/.

Kang, Yue, Zhao Cai, Chee-Wee Tan, Qian Huang, and Hefu Liu. 2020. "Natural Language Processing (NLP) in Management Research: A Literature Review." *Journal of Management Analytics* 7, no. 2: 139–172. https://doi.org/10.1080/23270012.2020.1756939.

Konstan, Joseph A., John Riedl, AI Borchers, and Jonathan L. Herlocker. 1998. "Recommender Systems: A GroupLens Perspective." In *AAAI Conference and Symposium Proceedings. Workshop Papers 1998*, 60-64. Association for the Advancement of Artificial Intelligence. https://aaai.org/papers/060-ws98-08-016/.

McDonald, David D. 2010. "Natural Language Generation." In *Handbook of Natural Language Processing* edited by Ninit Indurkhya and Fred J. Damerau. 2nd ed., 121–144. New York: Chapman and Hall.

Mitchell, Tom M. 1997. *Machine Learning.* New York: McGraw-Hill. Available at http://www.cs.cmu.edu/~tom/files/MachineLearningTomMitchell.pdf .

Qomariyah, Nunung Nurul. 2018. "Pairwise Preferences Learning for Recommender Systems." PhD diss., University of York, England. https://etheses.whiterose.ac.uk/20365/1/Main.pdf

Quinlan, John Ross. 1986. "Induction of Decision Trees." *Machine Learning* 1: 81-106. https://doi.org/10.1007/BF00116251. Available at https://hunch.net/~coms-4771/quinlan.pdf.

Ricci, Francesco, Lior Rokach, and Bracha Shapira. 2011. "Introduction to Recommender Systems Handbook." In *Recommender Systems Handbook*, edited by Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor. 1-35. New York: Springer. https://doi.org/10.1007/978-0-387-85820-3_1.

Rich, Elaine. 1979. "User Modeling Via Stereotypes." *Cognitive Science* 3: 329-354. https://onlinelibrary.wiley.com/doi/epdf/10.1207/s15516709cog0304_3.

Rosenblatt, Frank. 1958. "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain." *Psychological Review* 65, no. 6: 386–408. https://doi.org/10.1037/h0042519. Available at https://sci-hub.se/10.1037/h0042519.

Samuel, Arthur L. 1959. "Some Studies in Machine Learning Using the Game of Checkers." *IBM Journal of Research and Development* 3, no. 3: 210–229. doi: 10.1147/rd.33.0210. Available at https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5392560.

Samuel, Arthur L. 1988. "Some Studies in Machine Learning Using the Game of Checkers. II - Recent Progress." In *Computer Games I*, edited by David Neil Laurence Levy, 366-400. New York: Springer. https://doi.org/10.1007/978-1-4613-8716-9_15. Also published in *IBM Journal of Research and*

Development 11, no. 6: 601–617 (1967). https://doi.org/10.1147/rd.116.0601. Available at https://www.cs.virginia.edu/~evans/cs6501-s13/samuel.pdf.

Schafer, J. Ben, Joseph Konstan, and John Riedl. 1999. "Recommender Systems in E-Commerce." In *EC '99: Proceedings of the 1st ACM Conference on Electronic Commerce*, edited by Stuart I. Feldman and Michael P. Wellman, 158-166. Denver: Association for Computing Machinery. https://doi.org/10.1145/336992.337035.

Schank, Roger C. 1972. "Conceptual Dependency: A Theory of Natural Language Understanding." *Cognitive Psychology* 3: 552–631. https://doi.org/10.1016/0010-0285(72)90022-9.

Science Direct. 2024. "Vectorization." https://www.sciencedirect.com/topics/computer-science/vectorization.

Singh, Pramod. 2022. "Recommender Systems." In *Machine Learning with PySpark: With Natural Language Processing and Recommender Systems*, edited by Praod Singh, 157-187. Berkeley: Apress. https://doi.org/10.1007/978-1-4842-7777-5_8.

Werbos, Paul John. 1994. *The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting.* New York: John Wiley & Sons.