Automated Construction of Knowledge-Bases for Safety Critical Applications: Challenges and Opportunities

Amit Bhatia[a], Alessandro Pinto[b]

[a]*Raytheon Technologies Research Center, 2855 Telegraph Ave, Ste 410, Berkeley, CA, USA*
[b]*Raytheon Technologies Research Center, 2855 Telegraph Ave, Ste 410, Berkeley, CA, USA*

Abstract

Creation of machine-usable, high-quality knowledge-bases is a critical prerequisite for many important applications that rely on availability of high-level of autonomous decision-making and reasoning capabilities. Manual construction of knowledge-bases for complex applications is a time-consuming and expensive process. In such application domains, however, a vast amount of knowledge is available in human-readable format, and it could be leveraged to build knowledge-bases automatically. Natural Language Processing (NLP)-based techniques provide an attractive option for this process. The field of NLP has made rapid strides in last several years and resulted in increased usage across a variety of consumer-facing applications. However, their usage for knowledge-base construction in the aviation industry remains rather limited to date. We present our assessment of using various NLP-based tools for the creation of aviation-focused, high-quality, machine-processable, and human-legible knowledge bases (KBs) for various applications. We identify several gaps, both at the application and fundamental levels, and also identify potential directions for future research that could help overcome the challenges.

Keywords

Knowledge Acquisition, Knowledge Representation, Knowledge Fusion, Autonomy, Aviation

1. Introduction

Fueled by the fast-paced advances in the field of Artificial Intelligence (AI), in particular Machine Learning (ML), autonomous and intelligent systems are becoming pervasive in a broad range of applications, from tailored advertisements and suggestions, shaping the on-line user experience, to complex cyber-physical systems such as autonomous cars. These systems interact with the environment and the humans, understand the current situation and plan actions to achieve goals as commanded by other agents or predefined at design time. While no industrial sector seems to be able to resist the allure of delivering "intelligent" products and services, and realizing the potential economic benefits that AI may bring to their businesses, aerospace and defense have been slow adopters, playing the role of observers, still hovering around research, prototyping and small-scale demonstrations. We believe that the main reasons are operational complexity and stringent safety requirements. Here, we specifically refer to operational autonomy, namely the ability of an embedded system to replace humans in the execution of a mission that involves several cyber-physical platforms and a mix of physical and non-physical actions. For example, replacing pilots to bring passengers from one airport to another is a complex mission that involves executing pre-flight operations, pushing off the gate,

CEUR Workshop Proceedings (CEUR-WS.org)

reaching the runway, taking off, climbing, cruising, descending, landing, and taxiing to the destination gate. Currently, pilots deal with hundreds of tasks [1], including communicating with Air Traffic Control, monitoring the state of the aircraft, and actually flying the vehicle. Most importantly, pilots are critical in dealing with contingencies that require experience, deep understanding of the aircraft and air traffic management, root-cause analysis, and prediction of the outcome of alternative plans. Differently from an autonomous car, an aircraft has a higher level of complexity (comprising hundreds of thousands of components), no easily achievable safe state (when in the air) such as slowing down and stopping, and a higher cost of catastrophic events (hundreds of human lives).

Pilots operating commercial flights benefit from automation functions implemented in sub-systems such as autopilots, instrument landing, and low-level controls for electrical systems, air-management systems, and vehicle health assessment. However, not only these systems may fail or disengage at any time, but during off-nominal situations, their signals need to be integrated to understand the root-cause of a contingency and take appropriate actions. An intelligent system capable of replacing a pilot needs to have the same level of proficiency in both nominal and off-nominal situations. Considering that pilots have trained or flown for thousands of hours, and have studied the inner workings of the machine they are operating, such intelligent system must be equipped with a vast amount of background knowledge that reaches as far as physics. Furthermore, some common-sense knowledge should be taken into account to deal with other situations that are not directly related to flying such as keeping the hundreds of passengers happy and relaxed, or dealing with medical emergencies.

Knowledge base construction (KBC) is the process of populating a database with information from data such as text, tables, images, or video [2, 3]. One approach to construct a knowledge base relies on knowledge engineers and subject-matter experts (e.g., the CYC project [4], WordNet project [5], PaleoBioDB [6]). However, for domains with high complexity, this approach is labor-intensive and error prone [2, 7]. As an example, the creation of the PaleoBioDB knowledge base took nine person-years and a group of 380 scientists [8]. The quality of the result is evaluated by humans through extensive testing and revisions, but completeness is limited to the amount of input data that humans are able to consider given the available development time and cost.

Automated knowledge base construction (AKBC) is another approach where tools are employed to process a potentially much larger set of input sources, at a remarkably higher throughput rate compared to trained domain experts, and generate more comprehensive knowledge bases. As an example, PaleoDeepDive, a DeepDive-based approach for construction of entities and relations from PDF documents was able to process roughly 10x the number of documents, with per-document recall roughly 2.5x that of human annotators [8].

The aviation domain poses a unique set of challenges to automated knowledge base construction due to its stringent safety requirements. The constructed knowledge base cannot only contain correlation data among facts, but for critical decisions, the deduction process from inputs to conclusions must lead to actions that are safe. It is therefore important to revisit the architecture of an automated knowledge base construction pipeline to exploit its ability to be comprehensive while assuring correctness.

1.1. Automated knowledge base construction

AKBC starts from sources that have been created by humans for human consumption including text and tables, and generates knowledge bases that can be used by machines in a variety of applications. The techniques that are typically used in this process include Natural Language Processing (NLP) [9, 10], Natural Language Understanding (NLU) and Machine Reading Comprehension (MRC) [11, 12], whose performance has made remarkable leaps over the past decade as result of advances in big

data, natural language processing, and machine learning technologies [7, 2]. KnolwedgeVault [13], DeepDive [8], MinIE [14], NELL [15], Alexandria [16], Fonduer [3] is a non-exhaustive list of well-developed AKBC tools, some of which are publicly available. These tools, together with others, have been used to build large, high-quality knowledge bases (KBs) such as Freebase [17], YAGO [18], IBM Watson [19], PharmGKB [20], and Google Knowledge Graph [21]. Complete applications have been developed in domains such as chatbots [22], fake news detection [23], healthcare [24, 25], semantic biomedical resource discovery [26], data-driven materials discovery [27], finance [28], and law [29].
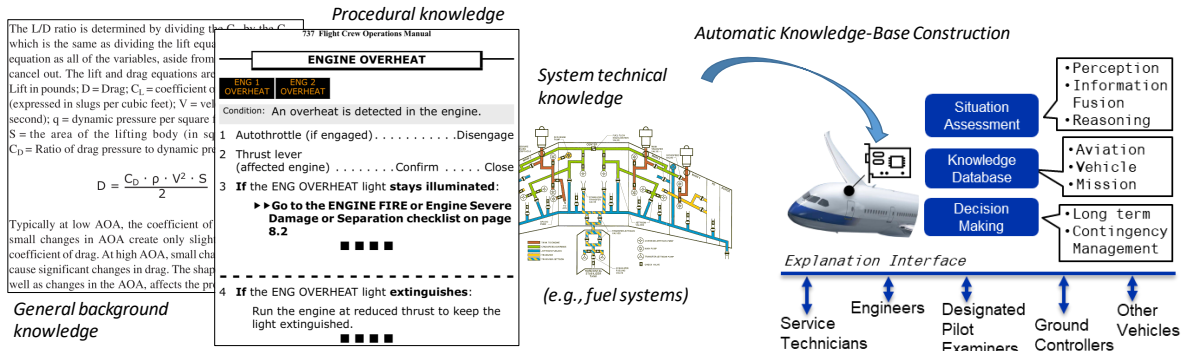
However, many use-cases for the constructed knowledge-bases require a deeper understanding of the world and the rules under which the system and processes that are the subject of the knowledge model operate. For many complex applications, the knowledge acquisition and reasoning processes need to understand and manipulate rich models of the world evoked by textual sources such as causal relations or adherence to given procedures and standards [30, 31, 32, 33].

1.2. Unique challenges in automated knowledge-base construction for aviation

Automated knowledge-base construction for aviation applications presents a set of unique challenges that are not necessarily present in consumer-facing applications. Consider the example of building an intelligent cyber-agent capable of safely and successfully operating an aircraft during a typical flight from the departure to the arrival gate. Such cyber-pilot (CP) (Figure 1(right)) must be knowledgeable about the vehicle that it is operating, the mission to be performed, the load that the vehicle is carrying (cargo and/or passengers, together with their properties such as value and health conditions), and aviation rules and regulations. The sources of aviation knowledge are vast and varied including flight manuals, maintenance manuals, accident reports, system models encoded as simulators, certification requirements documents, textbooks on flight dynamics, structural mechanics, aerodynamics, etc. (see Figure 1(left) for some examples).

To deploy a CP on commercial passenger vehicles, two other requirements must be fulfilled. First, it must be shown that safety will meet or exceed current standards, with a CP delivering a lower rate of mishaps than human-piloted aircraft in a variety of scenarios. Such stringent certification requirements are not considered in consumer-facing applications since errors don't typically lead to catastrophic failures, and the decision loop is never closed without human supervision (human-in-the-loop). Certification is a process used to expose risks, and to evaluate whether they are acceptable. Current standards rely on analysis tools, but also heavily on human revisions and inspection. Thus, the second requirement for a CP is transparency or explainability in decision-making. Explainability as a feature is crucial at run-time where a CP may be required to communicate with Air Traffic Control (ATC), technicians, engineers, or other vehicles, and it is not only required to inspect and revise the model that the CP has learned, but also as a mean to identify, isolate, and manage contingencies, which is key in redundant architectures where alternative actions may guarantee safety. The ability to generate explanations goes beyond having a human-understandable model, but requires a knowledge-base that enables abductive reasoning.

To add to the complexity of knowledge acquisition and reasoning, significant amount of knowledge is actually background knowledge that may have been gradually assimilated over a long period of time (e.g., basic laws of mechanics, thermodynamics, laws about cause-effects). Consider for example operational and procedural documents such as [34, 35, 36]. These documents describe how to operate an aircraft, but they assume that the reader possesses a rather large amount of background information. Moreover, these documents do not explain why certain steps should be taken while executing a procedure or a checklist since it is assumed that the causal relation between an action and its effect or precondition are known. Enabling such level of reasoning also seems to require the ability to

Procedural knowledge

System technical knowledge

Automatic Knowledge-Base Construction

The L/D ratio is determined by dividing the C_L by the C_D which is the same as dividing the lift equation... equation as all of the variables, aside from... cancel out. The lift and drag equations are... Lift in pounds; D = Drag; C_L = coefficient of... (expressed in slugs per cubic feet); V = vel... second); q = dynamic pressure per square... S = the area of the lifting body (in sq... C_D = Ratio of drag pressure to dynamic pr...

$$D = \frac{C_D \cdot \rho \cdot V^2 \cdot S}{2}$$

Typically at low AOA, the coefficient of... small changes in AOA create only slight... coefficient of drag. At high AOA, small cha... cause significant changes in drag. The shap... well as changes in the AOA, affects the pr...

General background knowledge

737 Flight Crew Operations Manual

ENGINE OVERHEAT

ENG 1 OVERHEAT   ENG 2 OVERHEAT

Condition: An overheat is detected in the engine.

1 Autothrottle (if engaged) . . . . .Disengage
2 Thrust lever
   (affected engine) . . . . . . .Confirm . . . . . Close
3 **If** the ENG OVERHEAT light **stays illuminated**:
   ▶▶ **Go to the ENGINE FIRE or Engine Severe Damage or Separation checklist on page 8.2**
   ▪ ▪ ▪ ▪

4 **If** the ENG OVERHEAT light **extinguishes**:
   Run the engine at reduced thrust to keep the light extinguished.
   ▪ ▪ ▪ ▪

(e.g., fuel systems)

Situation Assessment
- Perception
- Information Fusion
- Reasoning

Knowledge Database
- Aviation
- Vehicle
- Mission

Decision Making
- Long term
- Contingency Management

Explanation Interface

Service Technicians   Engineers   Designated Pilot Examiners   Ground Controllers   Other Vehicles

Figure 1: Examples of human-readable sources of aviation knowledge (left). Embedded cyber-pilot concept as an open, knowledgeable, explainable system (right).

target knowledge representation languages that are able to capture a typical ontology of classes and logical facts expressed in a formal language such as First Order Logic, and also operators that change the state of the system which could be expressed as actions with preconditions and effects. Extracting such structures is challenging and requires integrating, extending, and developing new AKBC tools.

2. Discussion of underlying technical gaps and potential solution approaches

In this section, we present some challenges faced by current NLP-based tools in the automatic construction of aviation-focused, machine-understandable, high-quality knowledge-bases. We also discuss some potential approaches for overcoming the challenges.

2.1. Aviation-contextualization

Most of the existing state-of-the-art NLP tools (e.g., Torch-T5 text summarization model [37], or BERT [38]) are trained over a given input corpus and benchmarked for specific tasks such as Name-Entity Recognition (NER), or Text Classification. However, it has been shown that accuracy within a given domain can be improved by expanding the input corpus to include sources from that domain (see for example BioBERT [39], SciBERT [40], or ClinicalBERT [41]). Thus, there is a need to create an extensive corpus of aviation text sources that can be used to train language models using a spectrum of techniques from supervised to unsupervised learning. AKBC tools such as DeepDive [8], Alexandria [16], Fonduer [3] use different innovative techniques for reducing the need for labeled data. DeepDive [8] leverages techniques based on distant supervision whereas Fonduer [3] leverages weak-supervision-based approaches through the concepts of matchers and throttlers in the proposed framework. Alexandria [16] supports automated extraction of features. These extracted inputs also need to be customized to be used in the aviation domain.

A domain specific corpus, or other filters and customization inputs to extraction tools alone would perhaps not be sufficient. Many relations used in aviation documents have a precise meaning that is captured by domain specific models, reasoning and simulation engines. Learning these models from a potentially large set of text sources or even data sets seems unnecessary, inefficient, and suboptimal. Consider the following example taken from the emergency descent non-nominal checklist of Boeing 737 [35] - *If structural integrity is in doubt, limit speed as much as possible and avoid high*

maneuvering loads. This safety check refers to the relation between structural integrity, airspeed and the aerodynamic and gravitational loads on an airplane. First, any ambiguity should be removed (see Section 2.3): in this sentence, "speed" refers to "airspeed", and "maneuvering loads" refers to the "gravitational and aerodynamic forces experienced by the fuselage and wings". Secondly, contextualization in this example means relating structural integrity to maximum loads, loads to maximum speed, and finally annotating these relations with their interpretation given precisely by aviation-specific quantitative models, such as physics-based models. The ability to integrate the extracted knowledge (in any form such as logic sentences, knowledge graphs, or databases) with precise domain specific models is clearly important towards delivering a high quality knowledge-base.

Recommendations. The creation of a dedicated corpus for aviation is needed: it should contain sources with a mix of procedural and descriptive knowledge, and a variety of formats (see Section 2.2 for some considerations on how structure carries information). New techniques are also needed to leverage domain specific models and reasoning engines that precisely capture large knowledge fragments. After disambiguating and grounding facts into aviation-specific contexts, predicates and relations need to be mapped to available models. This is not a one-to-one mapping because the same model (or even a combination of models) could be used as interpretation of many predicates. Domain specific tools typically require a set of inputs and parameters that will need to be synthesized from qualitative statements, or computed by other models. The quantitative results from these models will then need to be lifted back into the knowledge base. Finally, the execution of the various reasoning engines and models will also need to be orchestrated (see Section 2.4).

2.2. Information in Structure, and Structure of Information

Human-readable documents are typically written following a set of conventions, relying extensively on formatting and highlighting. In many cases, such conventions are also typically explained at the beginning of a document. Humans rely critically on this type of structure for efficiency and to avoid ambiguities. Removing this structure from the document would certainly reduce its legibility. More importantly (and relevant to the knowledge extraction process), *the semantics of relations between entities depends on the structure.* For example, the relation between a component and a numeric quantity in [36] is a limitation if found in a chapter starting with the letter "L", and a desired setting if found in a chapter starting with "NP" (Normal Procedures).

Another example of structure can be found in the procedures for stall recovery in [36], which are organized in a two-column format: one for the pilot flying the aircraft, and the other for the co-pilot who has a co-monitoring role. In this case, structure can be leveraged in the extraction process as the left column defines a sequence of actions, while the right column specifies the important quantities that need to be monitored. Ignoring the structure would miss the opportunity to use dedicated extraction tools, or knowledge representation languages for the two columns that may instead result in a more accurate and efficient knowledge base.

Among existing tools for AKBC, Fonduer [3] is a machine-learning-based approach that constructs relations from richly formatted multi-modal human-readable data sources. This approach goes in the right direction: it uses a model for the structure of a document that allows to keep track of where an entity or a relation is mentioned, and it allows users to provide input schema and filters which could be used to accommodate different document structures as decided by the authors of reports or manuals. The model could be expanded to include conventions typically found in aviation manuals such as capitalization, lists, and indentation that are used to describe how to operate a machine.

Once a document is processed from the formatting standpoint, the information needs to be organized such that one can reason over the encoded knowledge. We refer to this as Structure of Informa-

tion. One view of such structure is parts of speech (POS) which provides a grammatical understanding of paragraphs, sentences and words [42]. Another view is the logical and ontological structure contained in the text [43, 44, 45, 46]. POS tagging is relatively well-developed, and mature tools exist today for such task. However, extracting logical structures from raw sources such as text is less mature [43, 44, 45, 46]. Having a logical representation enables reasoning by deduction and abduction which are both important in situational assessment, decision-making, and root-cause analysis. Moreover, POS tagging alone cannot be used to resolve semantic ambiguity, where the same word may have two completely different roles and meaning in a sentence. A logical representation would instead enable disambiguation by reasoning and elimination of hypotheses. We also believe that the types of structures to be extracted are not only logical connectives such as And-Or patterns, but also include causal and temporal dependencies among events, hidden states, action models, and performance curves. As an example, the non-normal checklists, flight patterns and maneuvers described in [36] are designed to be read and acted upon sequentially (and often in time-constrained, emergency situations). The ordering of actions (e.g., changing various settings) have a drastic impact on the eventual outcome. There have been some attempts at extracting procedural information from text [47, 48, 49, 50] but these tools are not mature enough and the quality of the resulting knowledge-base for aviation applications needs to be investigated.

Recommendations. As a first step, we recommend performance evaluation of Fonduer and similar frameworks in identifying the structure present in aviation corpora discussed in Section 2.1. As a next step, we suggest extension of existing AKBC approaches to move beyond document-level structure and towards aviation-level hierarchical organization of concepts - for example, Aircraft $\rightarrow$ Boeing $\rightarrow$ 737-800. In effect, we are essentially proposing a path forward for extending AKBC approaches to go from Information in Structure towards Structure of Information. Creation of a large-scale aviation ontology would help in defining and organizing the structure of information and ontology learning techniques from text could help here [51]. We also recommend further research to extend current tools with capabilities to extract events, causal and temporal relationships, and action models to be used in decision-making, plan verification, and impact analysis.

2.3. Ambiguity in natural language and the issue of background vs. explicitly stated knowledge

Humans are incredibly good at attaching context to text and overcoming ambiguities in text through a deep understanding of rules of the world. However, it is highly non-trivial to replicate and automate this process [52]. The challenge in resolving ambiguity points to the underlying issue of background (unstated) vs. foreground (explicitly stated) knowledge. We as human readers make use of an incredible amount of background knowledge when understanding and attaching context to the meaning of words, sentences and paragraphs.

Common problems related to the ambiguous use of words in natural languages are less prevalent in the aviation domain as manuals and reports are written in such a way to make sure that the messages cannot be misinterpreted by humans. Clearly, a first step has to be taken towards removing some common potential ambiguity that may arise. For example, when processing a set of sentences such as *Every airplane has two wings; Boeing 747 has four engines; Every wing is a part; Every engine is a part*, it is important to identify "four" as the same as the number 4. Also, "Boeing 747" should be identified as an airplane (perhaps by processing other documents that mention such a fact). Semantically, it is also important to establish whether the closed or open world assumption [53, 54] is used since a fact such as `Boeing 747 has 6 parts` may or may not be inconsistent with the description above.

A more serious issue, however, is the semantic ambiguity of words and relations that, when in-

terpreted by a reader with enough background knowledge, have instead an unambiguous meaning. Consider for example a fragment of the Emergency Descent non-normal checklist for the Boeing 747 [36]: *Without delay, descend to the lowest safe altitude, or 10,000 feet, whichever is higher.* Clearly, delay in this case may refer to the time to start the descent procedure, or to reach the prescribed altitude (see Section 2.1 for another example). Context and background knowledge should be leveraged to resolve ambiguity. From this standpoint, contextualization is necessary but not sufficient. Reasoning and knowledge extraction should be interleaved to incrementally reduce semantic uncertainty.

Currently, no mature techniques exist to guide the separation between foreground and background knowledge [55]. Accompanying this challenge is the issue of innate knowledge (e.g., intuition about basic laws of physics) [56, 57], and it is not clear how to incorporate such knowledge within the AKBC process.

Recommendations: We recommend investigating the possibility of combining ideas from Controlled Natural Languages (CNLs) with existing AKBC frameworks. CNLs [52, 58, 59] help to partially overcome ambiguity present in textual sources written for human consumption. CNLs are subset of natural languages (e.g., English) and have well-defined formal semantics. However, the use of CNLs requires expert understanding of the CNL itself and hence encoding is still non-trivial, and challenges remain [58]. Hybrid CNLs that combine ideas from formal logic with CNL, e.g., Knowledge Authoring Logic Machine (KALM) [60, 61] hold more promise.

On the issue of discovering missing background knowledge, or to reduce semantic uncertainty, we recommend investigating approaches for combining abductive, deductive and inductive reasoning techniques with iterative learning [62, 63, 55].

2.4. Heterogeneity of input domains and tight coupling between encoding and reasoning

Knowledge-base creation is an iterative process. The ability to encode knowledge efficiently depends not only on the framework being used to encode knowledge facts, but also on availability of solvers that can test the encoded knowledge fragments for errors. It is rarely the case that a first attempt results in a correct and useful knowledge-base. Our experience indicates that the processes of encoding knowledge fragments and reasoning over them are in fact tightly interlinked.

There are a multitude of encoding and reasoning approaches, each designed or natural for processing a specific type of knowledge. We can distinguish two orthogonal ways of specializing encoding and reasoning: algorithmic domains and application domains. A variety of languages exist to model knowledge within algorithmic domains. For example, the Plan Domain Definition Language (PDDL) [64] focuses on encoding decision-making problems, while temporal logics [65] are more suitable for modeling and reasoning about the temporal relations among events. Similarly, knowledge graphs [21] are particularly efficient for approximate inference using embeddings while Markov Logic Networks [66] are better suited when dealing with knowledge fragments that contain probabilistic logic facts. These representation languages are domain independent and are efficient representations for specific algorithms such as planning, model-checking, or deductive reasoning.

Domain independent reasoning, however can become inefficient. In many cases, both the encoding and the reasoning algorithms can be specialized to particular domains where only certain queries are of interest. For example, the language, concepts and machinery used to model and solve computational fluid mechanics problems are different from the ones used in structural mechanics, which are in turn different from the ones used in dynamics and control.

No principled technique exist currently for a systematic orchestration of knowledge elicitation, encoding, fusion and testing of such multitude of algorithmic and applications domains [67]. We

note here that frameworks such as DeepDive [8] and Fonduer [3] focus on the multi-modal aspect of knowledge contained in the input sources, but they have not been stress-tested in the case of heterogeneous knowledge domains.

Recommendations: There have been some recent advances in the area of knowledge graphs and semantic web that try to achieve knowledge fusion by leveraging ideas used for data fusion [68]. Data-fusion inspired knowledge-fusion techniques [68], together with specialized domain solvers (e.g., [69] for combinatorial reasoning) could provide a useful starting point towards solving the knowledge fusion problem in general for aviation and autonomy applications. More advanced techniques could be borrowed from the theorem proving community that has been very active in defining general ways of combining theories and solvers as in the case of Satisfiability Modulo Theories [70]. In addition, development of interface languages, translators, and reasoners to orchestrate different solvers and cross the boundaries of different domains would also be needed.

2.5. Proficiency testing

As we mentioned in Section 2.1, a crucial gap in existing AKBC frameworks is the limited ability to define large-scale automated tests that can be used during the AKBC process. Existing benchmarks, for example those that are used in assessing performance of various NLP tools fall well short of testing real-world understanding [31]. Effective and scalable proficiency testing techniques are required in AKBC process for multiple reasons: (1) to catch non-obvious errors during the initial construction of knowledge-bases, (2) to resolve conflicts during the run-time update process when new information about the external world needs to be incorporated in the knowledge-base, and (3) to enable human-legibility and certification which would rely on demonstrating and explaining the understanding of aviation knowledge through performance on designed tests [71, 72, 73, 74].

Recommendations: We recommend borrowing ideas from formal methods and software testing communities [75] for expanding the current suite of tests used in various NLP and AKBC frameworks [76, 77, 31]. It is expected that the most effective proficiency testing approaches would be those that combine manually designed tests with those that have been instantiated from templates (through concretization to specific contexts).

3. Conclusions

The introduction of autonomy in the management of complex systems and their operations requires developing knowledge-bases that can be used for reasoning and decision-making. Manual construction of these knowledge-bases is inefficient, requiring multi-year efforts and hundreds of contributors. Thus, Automatic Knowledge Base Construction (AKBC) techniques are very much sought in related applications domains. Recent advances in Machine Learning have enabled the development of tools for Natural Language Processing, Natural Language Understanding, and Machine Reading Comprehension that show promise in ingesting raw sources such as text and tables to create knowledge-bases. The aviation industry could benefit from these new technologies, but the resulting knowledge-bases must satisfy stringent assurance requirements that support certification processes. We believe that an AKBC system for these kinds of applications requires the integration and enhancement of several techniques and that it is yet to be developed. The aviation context needs to be captured by a corpus of relevant documents and additional inputs such as extraction rules. Additional context should be injected as interpretation of relations provided by analytic and simulation models. The structure of aviation documents should be exploited in the extraction process to provide the right semantics to parts of speech in different paragraphs. The extracted information should be encoded in languages

that enable formal reasoning to further reduce ambiguity and noise. The encoded knowledge should be fused with background knowledge and a variety of domain-specific reasoning engines should be harmonized to lead to a high-quality knowledge-base. Finally, the constructed knowledge-base should undergo rigorous proficiency testing to provide assurance and refine its content.

While several gaps exist, we believe that the current set of technologies are promising and that a concrete research roadmap could be developed to construct high-assurance knowledge bases for safety critical applications.

References

[1] P. Schutte, Task analysis of two crew operations in the flight deck: Investigating the feasibility of using single pilot, in: 19th International Symposium on Aviation Psychology, 2017, p. 566.

[2] G. Weikum, M. Theobald, From information to knowledge: harvesting entities and relationships from web sources, in: Proceedings of the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, 2010, pp. 65–76.

[3] S. Wu, L. Hsiao, X. Cheng, B. Hancock, T. Rekatsinas, P. Levis, C. Ré, Fonduer: Knowledge base construction from richly formatted data, in: Proceedings of the 2018 International Conference on Management of Data, 2018, pp. 1301–1316.

[4] D. B. Lenat, Cyc: A large-scale investment in knowledge infrastructure, Communications of the ACM 38 (1995) 33–38.

[5] G. A. Miller, WordNet: An electronic lexical database, MIT press, 1998.

[6] Behrensmeyer, A. K., and A. Turner, Taxonomic occurrences of suidae recorded in the paleobiology database, 2013. URL: http://fossilworks.org.

[7] O. Din, Towards a flexible system architecture for automated knowledge base construction frameworks, in: 2019 IEEE International Conference on Big Data (Big Data), IEEE, 2019, pp. 3066–3071.

[8] C. Zhang, C. Ré, M. Cafarella, C. De Sa, A. Ratner, J. Shin, F. Wang, S. Wu, Deepdive: Declarative knowledge base construction, Communications of the ACM 60 (2017) 93–102.

[9] D. Khurana, A. Koli, K. Khatter, S. Singh, Natural language processing: State of the art, current trends and challenges, arXiv preprint arXiv:1708.05148 (2017).

[10] H. Li, Deep learning for natural language processing: advantages and challenges, National Science Review (2017).

[11] P. M. Nadkarni, L. Ohno-Machado, W. W. Chapman, Natural language processing: an introduction, Journal of the American Medical Informatics Association 18 (2011) 544–551.

[12] KDNuggets, NLP vs. NLU: from understanding a language to its processing, 2019. URL: https://www.kdnuggets.com/2019/07/nlp-vs-nlu-understanding-language-processing.html.

[13] X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, W. Zhang, Knowledge vault: A web-scale approach to probabilistic knowledge fusion, in: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, 2014, pp. 601–610.

[14] K. Gashteovski, R. Gemulla, L. d. Corro, MinIE: minimizing facts in open information extraction, Association for Computational Linguistics, 2017.

[15] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, B. Yang, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, et al., Never-ending learning, Communications of the ACM 61 (2018) 103–115.

[16] J. Winn, J. Guiver, S. Webster, Y. Zaykov, M. Kukla, D. Fabian, Alexandria: Unsupervised high-

precision knowledge base construction using a probabilistic program, in: Automated Knowledge Base Construction (AKBC), 2018.

[17] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, J. Taylor, Freebase: a collaboratively created graph database for structuring human knowledge, in: Proceedings of the 2008 ACM SIGMOD international conference on Management of data, 2008, pp. 1247–1250.

[18] A. Amarilli, L. Galárraga, N. Preda, F. M. Suchanek, Recent topics of research around the YAGO knowledge base, in: Asia-Pacific Web Conference, Springer, 2014, pp. 1–12.

[19] D. Ferrucci, E. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A. A. Kalyanpur, A. Lally, J. W. Murdock, E. Nyberg, J. Prager, et al., Building Watson: An overview of the DeepQA project, AI magazine 31 (2010) 59–79.

[20] M. Hewett, D. E. Oliver, D. L. Rubin, K. L. Easton, J. M. Stuart, R. B. Altman, T. E. Klein, PharmGKB: The pharmacogenetics knowledge base, Nucleic acids research 30 (2002) 163–165.

[21] Google, Google knowledge graph, 2012. URL: http://googleblog.blogspot.co.uk/2012/05/introducing-knowledge-graph-things-not.html.

[22] M. Mnasri, Recent advances in conversational nlp: Towards the standardization of chatbot building, arXiv preprint arXiv:1903.09025 (2019).

[23] R. Oshikawa, J. Qian, W. Y. Wang, A survey on natural language processing for fake news detection, arXiv preprint arXiv:1811.00770 (2018).

[24] O. G. Iroju, J. O. Olaleke, A systematic review of natural language processing in healthcare, International Journal of Information Technology and Computer Science 8 (2015) 44–50.

[25] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, J. Dean, A guide to deep learning in healthcare, Nature medicine 25 (2019) 24–29.

[26] P. Sfakianaki, L. Koumakis, S. Sfakianakis, G. Iatraki, G. Zachariudakis, N. Graf, K. Marias, M. Tsiknakis, Semantic biomedical resource discovery: a natural language processing framework, BMC medical informatics and decision making 15 (2015) 77.

[27] J. M. Cole, A design-to-device pipeline for data-driven materials discovery, Accounts of Chemical Research 53 (2020) 599–610.

[28] I. E. Fisher, M. R. Garnsey, M. E. Hughes, Natural language processing in accounting, auditing and finance: A synthesis of the literature with a roadmap for future research, Intelligent Systems in Accounting, Finance and Management 23 (2016) 157–214.

[29] L. Robaldo, S. Villata, A. Wyner, M. Grabmair, Introduction for artificial intelligence and law: special issue "natural language processing for legal texts", 2019.

[30] G. Marcus, E. Davis, Rebooting AI: Building artificial intelligence we can trust, Vintage, 2019.

[31] J. Dunietz, G. Burnham, A. Bharadwaj, J. Chu-Carroll, O. Rambow, D. Ferrucci, To test machine comprehension, start by defining comprehension, arXiv preprint arXiv:2005.01525 (2020).

[32] M. G. Dyer, In-Depth Understanding. A Computer Model of Integrated Processing for Narrative Comprehension., Technical Report, Yale Univ New Haven CT Dept Of Computer Science, 1982.

[33] H. Jung, J. Allen, N. Blaylock, W. de Beaumont, L. Galescu, M. Swift, Building timelines from narrative clinical records: initial results based-on deep natural language understanding, in: Proceedings of BioNLP 2011 workshop, 2011, pp. 146–154.

[34] Federal Aviation Administration (FAA), Aviation handbooks & manuals, 2020. URL: https://www.faa.gov/regulations_policies/handbooks_manuals/aviation/.

[35] B. Bulfer, Boeing 737 quick reference handbook, 2020. URL: http://www.cockpitcompanion.com/cat-quick.cfm.

[36] The Boeing Company, Boeing 737 operations manual, 1997.

[37] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, arXiv preprint

arXiv:1910.10683 (2019).

[38] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional trans-formers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[39] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, BioBERT: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (2020) 1234–1240.

[40] I. Beltagy, K. Lo, A. Cohan, SciBERT: A pretrained language model for scientific text, arXiv preprint arXiv:1903.10676 (2019).

[41] E. Alsentzer, J. R. Murphy, W. Boag, W.-H. Weng, D. Jin, T. Naumann, M. McDermott, Publicly available clinical BERT embeddings, arXiv preprint arXiv:1904.03323 (2019).

[42] C. Manning, H. Schutze, Foundations of statistical natural language processing, MIT press, 1999.

[43] J. Berant, A. Chou, R. Frostig, P. Liang, Semantic parsing on freebase from question-answer pairs, in: Proceedings of the 2013 conference on empirical methods in natural language processing, 2013, pp. 1533–1544.

[44] P. Yin, G. Neubig, TRANX: A transition-based neural abstract syntax parser for semantic parsing and code generation, arXiv preprint arXiv:1810.02720 (2018).

[45] V. Basile, E. Cabrio, C. Schon, KNEWS: Using logical and lexical semantics to extract knowledge from natural language, in: Proceedings of the European conference on artificial intelligence (ECAI) 2016 conference, 2016.

[46] H. Singh, M. Aggrawal, B. Krishnamurthy, Exploring neural models for parsing natural language into first-order logic, arXiv preprint arXiv:2002.06544 (2020).

[47] F. Maria, C. A. Aguirre, B. M. Anshutz, W. H. Hsu, MATESC: Metadata-analytic text extractor and section classifier for scientific publications., in: KDIR, 2018, pp. 259–265.

[48] H. Yang, C. A. Aguirre, F. Maria, D. Christensen, L. Bobadilla, E. Davich, J. Roth, L. Luo, Y. Theis, A. Lam, et al., Pipelines for procedural information extraction from scientific literature: Towards recipes using machine learning and data science, in: 2019 International Conference on Document Analysis and Recognition Workshops (ICDARW), volume 2, IEEE, 2019, pp. 41–46.

[49] M. Chang, L. V. Guillain, H. Jung, V. M. Hare, J. Kim, M. Agrawala, Recipescape: An interactive tool for analyzing cooking instructions at scale, in: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, 2018, pp. 1–12.

[50] C. X. Chu, G. Weikum, N. Tandon, J. Vreeken, Mining How-to Task Knowledge From Online Communities, Ph.D. thesis, Universität des Saarlandes Saarbrücken, 2016.

[51] R. Lourdusamy, S. Abraham, A survey on methods of ontology learning from text, in: International Conference on Information, Communication and Computing Technology, Springer, 2019, pp. 113–123.

[52] R. Schwitter, Controlled natural languages for knowledge representation, in: Coling 2010: Posters, 2010, pp. 1113–1121.

[53] R. Reiter, On closed world data bases, in: Readings in artificial intelligence, Elsevier, 1981, pp. 119–140.

[54] G. Bossu, P. Siegel, Saturation, nonmonotonic reasoning and the closed-world assumption, Artificial Intelligence 25 (1985) 13–63.

[55] G. Marcus, Deep learning: A critical appraisal, arXiv preprint arXiv:1801.00631 (2018).

[56] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, S. J. Gershman, Building machines that learn and think like people, Behavioral and brain sciences 40 (2017).

[57] G. Marcus, Innateness, alphazero, and artificial intelligence, arXiv preprint arXiv:1801.05667 (2018).

[58] T. Kuhn, A survey and classification of controlled natural languages, Computational linguistics 40 (2014) 121–170.

[59] T. Gao, Controlled natural languages for knowledge representation and reasoning, in: Technical Communications of the 32nd International Conference on Logic Programming (ICLP 2016), Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

[60] T. Gao, P. Fodor, M. Kifer, High accuracy question answering via hybrid controlled natural language, in: 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI), IEEE, 2018, pp. 17–24.

[61] T. Gao, P. Fodor, M. Kifer, Knowledge authoring for rule-based reasoning, in: OTM Confederated International Conferences" On the Move to Meaningful Internet Systems", Springer, 2018, pp. 461–480.

[62] A. Weichselbraun, P. Kuntschik, A. M. Braşoveanu, Mining and leveraging background knowledge for improving named entity linking, in: Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics, 2018, pp. 1–11.

[63] J. B. Tenenbaum, C. Kemp, T. L. Griffiths, N. D. Goodman, How to grow a mind: Statistics, structure, and abstraction, science 331 (2011) 1279–1285.

[64] P. Haslum, N. Lipovetzky, D. Magazzeni, C. Muise, An introduction to the planning domain definition language, Synthesis Lectures on Artificial Intelligence and Machine Learning 13 (2019) 1–187.

[65] E. A. Emerson, Temporal and modal logic, in: Formal Models and Semantics, Elsevier, 1990, pp. 995–1072.

[66] M. Richardson, P. Domingos, Markov logic networks, Machine learning 62 (2006) 107–136.

[67] X. L. Dong, D. Srivastava, Knowledge curation and knowledge fusion: challenges, models and applications, in: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, 2015, pp. 2063–2066.

[68] X. L. Dong, E. Gabrilovich, G. Heitz, W. Horn, K. Murphy, S. Sun, W. Zhang, From data fusion to knowledge fusion, arXiv preprint arXiv:1503.00302 (2015).

[69] L. De Moura, N. Bjørner, Z3: An efficient SMT solver, in: International conference on Tools and Algorithms for the Construction and Analysis of Systems, Springer, 2008, pp. 337–340.

[70] C. Barrett, C. Tinelli, Satisfiability modulo theories, in: Handbook of Model Checking, Springer, 2018, pp. 305–343.

[71] E. E. Alves, D. Bhatt, B. Hall, K. Driscoll, A. Murugesan, J. Rushby, Considerations in assuring safety of increasingly autonomous systems (2018).

[72] M. Cummings, Adaptation of human licensing examinations to the certification of autonomous systems, in: Safe, autonomous and intelligent vehicles, Springer, 2019, pp. 145–162.

[73] R. A. Clothier, B. I. Williams, T. Perez, et al., Autonomy from a safety certification perspective, in: AIAC18: 18th Australian International Aerospace Congress (2019): HUMS-11th Defence Science and Technology (DST) International Conference on Health and Usage Monitoring (HUMS 2019): ISSFD-27th International Symposium on Space Flight Dynamics (ISSFD), Engineers Australia, Royal Aeronautical Society., 2019, p. 278.

[74] L. Chiticariu, Y. Li, F. Reiss, Transparent machine learning for information extraction: state-of-the-art and the future, EMNLP (tutorial) (2015).

[75] M. Kassab, J. F. DeFranco, P. A. Laplante, Software testing: The state of the practice, IEEE Software 34 (2017) 46–52.

[76] G. Marcus, The next decade in AI: Four steps towards robust artificial intelligence, arXiv preprint arXiv:2002.06177 (2020).

[77] V. Kocijan, T. Lukasiewicz, E. Davis, G. Marcus, L. Morgenstern, A Review of Winograd Schema Challenge Datasets and Approaches, arXiv preprint arXiv:2004.13831 (2020).