UtilitasMathematica

ISSN 0315-3681 Volume 122, 2025

Knowledge-Based Natural Language Processing for Trained Language Models: A Technical Analysis

Dr. Anil Kumar[1], Dr. Mukesh Yadav[2], Dr Vandana Yadav[3] and Dr Narender Singh[4]

[1]Department of Applied Science and Humanities (Mathematics)
IMS Engineering College Ghaziabad UP India
[2] Department of Computer Science and Engineering

DPG Institute of Technology and Management Gurugram Haryana India

[3]Department of Computer Science and Engineering

Brainware University Barasat Kolkata West Bengal India

[4]Department of Mathematics

Govt. College Birohar Jhajjar, Haryana India

dranilkumar73@rediffmail.com

Abstract: This paper investigate the era of data-driven technologies, when the development of Natural Language Processing (KI-NLP) systems focuses more and more on rich and varied external knowledge sources, including domain-specific resources, mathematical equations, online information available, and common sense reasoning. Knowledge-Intensive NLP has greatly evolved due to the expanding capabilities of pre-trained language models (PLMs), which provide improved performance, flexibility, and robustness across a range of arduous applications. Because it is difficult to integrate, retrieve, and reason over external knowledge, PLMs are inherently limited in their ability to manage knowledge-intensive jobs, even with these improvements. Using directed graphs, Kepler offers a straightforward method for creating and carrying out intricate operations. The primary problems, major patterns, and future prospects of KI-NLP research are highlighted in this thorough technical analysis of the field's current state. We investigate the development and application of knowledge-enhanced PLMs, which integrate framework dependencies and mathematical models to direct pre-training approaches across several languages. Additionally, through investigating three crucial elements—knowledge sources, KI-NLP task categories, and knowledge fusion techniques—we classify the development of Pretrained Language Model-based Knowledge-Enhanced systems/PLMKEs and survey recent work. The current study investigates how external knowledge and language model pre-training can work together to improve future developments in the sector.

Keywords: Mathematical equations, fusion approaches, knowledge-intensive natural language processing, mathematical models, machine learning models, and BERT-Cap models.

1. Introduction

A wide range of NLP issues have been resolved by applying knowledge-based solutions. Some of the well-known problems are illustrated in this section, along with techniques for solving them that use knowledge-based selection. A number of excellent systems that have tackled the issues are also shown to the reader. A full description of some of the most effective KB-NLP systems to date is also provided by these system descriptions. We will discuss systems that use knowledge-based inference, which may be thought of as NLP applications. With state-of-the-art performance on a variety of tasks, the BERT [1] and GPT [2] families of massive, pre-trained transformer-based language models (PLMs) have completely transformed the area of natural language processing (NLP). These models have caused a paradigm shift in natural language processing (NLP), with end-to-end learning frameworks taking the place of conventional feature engineering techniques.

In contrast, deep learning-based methods utilize neural networks to jointly learn latent feature representations and the classification function [3-4]. Basic research issues in NLP knowledge acquisition deal with potential methods of automating knowledge acquisition and the creation of strategies for collecting world and linguistic knowledge simultaneously in a way that is compatible with each other. It is evident that various sources, specialists, and limitations influence how lexical and world knowledge are acquired, particularly in multilingual systems. It is crucial to have a situated, integrated approach to acquisition in which the teams responsible for acquiring language and world information regularly communicate with one another and discuss their options for each representation. To ensure compatibility between the information sources, such an incremental process together with a set of tools for quality control and interaction assistance is unavoidable. Additionally, it will ensure that all of the learned material is indeed applicable to NLP.

Named entity disambiguation (NED), a challenging endeavor in natural language processing, entails connecting textual references to named entities to their corresponding knowledge network entries. Two new pre-training objectives have been added to a revolutionary pre-training NED model in order to address this. The suggested model performs substantially higher than current methods, according to experimental tests on the CoNLL and TAC datasets as well as benchmarks from the GERBIL platform [4].

Because denoising autoencoding techniques, such as BERT, can capture bidirectional contextual information, they have outperformed autoregressive language modeling techniques among pre-training strategies [5,40]. Even while neural networks for language modeling have demonstrated impressive performance on a variety of NLP subtasks, deep language model training is still time-consuming and computationally costly. It's interesting to note that multi-class text classification task results indicate that general-purpose models like BERT may not always outperform domain-specific models like FinBERT [6].

For many PLMs, extracting factual knowledge from textual data is still a challenge. Although relational facts from knowledge graphs can be effectively represented by knowledge embedding (KE) techniques using entity embeddings, rich textual information is difficult for typical KE models to utilize. A unified approach known as KEPLER, which blends Knowledge Embedding with Pre-trained Language Representation, has been proposed to close this gap. KEPLER improves PLM performance and knowledge graph representation by skillfully combining textual and factual knowledge [7].Despite their success, pre-trained language models remain susceptible to adversarial inputs. For example, synonym-based word substitution attacks can mislead even well-trained BERT-based sentiment analysis systems [8].

Text mining has been essential to the process of gleaning knowledge from unstructured text during the last few decades. NLP's incorporation of deep learning and neural networks has produced several real-world achievements. The ability of these models to capture intricate language aspects like sentiment, long-range connections, and hierarchical organization is comparable to how convolutional neural networks transformed image processing with ImageNet [9, 10].

2. Computational Model

(i) The BERT-Cap Model

A BERT-Cap hybrid model with focused loss based on pretrained BERT and capsule network has been presented for the classification of user intent [1,6]. The BERT-Cap model consists of four modules: input embedding, sequence encoding, feature extraction, and intent categorization. Our model's architecture is displayed in Figure 1. Given a sentence as input, the input embedding module preserves token, position, and segment information to represent the sentence to a series of embeddings. The sequence encoding module uses the transformer encoder to encode sentences after loading the pre-trained language model created using transfer learning.

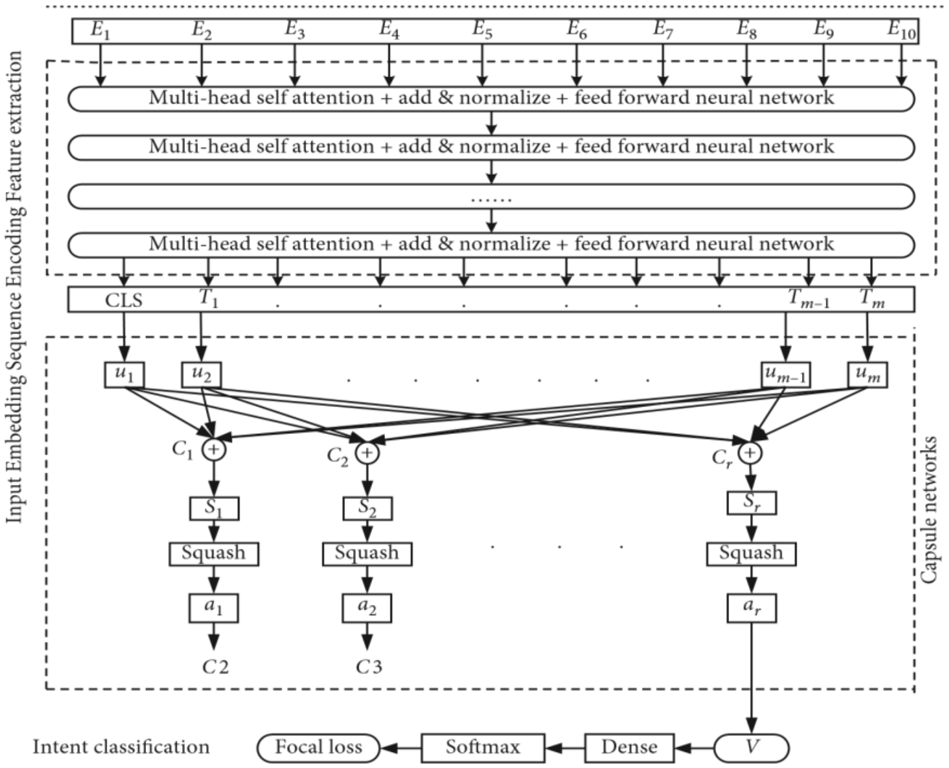

Fig 1: BERT Architecture Model

(i) Token-based Pre-trained Models

Because all of the previous token-based pre-trained models have emerged, word embeddings have become a popular method for representing text in NLP tasks. Despite their straightforwardness and effectiveness, these models are not suitable for capturing polysemy; instead, they are only suitable for achieving fixed representations. For this reason, we also refer to these models as static pre-trained models..

(ii) Context-based Pre-trained Models

It is necessary for pre-trained models to differentiate between word semantics and dynamically produce word embedding's in various settings in order to tackle the issue of polysemy.. Given a text $x_1, x_2, \ldots, x_T$where each token x_t is a word.
Suppse x_t depends on the whole text.

$$\left[t_1, \mathrm{t}, \ldots, t_T\right] = f_{enc}(x_1, x_2, \ldots, x_T)\,, \qquad (1)$$

where $f_{enc}(\cdot)$ is neural encoder and h_t is contextual embedding.

3. Models of Language Structures

To tackle the difficulties related to software security, it is crucial to comprehend the language model construct that successfully applies. Software security issues can be resolved by utilizing the natural language data that is accessible to businesses and sectors. Deriving the information concealed in this data can be aided by effective language modeling capabilities. In addition, the software development team can use security-related information when their needs arise. The author's models employed in this study are based on language models. Language models are derived from N-gram modeling techniques. Through the use of a word's history, N-gram modeling determines the likelihood of a given word. [31-37].

. In the sentence "Jack and Jill went up the," for instance, the likelihood that the following word will be "hill" is high.

P (hill|Jack and Jill went up the) (1)

Relative frequency is one method used to calculate this likelihood. The following formula can be used to determine how frequently the word "hill" follows the phrase given the corpus of language as a base:P(hill|Jack and Jill went up the) = $\frac{C(JackandJillwentupthehill)}{C(JackandJillwentupthe)}$ (2)

where C denotes the count of occurrence of the phrase.
When words are subjected to the chain rule of probability, the following expression is produced:

$$P\left(W_{1:n}\right) = P\left(w_1\right) P\left(w_2 \middle| w_1\right) P\left(w_3 \middle| w_{1:2}\right) \ldots P(w_n | w_{1:n-1}) \\ = \prod_{k=1}^{n} P\left(w_k \middle| w_{1:k-1}\right), \qquad (3)$$

where k is the length of the sequence, n is the word count, and w is the word. Markov will only need to examine n − 1 prior words to simplify the intricacy of dependence on the term. The probability of N-grams are computed using maximum likelihood estimation in the manner described below:

$$P\left(w_n \backslash wn - 1\right) = \frac{c\left(w_{n-1} w_n\right)}{w_{n-1}}$$

KEPLER

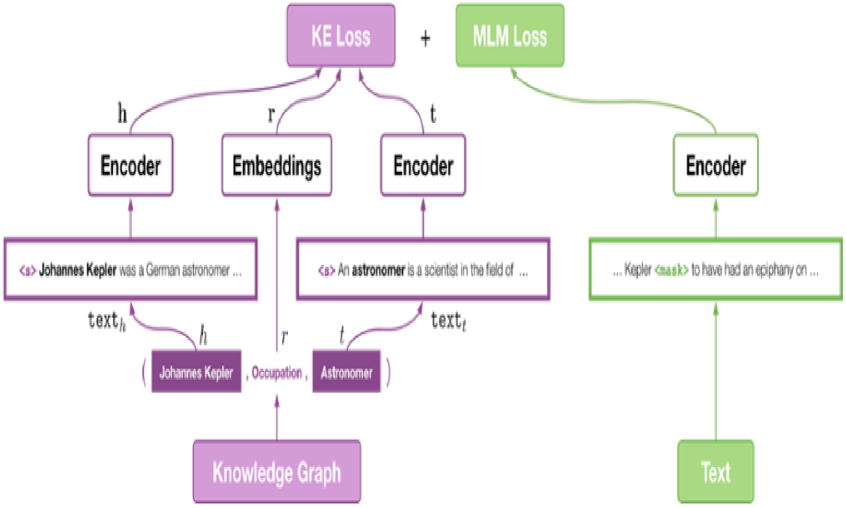

***Fig 2**: The KEPLER framework*

The knowledge embedding (KE) and masked language modeling (MLM) objectives are concurrently trained on the same PLM after we encode entity descriptions as entity embeddings [32-39].

4. Knowledge Embedding in KEPLER

In our pre-training, we incorporate factual knowledge into KEPLER using the knowledge embedding (KE) goal. KE helps with a number of downstream tasks, such as link prediction and relation extraction, by encapsulating entities and relations in knowledge graphs (KGs) using distributive representations.
The description of an entity as an embedding A relational triplet (h, r, t) has the following:h=$E_{<s>}\left(text_h\right)$,

$$t = E_{<s>}(text_t), \\ r = T_r,$$

Where $text_h$ and $text_t$ are the descriptions for h and t, with a special token <s> at the beginning. $T \epsilon \mathbb{R}^{|R| \times d}$is the relation embeddings and h, t, r are the embeddings for h, t, and r.

4.1 Knowledge Embedding in KEPLER

It is rare to find 10,000-word NLP systems that are highly comprehensive in the topic under consideration. Some authors have accomplished this goal by utilizing prototypes that leverage existing corpora of linguistic knowledge. This is the most effective method if we do not wish to rebuild the wheel. Reusing existing sources is therefore necessary.

One by one, think about the numerous sources that are accessible. To highlight their distinctiveness, their evaluation will be based on the kind of language expertise they can provide.

While broad classifications in Natural Language Processing (NLP) provide a useful framework, they must be refined further to reflect the complexities of real-world implementations. Effective NLP systems rely not only on algorithms but also on a comprehensive set of linguistic rules and rich language resources.

At its core, NLP is built upon vocabulary and syntactic knowledge. Large-scale lexicons are essential; without them, even the most sophisticated models struggle to function effectively. However, challenges arise when extending these lexicons to a multilingual context. Specifically, how can we align equivalent terms across different languages?

A straightforward approach is to map words from various languages to a shared abstract representation—commonly referred to as an *idea* or *concept*. Once a few thousand such concepts are gathered, the next step is to organize them systematically. This structured representation is often termed a *concept typology* or *ontology*.

Implementations that focus on naming and categorization frequently employ single-inheritance hierarchies. In contrast, more advanced systems leverage domain-compositional models, which are better suited to provide the rich semantic structures required by modern NLP tools. Despite their importance, comprehensive domain representation initiatives remain limited in number.

One notable effort in this area is the **Unified Medical Language System (UMLS)**, which serves as a key resource for semantic interoperability in the biomedical domain. UMLS integrates a wide range of terminologies and ontologies, offering a model for how complex domain knowledge can be structured and linked for NLP applications.The **Unified Medical Language System (UMLS)** provides the medical community with extensive access to a wealth of linguistic and conceptual data. However, the structure and organization of this knowledge base represent a compromise among various modeling approaches, balancing coverage, usability, and interoperability.

4.2 The GALEN Knowledge Source

The **GALEN** (Generalised Architecture for Languages, Encyclopaedias, and Nomenclatures in medicine) initiative has been active since 1992. It has developed a comprehensive, generic model of medicine encompassing approximately 6,000 concepts, represented using the **GRAIL** (GALEN Representation And Integration Language) formalism. This representation is compositional in nature, allowing for a high degree of conceptual complexity and fine-grained semantic granularity. GALEN's modeling approach is compatible with conceptual graphs and shares similarities with frame-based systems, offering a structured and expressive view of medical concepts.

Despite its strengths, the GALEN framework presents challenges in terms of scalability and practical deployment. Its detailed compositional nature makes it difficult to achieve broad, ready-to-use coverage across the medical domain, limiting its utility in real-time clinical applications without significant customization.

4.3 The MED Knowledge Source

The Medical Entity Dictionary (MED) serves as a domain-specific, frame-like model aimed at supporting medical applications with a controlled vocabulary. Built upon the semantic framework of the UMLS, MED is designed to offer a more focused and operational terminology resource. It is continually extended to meet the evolving needs of associated healthcare information systems, prioritizing practical integration over broad conceptual modeling..

Knowledge Types	Knowledge Sources	Knowledge Domains
Encyclopaedic Knowledge	Wikipedia /Wikidata [31]	Open domain
	DBpedia	
	Freebase	
	UMLS Bodenreider	Biomedicine
	AMiner	Science
Commonsense Knowledge	Concept Net	Open domain
	TransOMCS	
	CSKG	
	ATOMIC	Human interaction
	ATOMIC	
	ASER	Eventuality

Table 1: Common knowledge sources used in PLMKEs

5. Knowledge-Intensive NLP

5.1 Knowledge-Intensive Tasks and Evidence-Based Generation

Knowledge-intensive tasks—such as open-domain Question Answering (QA) and fact-checking—require the retrieval of relevant evidence passages from large corpora, such as Wikipedia, in response to a given query. Among the most effective approaches for these tasks is retrieval-augmented generation (RAG) [33], in which a pre-trained retriever first gathers candidate passages, and a separate generator is then trained to produce the final answer based on the retrieved content.

However, recent studies (e.g., Xu et al.) have highlighted a critical limitation of this paradigm: the training process often ignores the *evidentiality* of the retrieved passages—i.e., whether or not the passages actually support the generated answer. This can lead to the model relying on spurious lexical cues or producing *hallucinated* responses. For example, in QA tasks, answers may be incorrectly generated from passages that have high lexical similarity to the question but lack factual grounding. Such errors are often exacerbated by the model's tendency to memorize outdated or non-evidence-based knowledge during training.

To mitigate this issue, some QA systems employ heuristics that prioritize passages containing specific target strings during training. While this can reduce hallucination in closed-form QA, it often fails to guarantee true evidential support and is generally ineffective for more complex applications involving open-ended generation or classification tasks.

5.2 Base Generator: Fusion-in-Decoder

For our base generator model GGG, we utilize **FiD (Fusion-in-Decoder)**—a state-of-the-art retrieval-augmented generation framework. FiD is designed to process multiple retrieved passages by independently encoding each one and then fusing the representations within the decoder to generate an informed response. For a detailed architectural and implementation overview, readers are referred to Izacard and Grave (2021b).

- Encoder:
- Using a pre-trained T5 encoder, we first encode an input query and passages. Each passage has the input question x prepended to it, and the encoder encodes all N passages individually. We convert passage pi into pi R(Lh), where L is the length of the input text and h is the concealed size.
- Answer generator:

$\hat{P}$ is a concatenation of the encoded sections that forms a summary representation of the input. P is input into the response generator, which generates the final answer autoregressively. It produces the sequence probability for y in the following format:

$$P\,(y/x\,,\hat{P}) = \prod_{j=1}^{T} p(y_i|y_{<j}, x, \hat{P}).$$

where y_i denotes the j th token of the generated output y and T is the length of the final output. The generator is based on the T5 architecture and uses cross attentions to model the interactions between retrieved passages. Mining Silver Evidentiality E^{silver}

In most existing datasets and tasks, annotations are limited to query-answer pairs (x,y)(x, y)(x,y), without providing explicit **evidentiality labels** EEE for the supporting passages. While some datasets, such as **Natural Questions** and **HotpotQA** [34], include gold-labeled evidence passages, they typically cover only a limited subset of relevant documents (e.g., specific Wikipedia articles). Consequently, the full passage set PPP may still contain unlabeled but potentially valid evidence from other sources or pages.

A common but imperfect workaround involves using **heuristic labeling**, where passages containing the target answer string are automatically labeled as evidentially positive. However, this approach often introduces **false positives**. For instance, a passage p2p_2p2 might contain the answer string “seven” but lack any actual relevance to the input query. In such cases, the superficial presence of the answer token does not equate to genuine evidential support.

Crucially, even this noisy heuristic cannot be applied to more complex tasks—such as **knowledge-enhanced discourse generation** or **fact verification**—which require open-ended text generation or classification rather than direct string matching. These tasks demand a deeper, semantically grounded understanding of evidence, which remains a significant challenge in current NLP systems..

- Leave-one-out evidentiality mining:

Gold evidence annotations are missing from the majority of knowledge-intensive datasets. We provide a new method for mining evidentiality data that involves determining whether passages give enough information for a trained model to create.

Task	Datasets	Data Sources

	NATURAL QUESTIONS	Wikipedia
	HOTPOTQA [35]	Wikipedia
Fact Verification	FEVER	Wikipedia
	BaolQ	Wikipedia
Entity Linking	ACE2004 [36]	news
	AIDA CoNLL-YAGO	DBPedia & YAGO
	WNWI	Wikipedia
	WNCW	Clueweb

***Table 2:** Detailed information about representative encyclopedic knowledge-intensive tasks and datasets.*

6. **Result and Discussion**

This paper investigates the key challenges associated with the construction of knowledge-intensive NLP systems. In response to these challenges, a growing body of research has focused on integrating PLMs with external knowledge sources, leading to the rapid emergence of knowledge-enhanced models. We explore this evolving landscape by examining the difficulties inherent in building such models, particularly in terms of multilingual pre-training and framework reliability. Knowledge can be abstractly represented through *entities* (nodes), *relations*, and *attributes* (edges). In the context of *open network knowledge fusion*, the process begins with identifying and linking entities based on their relationships and attributes. These linked entities are then merged with the existing knowledge base, updating and enriching it based on the inferred connections. The core of this process is *entity matching and linking*—often referred to as the *entity linking technique*. The objective is to dynamically associate entities identified from diverse sources, such as the web, with entries in a structured knowledge base. This facilitates the continual evolution and accuracy of knowledge representations.

Named Entity Recognition (NER) plays a foundational role in the entity linking pipeline by identifying entity mentions within unstructured text and categorizing them into predefined classes (e.g., person, organization, location).NER methods generally fall into three categories:

- Rule-based approaches
- Statistical learning-based approaches
- Neural network-based approaches

Rule-based methods [48], which include techniques such as pattern matching and finite automata, were the dominant paradigm in the early stages of NER development. These methods have shown success in both English and Chinese. For instance, Chinese organizational names—such as *Dalian Wanda Group* or Despite their initial success, rule-based systems suffer from several limitations:

- High dependence on handcrafted rules and domain expertise
- Poor adaptability across languages or domains
- Difficulty in scaling and maintaining rule sets

As a result, modern NER increasingly relies on statistical and deep learning models, which offer greater robustness and generalizability across diverse text corpora and languages..

Task	Dataset	Models	Fusion Types	Fused Knowledge	
	NATURALS QUESTIONS [34]		UnitedQA	Post-fusion	Wikipedia
	WEBQUESTIONS		EMDR	Post-fusion	Wikipedia
Open-domain QA	TEIVIAQA		EMDR	Post-fusion	Wikipedia

		REALM	Hybrid-fusion	Wikipedia
Other Representative model		Hybrid-fusion	Wikipedia	

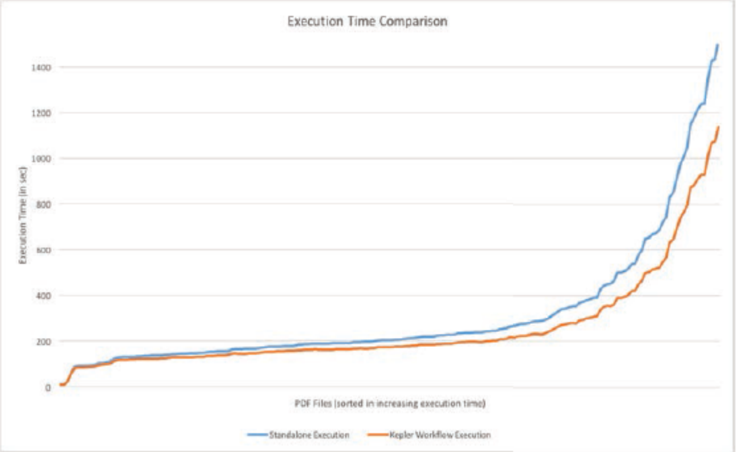

Figure 3: Comparison of Standalone and Kepler Workflow Execution Times

7. CONCLUSIONS

The advancement of **knowledge-intensive natural language processing (KI-NLP)** has been significantly driven by the increasing capabilities of **pre-trained language models (PLMs)**. These models have enabled the development of more **stable, flexible, resilient, and efficient NLP systems**. Despite their remarkable performance, pre-trained models also present notable limitations when applied to KI-NLP tasks, particularly in areas requiring deep integration of external knowledge and factual consistency.

Additionally, we propose a set of **mathematical models and frameworks** aimed at improving the dependability of PLMs across various languages. To provide a comprehensive view, this work also synthesizes a range of **literature reviews** and surveys recent advancements in **pre-trained language model-based knowledge-enhanced models (PLMKEs)**. Our discussion is structured around three foundational components:

1. **Information sources**
2. **Knowledge-intensive NLP tasks**
3. **Knowledge fusion methods**

Finally, we argue that future NLP models stand to benefit significantly from these integrated approaches—achieving higher **efficiency, accuracy, and precision**, and unlocking a broader spectrum of real-world applications. Only when there is a substantial quantity of common knowledge about the world between the many participants can communication in natural languages take place. An NLP system may tackle a variety of challenging NLP problems by utilizing both general knowledge of the world and specialized knowledge of the particular topic. We have demonstrated in this chapter how such knowledge may be used in practice to resolve syntactic and semantic problems and draw the necessary conclusions. We have outlined the algorithms for these solutions, directed the reader to some of the most well-executed systems, and highlighted some of the most recent advances and open research questions in this area.

REFERENCES

1. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
2. Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving Language Understanding by Generative Pre-Training. OpenAI blog, page 12.
3. Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. nature, 521(7553):436–444.
4. ZIZHENG JI 1, LIN DAI 1, JIN PANG 2, AND TINGTING SHEN," Leveraging Concept-Enhanced Pre-Training Model and Masked-Entity Language Model for Named Entity Disambiguation", Received April 7, 2020, accepted April 24, 2020, date of publication May 12, 2020, date of current version June 9, 2020.Digital Object Identifier 10.1109/ACCESS.2020.2994247.
5. Zhilin Yang 1, Zihang Dai 12, Yiming Yang1, Jaime Carbonell1, Ruslan Salakhutdinov1, Quoc V. Le," XLNet: Generalized Autoregressive Pretraining for Language Understanding", 19 Jun 2019
6. Wang, Xiaozhi & Gao, Tianyu & Zhu, Zhaocheng & Liu, Zhiyuan & Li, Juanzi & Tang, Jian. (2019). KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation.
7. Hai Liu,1,2 Yuanxia Liu,1 Leung-Pun Wong,3 Lap-Kei Lee ,3 and Tianyong Hao," A Hybrid Neural Network BERT-Cap Based on Pre-Trained Language Model and Capsule Network for User Intent Classification", Volume 2020, Article ID 8858852, 11 pages https://doi.org/10.1155/2020/8858852
8. Schomacker, T.; Tropmann-Frick, M. Language Representation Models: An Overview. Entropy 2021, 23, 1422. https:// doi.org/10.3390/e23111422
9. Munazza Zaib, Quan Z. Sheng, and Wei Emma Zhang. 2020. A Short Survey of Pre-trained Language Models for Conversational AI-A New Age in NLP. In Proceedings of the Australasian Computer Science Week Multiconference (ACSW 2020), February 4–6, 2020, Melbourne, VIC, Australia. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3373017.3373028.
10. Bonan Min , Hayley Ross , Elior Sulem , Amir Pouran Ben Veyseh ,Thien Huu Nguyen, Oscar Sainz , Eneko Agirre , Ilana Heinz , and Dan Roth, "Recent Advances in Natural Language Processing via Large Pre-Trained Language Models: A Survey".
11. Da Yin1∗ , Li Dong2 , Hao Cheng2 , Xiaodong Liu2 , Kai-Wei Chang1 , Furu Wei2 , Jianfeng Gao2, "A Survey of Knowledge-Intensive NLP with Pre-Trained Language Models".
12. Hai Liu,1,2 Yuanxia Liu,1 Leung-Pun Wong,3 Lap-Kei Lee ,3 and Tianyong Hao, "A Hybrid Neural Network BERT-Cap Based on Pre-Trained Language Model and Capsule Network for User Intent Classification", Volume 2020, Article ID 8858852, 11 pages https://doi.org/10.1155/2020/8858852
13. Jian Yang, Gang Xiao, Yulong Shen, Wei Jiang, Xinyu Hu, Ying Zhang, Jinghui Peng, "A Survey of Knowledge Enhanced Pre-trained Models".
14. Junyi Li1,3† , Tianyi Tang2† , Wayne Xin Zhao1,3∗ and Ji-Rong Wen1,2,3 1 Gaoling, "Pretrained Language Models for Text Generation: A Survey",2021.
15. Raghavendra Rao Althar , Debabrata Samanta , Manjit Kaur Abeer Ali Alnuaim, Nouf Aljaffan, and Mohammad Aman Ullah , "Software Systems Security Vulnerabilities Management by Exploring the Capabilities of Language Models Using NLP", Computational Intelligence and Neuroscience Volume 2021, Article ID 8522839, 19 pages https://doi.org/10.1155/2021/8522839

16. Bonan Min , Hayley Ross , Elior Sulem , Amir Pouran Ben Veyseh ,Thien Huu Nguyen4, Oscar Sainz, Eneko Agirre, Ilana Heinz, and Dan Roth, “Recent Advances in Natural Language Processing via Large Pre-Trained Language Models: A Survey”.
17. Xiaozhi Wang , Tianyu Gao , Zhaocheng Zhu , Zhengyan Zhang , Zhiyuan Liu , Juanzi Li , and Jian Tang , “KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation”.
18. Xipeng Qiu*, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai & Xuanjing Huang, “Pre-trained Models for Natural Language Processing: A Survey”.
19. Adithya V Ganesan, Matthew Matero, Aravind Reddy Ravula, Huy Vu, H. Andrew Schwartz,” Empirical Evaluation of Pre-trained Transformers for Human-Level NLP: The Role of Sample Size and Dimensionality”, Proc Conf. 2021 June ; 2021: 4515–4532. doi:10.18653/v1/2021.naacl-main.357.
20. Anthony Gillioz,” Overview of the Transformer-based Models for NLP Tasks”, Proceedings of the Federated Conference on Computer Science and Information Systems pp. 179–183 DOI: 10.15439/2020F20 ISSN 2300-5963 ACSIS, Vol. 21
21. Cong Sun1, Zhihao Yang1*, Lei Wang2*, Yin Zhang2, Hongfei Lin1 and Jian Wang1,” Deep learning with language models improves named entity recognition for PharmaCoNER”, Sun et al. BMC Bioinformatics (2021) 22:602 https://doi.org/10.1186/s12859-021-04260-y
22. Fang Liu, Ge Li∗, Yunfei Zhao, Zhi Jin∗,” Multi-task Learning based Pre-trained Language Model for Code Completion”, 29 Dec 2020
23. Hai Liu,1,2 Yuanxia Liu,1 Leung-Pun Wong,3 Lap-Kei Lee ,3 and Tianyong Hao,” A Hybrid Neural Network BERT-Cap Based on Pre-Trained Language Model and Capsule Network for User Intent Classification”, Volume 2020, Article ID 8858852, 11 pages https://doi.org/10.1155/2020/8858852
24. Jia Tracy Shen, Michiharu Yamashita, Ethan Prihar, Neil Heffernan, XintaoWu, Ben Graff, Dongwon Lee,” MathBERT:A Pre-trained Language Model for General NLP Tasks in Mathematics Education”, 35th Conference on Neural Information Processing Systems (NeurIPS 2021), Workshop on Math AI for Education (MATHAI4ED)
25. Junyi Li1;3y , Tianyi Tang2y , Wayne Xin Zhao1;3———————————— and Ji-Rong Wen,” Pretrained Language Models for Text Generation: A Survey”, Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21) Survey Track
26. Linqing Liu,” Towards Elective Utilization ofPretrained Language Models”, 2020
27. Ming Zhou, Nan Duan, Shujie Liu, Heung-Yeung Shum,” Progress in Neural NLP: Modeling, Learning, and Reasoning”, Received 30 April 2019 Revised 30 August 2019 Accepted 13 October 2019 Available online 7 January 2020
28. Nankai Lin, Yingwen Fu, Chuwei Chen, Ziyu Yang, Shengyi Jiang,” LaoPLM: Pre-trained Language Models for Lao”,2020.
29. Wikipedia, n-Gram, https://en.wikipedia.org/wiki/N-gram, 2021.
30. Xiaozhi Wang , Tianyu Gao , Zhaocheng Zhu , Zhengyan Zhang , Zhiyuan Liu , Juanzi Li , and Jian Tang , “KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation”.
31. Patrick S. H. Lewis, Ethan Perez, et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In NeurIPS, 2020.
32. Tom Kwiatkowski, Jennimaria Palomaki, et al. Natural Questions: A Benchmark for Question Answering Research. TACL, 2019.
33. Zhilin Yang, Peng Qi, et al. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In EMNLP, 2018.
34. Lewis, R. L. 1993b. An architecturally-based theory of human sentence comprehension. PhD thesis, Carnegie Mellon University, Computer Science Department, Pittsburgh, PA. Tech. Report. CMU-CS-93-226.

35. Mahesh, K. 1995. Syntax-semantics interaction in sentence understanding. PhD Thesis, College of Computing, Georgia Institute of Technology, Atlanta, GA. Technical Report GIT-CC-95/10
36. Birnbaum, L. and Selfridge, M. 1981. Conceptual analysis of natural language. In Inside Computer Understanding, ed. R. Schank and C. Riesbeck, p. 318-353. Lawrence Erlbaum Associates.
37. Burton, R. 1976. Semantic grammar: An engineering technique for constructing natural language understanding systems. BBN Report No. 3453, Bolt, Beranek, and Newman, Cambridge, MA.
38. Cardie, C. and Lehnert, W. 1991. A cognitively plausible approach to understanding complex syntax. In Proceedings of the Ninth National Conference on Artificial Intelligence, p. 117-124. Morgan Kaufmann.
39. Carlson, L. and Nirenburg, S. 1990. World Modeling for NLP. Technical Report CMU-CMT-90-121, Center for Machine Translation, Carnegie Mellon University, Pittsburgh, PA.
40. T. Young, D. Hazarika, S. Poria, and E. Cambria. 2018. Recent trends in deep learning based natural language processing. *IEEE Computational Intelligence Magazine* 13, 3 (2018), 55–75.