Identifying specific details from text to populate databases and generate summaries using Named Entity Recognition

Amruta Vikas Patil[a], Hiren Dand[b], Sandeep Kadam[c],

[a] ORCID iD- 0000-0001-6660-725X,

amrutayadav2010@gmail.com, Research Scholar, SJJTU, Jhunjunu, Rajasthan 3333010,
[b] dandhiren@yahoo.co.in, SJJTU, Jhunjunu, Rajasthan 3333010,
[c] sukadam.bscoer@gmail.com, APCOR, Pune, Maharashtra 411041.

***Abstract*—Named Entity Recognition (NER) is a natural language processing (NLP) technique that focuses on identifying and classifying entities within a text. Entities are typically real-world objects that have names, such as people, organizations, locations, dates, percentages, currencies, and more. NER is used to extract structured information from unstructured text data and is an important component of various NLP applications, including question generation, information retrieval, text summarization, and more.**

***Index Terms*— Named Entity Recognition (NER), Natural language processing (NLP), text summarization.**

I. INTRODUCTION

Named Entity Recognition (NER) is a natural language processing (NLP) technique that involves the identification and classification of entities (such as names of people, organizations, locations, dates, monetary values, etc.) within a text. The goal of NER is to extract structured information from unstructured text and categorize it into predefined classes.

Here's how Named Entity Recognition generally works:

1. **Document and use markup styles:** The pull-down style menu is at the left of the Formatting Toolbar at the top of your *Word* window (for example, the style at this point in the document is “Text”). **Tokenization:** The text is first divided into individual words or tokens.
2. **Part-of-Speech Tagging:** Each token is tagged with its part of speech (e.g., noun, verb, adjective).
3. **Named Entity Recognition:** The system then identifies and classifies tokens that belong to named entities. This involves recognizing patterns or features that indicate the presence of entities.

For example, given the sentence: "Apple Inc. was founded by Steve Jobs in Cupertino," a Named Entity Recognition system might identify "Apple Inc." as an organization, "Steve Jobs" as a person, and "Cupertino" as a location.

Popular techniques and models for Named Entity Recognition include:

Rule-based Approaches: These rely on predefined rules and patterns to identify named entities. While simple, they may not be as effective on diverse and complex texts.

Machine Learning Approaches: Supervised learning models, such as Conditional Random Fields (CRF) and Support Vector Machines (SVM), have been used for NER by training on labeled datasets.

Deep Learning Approaches: More recently, deep learning models, including recurrent neural networks (RNNs), long short-term memory networks (LSTMs), and transformers, have shown significant improvements in NER tasks. Pre-trained models like BERT (Bidirectional Encoder Representations from Transformers) have been fine-tuned for NER tasks.

Named Entity Recognition is a crucial component in various natural language processing applications, such as information retrieval, question answering systems, and language understanding in chatbots or virtual assistants. It helps in extracting structured information from unstructured text, making it more accessible for further analysis and processing.

Here are some tips to improve the performance of NER systems:

A. Quality Training Data:

Use high-quality labeled training data that is representative of the types of entities you expect to encounter in real-world scenarios. Ensure a diverse dataset that covers various domains and writing styles.

B. Pre-trained Models:

Utilize pre-trained language models like BERT, GPT, or others as a starting point. These models have learned rich contextual representations from vast amounts of data and can significantly boost NER performance.

C. Fine-tuning:

Fine-tune pre-trained models on your specific NER task. This helps the model adapt to the nuances of your domain and improves performance on entity recognition.

D. Transfer Learning:

Leverage transfer learning techniques to transfer knowledge from a pre-trained model to your NER task. This is particularly effective when you have limited labeled data for your specific task.

E. Ensemble Models:

Combine predictions from multiple models or different architectures to create an ensemble. Ensemble models often provide more robust and accurate results.

F. Hyper parameter Tuning:

Experiment with hyperparameter tuning to optimize the model's performance. This includes adjusting learning rates, batch sizes, and other relevant parameters.

G. Custom Features:

Depending on your specific NER task, consider incorporating additional features or domain-specific knowledge to enhance the model's understanding of entities.

H. Post-processing:

Implement post-processing techniques to refine the extracted entities. This may involve additional validation steps or rules to filter out false positives.

I. Evaluation Metrics:

Choose appropriate evaluation metrics for your task, such as precision, recall, and F1 score. These metrics can guide you in understanding the strengths and weaknesses of your NER system.

J. Error Analysis:

Conduct thorough error analysis to identify common mistakes made by the model. This can guide further improvements and adjustments to the training process.

It's essential to keep in mind that the optimal approach may vary depending on the specific requirements of your NER task. Experimentation, continuous evaluation, and fine-tuning are key components of achieving the best results in Named Entity Recognition. Entity Recognition (NER) can help illustrate the key components and processes involved. Here's a simplified diagram outlining the major steps in a typical NER pipeline:

<table>
<tr><td>Raw Text</td><td>Tokenization</td><td>Part-of-Speech Tag</td></tr>
<tr><td>Apple Inc. was…..</td><td>['Apple', 'Inc.',...|| [('Apple', 'NNP'),]</td><td>('Inc.', 'NNP'), ('was', 'VBD'),</td></tr>
</table>

<table>
<tr><td>Named Entity</td><td>Named Entity</td><td>Recognition</td><td>Named Entity</td><td>Classification</td><td>Recognition</td></tr>
<tr><td>'Apple Inc.',</td><td>'Apple Inc.':</td><td>'Apple Inc.':</td><td>'ORG') ,|Steve | Jobs', 'PERSON'),</td><td>'ORG' , 'Steve |Jobs': 'PERSON',</td><td>Jobs': 'PERSON',|'ORG' , 'Steve</td></tr>
</table>

Named Entity Recognition (NER) is a natural language processing technique that involves identifying and classifying named entities (specific words or phrases that refer to real-world objects such as people, places, organizations, dates, etc.) in text. NER plays a crucial role in various language-related tasks such as information extraction, question generation, sentiment analysis, and more. It helps computers understand the context and relationships between different entities mentioned in a text.

Here's how NER generally works:

Tokenization: The text is divided into individual words or sub words (tokens).

Part-of-Speech Tagging: Each token is tagged with its grammatical part of speech (noun, verb, etc.).

Entity Recognition: Based on patterns and contextual information, NER algorithms identify sequences of tokens that likely correspond to named entities.

Entity Classification: Once identified, the named entities are classified into predefined categories like person, organization, location, date, etc.

For example, consider the sentence: "Apple Inc. was founded by Steve Jobs in Cupertino."

NER might process this sentence as follows:

"Apple Inc." -> Organization
"Steve Jobs" -> Person
"Cupertino" -> Location
Applications of NER include:

Information Extraction: Identifying specific details from text to populate databases or generate summaries.
Question Generation: Helping generate questions that refer to specific entities in the text.
Search Engine Enhancement: Improving search results by recognizing and highlighting relevant entities in search queries.
Document Categorization: Automatically classifying documents based on the entities mentioned.
Sentiment Analysis: Recognizing entities that are associated with positive or negative sentiments.
State-of-the-art NER models often leverage machine learning techniques, especially models like recurrent neural networks (RNNs), transformers, and conditional random fields (CRFs). These models are trained on large labeled datasets to learn patterns and contexts that signify named entities. Libraries such as NLTK, spaCy, and the NER models provided by Hugging Face's Transformers can be used for NER tasks.
Named Entity Recognition (NER) is a natural language processing (NLP) technique that focuses on identifying and classifying entities within a text. Entities are typically real-world objects that have names, such as people, organizations, locations, dates, percentages, currencies, and more. NER is used to extract structured information from unstructured text data and is an important component of various NLP applications, including question generation, information retrieval, text summarization, and more.

The primary goal of NER is to locate and label entities in the text. Here's a basic overview of how NER works:

Tokenization: The input text is divided into individual tokens, which can be words or subwords.

Part-of-Speech Tagging: Each token is assigned a part-of-speech tag (noun, verb, adjective, etc.).

Named Entity Recognition: Using contextual information from the text and patterns learned from training data, the NER algorithm identifies spans of tokens that correspond to named entities.

Entity Classification: The identified spans are then classified into predefined categories such as "PERSON" for names of people, "ORG" for organizations, "LOC" for locations, "DATE" for dates, etc.

For example, consider the sentence: "Apple Inc. was founded by Steve Jobs and Steve Wozniak on April 1, 1976, in Cupertino."

NER would analyze this sentence and identify the following named entities:
"Apple Inc." -> ORGANIZATION
"Steve Jobs" -> PERSON
"Steve Wozniak" -> PERSON
"April 1, 1976" -> DATE
"Cupertino" -> LOCATION
NER models are trained on large labeled datasets that provide examples of entities and their corresponding categories. Machine learning algorithms, such as conditional random fields, support vector machines, and more recently, deep learning approaches like bidirectional LSTM and transformer-based models (such as BERT and GPT), are used to perform NER.

NER is valuable for a wide range of applications, including:

Information Extraction: Identifying structured information from unstructured text.
Question Generation: Creating questions that refer to specific entities mentioned in the text.
Information Retrieval: Improving search results by identifying key entities.
Text Summarization: Focusing on important entities when generating summaries.
Entity Linking: Associating entity mentions with knowledge bases or databases.Overall, NER plays a crucial role in transforming textual data into structured and useful information for various downstream tasks in natural language processing.

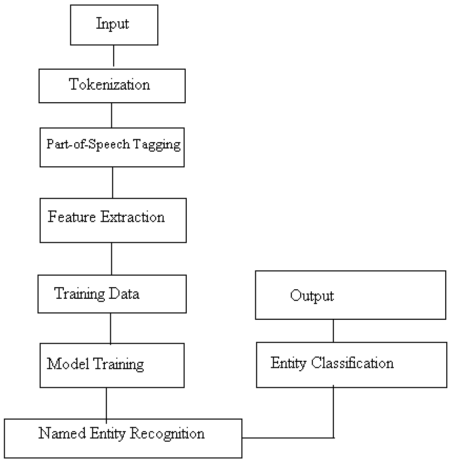

Fig. Flowchart for Named Entity Recognition (NER)

1. Input Text: Start with the unstructured text that you want to perform NER on.
2. Tokenization: Divide the text into individual tokens (words or sub words).
3. Part-of-Speech Tagging: Assign a part-of-speech tag to each token (noun, verb, adjective, etc.).
4. Feature Extraction: Extract relevant features from tokens and their context, such as word embeddings or linguistic features.
5. Training Data: Use a labeled dataset with examples of text and corresponding entity labels (e.g., "Apple Inc." -> ORGANIZATION). This data is used to train the NER model.
6. Model Training: Train a machine learning model (e.g., conditional random fields, LSTM, transformer-based models like BERT) on the labeled dataset. The model learns to recognize patterns that indicate named entities.
7. Named Entity Recognition: Apply the trained model to the tokenized and tagged text. The model identifies spans of tokens that represent named entities.
8. Entity Classification: For each identified span, classify the entity into predefined categories (e.g., PERSON, ORGANIZATION, LOCATION) based on the model's predictions.
9. Output: Generate a list of recognized named entities along with their corresponding categories.
10. Post-Processing: Apply any necessary post-processing steps, such as merging adjacent entity spans, filtering out false positives, or resolving overlapping entities.
11. Visualization (Optional): Visualize the recognized entities in the original text or in a separate display.
12. Use in Applications: The recognized named entities and their categories can be used in various downstream applications, such as question generation, information extraction, or search.

Named Entity Recognition (NER) techniques offer several advantages in various natural language processing (NLP) applications:
1. Information Extraction: NER helps extract structured information from unstructured text, making it easier to identify and organize key entities such as names of people, organizations, locations, dates, and more.
2. Improved Search and Retrieval: NER can enhance search engines by identifying and highlighting important entities in search results, improving the accuracy and relevance of search queries.
3. Question Generation: NER is useful for generating questions based on specific entities mentioned in the text. This is valuable for educational platforms, chatbots, and assessment tools.
4. Text Summarization: NER can aid in identifying and focusing on the most important entities when generating concise summaries of longer texts.
5. Sentiment Analysis: By recognizing entities, NER can help determine the sentiment associated with specific entities, providing more granular insights into opinions and attitudes.
6. Event and Trend Analysis: NER can assist in tracking and analyzing events, trends, or occurrences related to specific entities over time.
7. Document Categorization: NER can contribute to classifying documents or text segments based on the types of entities mentioned, aiding in content categorization.
8. Relationship Extraction: NER can help identify relationships between entities, leading to insights about connections between people, organizations, and locations.
9. Knowledge Graph Construction: NER plays a role in building knowledge graphs by identifying nodes (entities) and edges (relationships) between them, enhancing knowledge representation.
10. Named Entity Disambiguation: NER can assist in distinguishing between entities with the same name but different meanings, contributing to contextually accurate analysis.
11. Language Translation: NER can aid in improving the accuracy of language translation by identifying and correctly translating named entities.
12. Entity Linking: NER can link recognized entities to external knowledge bases, enriching the understanding and context of the text.
13. Data Mining and Analytics: NER helps extract relevant information from large volumes of text data, enabling effective data mining and analysis.
14. Legal and Regulatory Compliance: NER assists in identifying and categorizing legal entities, regulations, dates, and other critical information in legal documents.
15. Medical and Scientific Research: NER is valuable in identifying medical entities (diseases, medications, etc.) and scientific terms, aiding in research and analysis.
16. Content Recommendation: NER can contribute to personalized content recommendations by understanding user preferences based on recognized entities.

NER vs. Rule-Based Approaches:
• NER: NER utilizes machine learning algorithms to learn patterns and context from data, allowing it to recognize entities even in complex and varied contexts. It is more flexible and adaptable to different languages and domains. Rule-Based Approaches: Rule-based systems rely on hand-crafted rules and patterns to identify entities. While they can be effective for simple cases, they may struggle with handling variations, nuances, and evolving language patterns.

NER vs. Keyword Matching:
NER: NER can identify entities based on context, considering not only the keyword itself but also its linguistic environment. It can handle synonyms, ambiguous cases, and variations of entities more effectively. Keyword Matching: Keyword-based approaches are limited to specific keywords or phrases and may miss entities not explicitly included in the keyword list.

NER vs. Named Entity Linking:
NER: NER identifies and classifies entities within the text. Named Entity Linking: This extends NER by linking recognized entities to external knowledge bases or databases. It enriches the understanding of entities and provides more context.

NER vs. Named Entity Disambiguation:
NER: NER identifies entities, but in some cases, entities can have multiple meanings. Disambiguation is not a direct part of NER but can be a separate step. Named Entity Disambiguation: This process disambiguates entities with multiple possible meanings based on the context in which they appear.

NER vs. Information Extraction:
NER: NER is a component of information extraction, focused on recognizing entities. Information Extraction: This involves extracting structured information from unstructured text, which may involve NER as well as other techniques to extract relationships and events.

NER vs. Co reference Resolution:
NER: NER identifies entities, but it doesn't inherently handle co reference resolution (linking different mentions of the same entity). Co reference Resolution: This process identifies when different mentions in the text refer to the same entity, which can be a separate step after NER.

NER is a fundamental technique for identifying and classifying named entities in text. It offers advantages over rule-based systems and simple keyword matching by leveraging machine learning to handle context, variations, and complexities. NER can be part of a larger pipeline involving other NLP techniques, such as named entity linking, disambiguation, and co reference resolution, to achieve more comprehensive information extraction and analysis.

Advantages of NER:
1. Focused Entity Recognition: NER is designed specifically for identifying and classifying named entities, making it highly effective for information extraction and organization.
2. Structured Information: NER outputs structured information, which is valuable for applications like database population, content categorization, and search enhancement.
3. Domain Adaptation: NER systems can be tailored to specific domains or industries, enabling accurate recognition of domain-specific entities.
4. Data Efficiency: NER models can achieve good performance with relatively small amounts of labeled data, making them useful for specialized applications.
5. Entity Linking: NER can be extended to link recognized entities to external knowledge bases, enriching the understanding of entities.
Comparison:
• Scope and Use Cases: T5 is a general-purpose architecture suitable for various text-based tasks, while NER is focused on recognizing specific types of named entities. T5 can handle broader tasks like summarization and translation, while NER is particularly useful for tasks requiring entity recognition and categorization.
• Flexibility: T5 can be fine-tuned for a wide range of tasks, offering more flexibility. NER is tailored for entity recognition tasks.
• Structured Output: NER provides structured output in terms of recognized entities and their categories, which is advantageous for applications requiring organized data.
• Training Data: NER models can perform well with smaller amounts of labeled data, while T5 models typically require more data for effective fine-tuning.

RESULT AND DICUSSION

Example Sentence:
"The capital of France is Paris, and the Eiffel Tower is located in that city."
Tokenization:
["The", "capital", "of", "France", "is", "Paris", ",", "and", "the", "Eiffel", "Tower", "is", "located", "in", "that", "city", "."]
Part-of-Speech Tagging:
[("The", "DT"), ("capital", "NN"), ("of", "IN"), ("France", "NNP"), ("is", "VBZ"), ("Paris", "NNP"), (",", ","), ("and", "CC"), ("the", "DT"), ("Eiffel", "NNP"), ("Tower", "NNP"), ("is", "VBZ"), ("located", "VBN"), ("in", "IN"), ("that", "DT"), ("city", "NN"), (".", ".")]

Named Entity Recognition:
[("The", "O"), ("capital", "O"), ("of", "O"), ("France", "LOCATION"), ("is", "O"), ("Paris", "LOCATION"), (",", "O"), ("and", "O"), ("the", "O"), ("Eiffel", "LOCATION"), ("Tower", "LOCATION"), ("is", "O"), ("located", "O"), ("in", "O"), ("that", "O"), ("city", "O"), (".", "O")]

II. CONCLUSION

A conclusion section is not required. Although a conclusion may review the main points of the paper, do not replicate the abstract as the conclusion. A conclusion might elaborate on the importance of the work or suggest applications and extensions.

APPENDIX

Appendixes, if needed, appear before the acknowledgment.

ACKNOWLEDGMENT

REFERENCES

[1] G. O. Young, “Synthetic structure of industrial plastics (Book style with paper title and editor),” in *Plastics*, 2nd ed. vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15–64.

[2] W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.

[3] H. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer-Verlag, 1985, ch. 4.

[4] B. Smith, “An approach to graphs of linear forms (Unpublished work style),” unpublished.

[5] E. H. Miller, “A note on reflector arrays (Periodical style—Accepted for publication),” *IEEE Trans. Antennas Propagat.*, to be published.

[6] J. Wang, “Fundamentals of erbium-doped fiber amplifiers arrays (Periodical style—Submitted for publication),” *IEEE J. Quantum Electron.*, submitted for publication.

[7] C. J. Kaufman, Rocky Mountain Research Lab., Boulder, CO, private communication, May 1995.

[8] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interfaces (Translation Journals style),” *IEEE Transl. J. Magn.Jpn.*, vol. 2, Aug. 1987, pp. 740–741 [*Dig. 9th Annu. Conf. Magnetics* Japan, 1982, p. 301].

[9] M. Young, *The Techincal Writers Handbook*. Mill Valley, CA: University Science, 1989.

[10] J. U. Duncombe, “Infrared navigation—Part I: An assessment of feasibility (Periodical style),” *IEEE Trans. Electron Devices*, vol. ED-11, pp. 34–39, Jan. 1959.

[11] S. Chen, B. Mulgrew, and P. M. Grant, “A clustering technique for digital communications channel equalization using radial basis function networks,” *IEEE Trans. Neural Networks*, vol. 4, pp. 570–578, Jul. 1993.

[12] R. W. Lucky, “Automatic equalization for digital communication,” *Bell Syst. Tech. J.*, vol. 44, no. 4, pp. 547–588, Apr. 1965.

[13] S. P. Bingulac, “On the compatibility of adaptive controllers (Published Conference Proceedings style),” in *Proc. 4th Annu. Allerton Conf. Circuits and Systems Theory*, New York, 1994, pp. 8–16.

[14] G. R. Faulhaber, “Design of service systems with priority reservation,” in *Conf. Rec. 1995 IEEE Int. Conf. Communications*, pp. 3–8.

[15] W. D. Doyle, “Magnetization reversal in films with biaxial anisotropy,” in *1987 Proc. INTERMAG Conf.*, pp. 2.2-1–2.2-6.

[16] G. W. Juette and L. E. Zeffanella, “Radio noise currents n short sections on bundle conductors (Presented Conference Paper style),” presented at the IEEE Summer power Meeting, Dallas, TX, Jun. 22–27, 1990, Paper 90 SM 690-0 PWRS.

[17] J. G. Kreifeldt, “An analysis of surface-detected EMG as an amplitude-modulated noise,” presented at the 1989 Int. Conf. Medicine and Biological Engineering, Chicago, IL.

[18] J. Williams, “Narrow-band analyzer (Thesis or Dissertation style),” Ph.D. dissertation, Dept. Elect. Eng., Harvard Univ., Cambridge, MA, 1993.

[19] N. Kawasaki, “Parametric study of thermal and chemical nonequilibrium nozzle flow,” M.S. thesis, Dept. Electron. Eng., Osaka Univ., Osaka, Japan, 1993.

[20] J. P. Wilkinson, “Nonlinear resonant circuit devices (Patent style),” U.S. Patent 3 624 12, July 16, 1990.

[21] *IEEE Criteria for Class IE Electric Systems* (Standards style), IEEE Standard 308, 1969.

[22] *Letter Symbols for Quantities*, ANSI Standard Y10.5-1968.

[23] R. E. Haskell and C. T. Case, “Transient signal propagation in lossless isotropic plasmas (Report style),” USAF Cambridge Res. Lab., Cambridge, MA Rep. ARCRL-66-234 (II), 1994, vol. 2.

[24] E. E. Reber, R. L. Michell, and C. J. Carter, “Oxygen absorption in the Earth’s atmosphere,” Aerospace Corp., Los Angeles, CA, Tech. Rep. TR-0200 (420-46)-3, Nov. 1988.

[25] (Handbook style) *Transmission Systems for Communications*, 3rd ed., Western Electric Co., Winston-Salem, NC, 1985, pp. 44–60.

[26] *Motorola Semiconductor Data Manual*, Motorola Semiconductor Products Inc., Phoenix, AZ, 1989.

[27] (Basic Book/Monograph Online Sources) J. K. Author. (year, month, day). *Title* (edition) [Type of medium]. Volume (issue). Available: http://www.(URL)

[28] J. Jones. (1991, May 10). Networks (2nd ed.) [Online]. Available: http://www.atm.com

[29] (Journal Online Sources style) K. Author. (year, month). Title. *Journal* [Type of medium]. Volume(issue), paging if given. Available: http://www.(URL)

[30] R. J. Vidmar. (1992, August). On the use of atmospheric plasmas as electromagnetic reflectors. *IEEE Trans. Plasma Sci.* [Online]. *21(3)*. pp. 876–880. Available: http://www.halcyon.com/pub/journals/21ps03-vidmar

.