THÈSE

présentée pour obtenir le grade de

present to obtain the degree of

Docteur de l'Université de Strasbourg
Doctor of the University of Strasbourg

Spécialité: Informatique

Speciality: Computer Science

Inventive Solutions Retrieval from Patent Documents via Natural Language Processing

Présentée et soutenue par

Xin NI

Thèse soutenue publiquement le 17/12/2021
devant le jury composé de

Directeur de thèse:	Prof. Denis Cavallucci	Professeur, INSA Strasbourg
Encadrant de thèse:	Dr. Ahmed Samet	Maître de conférences, INSA Strasbourg
Rapporteur externe:	Prof. Cecilia Zanni-Merk	Professeur, INSA Rouen
Rapporteur externe:	Prof. Hervé Panetto	Professeur, Université de Lorraine
Examinateur:	Prof. Pierre Collet	Professeur, Université de Strasbourg
Examinateur:	Prof. Florence le Ber	IGPEF HDR, Ingénieure des ponts et forêts
Examinateur:	Dr. Magali Barbaroux	Responsable de programmes de recherche, Sartorius
Invité:	Dr. Marion Moliner	Data science manager, Data Excellence Factory, Hager Group

Acknowledgements

Time flies. I come to the end of my Ph.D. journey. Three and a half years of my Ph.D. research life is not short for me. Much difficult time, struggling time, and exciting time for pursuing this Ph.D. degree all appear in my mind now. During the Covid-19 period, many things become more difficult and many working habits have been changed. I had a struggling experience in this period, but all of these have been fortunately passed away. During these years, I received much help from my supervisors, colleagues, friends, families, and China Scholarship Council (CSC). I would like to sincerely thank them here.

First of all, I would like to especially appreciate the financial support from CSC. They let me continue my research work in the team of Design, Information System and Inventive Processes (CSIP) at the National Institute of Applied Science (INSA) of Strasbourg, and the Engineering Science, Computer Science and Imaging Laboratory (ICube) in the University of Strasbourg during these years without any finical concerns. Besides, I would like to express my sincere gratitude to my supervisor Prof. Denis Cavallucci, for his continuous support of my Ph.D. research. His motivation and enthusiasm encourage me all the time. He also provided me many opportunities to attend well-known industrial projects and hugely improve my research works and English expression skills with the cooperation of these industrial partners.

Secondly, I would like to huge thank my thesis co-supervisor, Dr. Ahmed Samet. With the numerous significant help of Dr.Ahmed Samet, I can eventually reach the current research achievements. He opened my mind to enter the Artificial Intelligence (AI), Natural Language Processing (NLP) field. His guidance helped me in all the time of research and writing of the thesis. His continuous engagement, invaluable insights, and careful inspection for each detail during my Ph.D. study make me possible to arrive at the end of my Ph.D. journey. He tirelessly pushed me during these years to reach the next achievement. I have greatly benefited from his supervision and help. Thank you for all your great discussions and guidance!

In addition, I would like to express my appreciation to the great colleagues of the CSIP research team. They provide me with much help on my research works. They are Dr. Masih Hanifi, Dr. Pei Zhang, Ph.D. Guillaume Guarino, Ph.D. Naser Ghannad, Ph.D. Daria Berdyuigina. I will remember all of your help forever. More importantly, I want to thank two of my Chinese friends. They are Ph.D. Zhifeng Guo in Business School of the University of Kent, United Kingdom, and Wentai Wang, a back-end development engineer in ByteDance (well-known due to TikTok) Ltd. During the first two years of my Ph.D. journey, they all helped me several times when I was struggling with my research works. With jet lag, they were working after midnight for helping me address my own issues without any complaints. Their significant help lets me could conquer several challenges in urgent situations. I sincerely thank them.

Moreover, I would like to thank my friends, Dr. Fujiao Tang, Dr. Guixian Liu, Dr. Qiushi Cao, Dr. Peng Chao, Dr. Jianyu Chen, Ph.D. Jiang Jing, Ph.D. Lei Ma, Ph.D. Lei He, Ph.D. Xuefang Li, Ph.D. Xuan Wu, Ph.D. Chong Wang, Ph.D. Xiang Zhang, Ph.D. Qinglin Deng, Ph.D. Haitao Ge, Ph.D. Yuzhou Sun, Ph.D. Ke Zhang, Jingshu Lu. We had a lot of great time and fun in France. They also helped me to be familiar with French life when I first arrived here. With a lot of their warm-heart help, many problems can be solved throughout my entire Ph.D. period. I will remember all of those valuable moments we experienced.

Last but not the least, I would like to appreciate my parents, Ms. Yan Jing and Mr. Tuanjie Ni. They unconditionally support me and kindly care about me from remote China all the time. They care about my safety, my daily life, my health, and my research work during my stay in France. In special, a super thank you to my girlfriend, Qianqian Jiang. Her self-giving care, support, understanding, patience, and encouragement help me a lot in finishing this not-easy Ph.D. journey. Already two years and a half because of Covid-19, I could not easily go back and spend time with them. I feel very sorry and thank their understanding very much. I hope they could see my success and be proud of me.

This is the end of my Ph.D. life but will be the new start in my life.

Xin Ni
Strasbourg, France
31/10/2021

Abstract

Innovation is a key factor for companies developing products and engaging in continuous progress in a highly competitive market. In recent years, in the context of this growing concern for engineering innovation, the demand for inventive engineering solutions has been increasing rapidly in companies. Besides, a large number of published patent documents from wider domains tend to contain the latest inventive knowledge in the world. Mining this sort of knowledge is a significant way to enable industrial innovation. It is also an important alternative to brace the complex manufacturing challenges.

Nevertheless, it is always a significant challenge for engineers without a broad understanding of different domain knowledge to make full use of the inventive knowledge contained in patent documents. Especially, exploring several patents by an expert rapidly turns to be an arduous task. Theory of Inventive Problem Solving (TRIZ) was proposed to provide a logical approach to enhance creativity. However, its lack of formalization and complex principles generate a huge obstacle to implementing it, even for engineers to understand it.

In order to address the aforementioned challenges, in the thesis, we aim to automate the entire inventive problem-solving process by using patent documents based on Natural Language Processing (NLP) techniques. In particular, we propose four main contributions: i) two similar problem retrieval models called IDM-Similar based on Word2vec neural networks and SAM-IDM based on LSTM neural networks are proposed to retrieve similar problems from different domain patents; ii) a problem-solution matching model named IDM-Matching according to XLNet neural networks is proposed to build connections between problems and solutions in patent documents; iii) an inventive solutions ranking model called PatRIS based on multiple criteria decision analysis approach is proposed to rank potential inventive solutions; iv) a software prototype named PatentSolver combining aforementioned models is developed to provide engineers with a real tool to prepare inventive solutions from patent documents. These models have been evaluated on both benchmark and real-world patent datasets.

Keywords: Natural Language Processing (NLP), Patent Mining, Neural Network, Semantic Similarity Computation, Question Answering System, Multiple Criteria Decision Analysis, Theory of Inventive Problem Solving (TRIZ)

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter provides a general introduction of the thesis realized at the laboratoire des sciences de l'ingénieur, de l'informatique et de l'imagerie (ICube). It starts with a general background performed to set the context of the work. Our research problems are also introduced. Next, the main motivation for conducting this research and our contributions are presented. The thesis structure is eventually introduced in detail.

1.1 General Background

Innovation is a key factor for companies to develop competitive products and feature continual progress in the global competitive arena. Inventive solutions, as a useful component to achieve significant innovation outputs, are sought-after to be used to address tough issues and facilitate inventive R&D activities. In recent years, an increasing number of researchers have been investigating how knowledge cross-fertilization between the different domains of industry might be useful to build inventive solutions and solve complex problems.

Nowadays, with the background of this ever-increasing concern on engineering innovation, the pressure for permanent innovative engineering solutions has also been increasing rapidly for companies (Shirwaiker and Okudan, 2008; Jardim-Goncalves et al., 2011; Smirnov et al., 2013). Besides, exploring broader knowledge fields to achieve innovative inspirations has become a significant alternative to embrace complex challenges during manufacturing (Ni, Samet, and Cavallucci, 2021). However, most companies still rely on engineers' experience or brainstorming among different experts or searching solutions on the internet by manual work to promote R&D activities. Although these methods had once been on the hype at the early time of the internet, they cannot handle the current rise of infinite and permanent renewal of information and data throughout all domains and in various forms.

In addition, as an important part of their strategy, innovative design has become a significant factor for companies to survive in the competition arena (Hao et al., 2019; Renjith, Park, and Kremer, 2020). Product innovation level is more amenable to open manufacturing (Kusiak, 2016). Despite the fact that most engineers have realized the significance of different domains knowledge cross-fertilization for creating and developing products (Whiteside et al., 2009), innovative knowledge is still intrinsically linked to the people who use it (Girodon et al., 2015) and catching up on different domains knowledge has always been difficult.

Since the middle of the 20th century, TRIZ, from its Russian acronym "theory of inventive problem solving" (Chapter 3) was initially proposed by Altshuller (Altshuller, Shulyak, and Rodman, 0040) through analyzing hundreds of thousands of patents all over the world. This approach is now internationally used to improve and facilitate the resolution of technological problems (Altshuller, 1999). TRIZ tools and techniques like 40 innovative principles are used to find innovative solutions to targeted problems. In TRIZ, a problem can be associated with the notion of contradiction. Contradictions are intrinsically contained in engineering problems tacitly or explicitly. They are classified as technical or physical. The contradiction matrix is the most widely known technique employed to resolve contradictions in conjunction with the 40 innovative principles and 39 generic engineering parameters. As shown in Fig. 1.1, a classical TRIZ problem-solving process is presented and the major steps are illustrated as follows.

- The specific target problem is firstly prepared by the user.
- With the help of 39 engineering parameters, TRIZ abstracts the specific problem into a generic problem.
- Find out generic solutions by TRIZ models for the generic problem.
- Transform generic solutions to specific solutions by applying your own knowledge and interpretation.

However, the drawbacks of TRIZ, which are mostly relying on expertise from experienced users, are the lack of formalization and the difficulty of operating TRIZ. Therefore, in recent decades, an increasing number of research works have been proposed to further facilitate problem-solving issues. New techniques and materials are used in the problem-solving field, such as machine learning approaches, patent mining, deep learning methods, or on a broader scale computer-aided innovation.

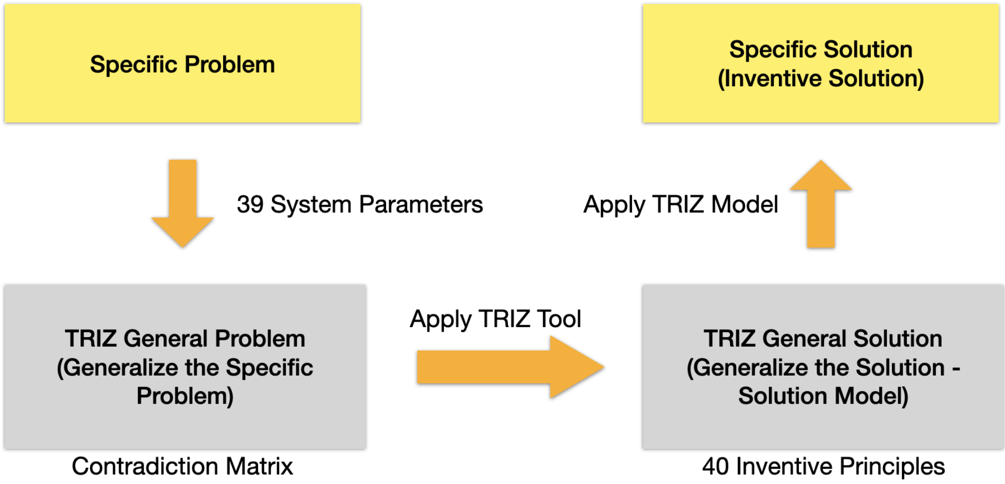


FIGURE 1.1: The problem-solving process of TRIZ

Particularly, compared to other written materials, patent documents play a significant role in containing the latest innovative knowledge in each domain. For instance, the Innovative Design Method (IDM) derived from TRIZ (Sheu, Chen, and Yu, 2012) mainly focuses on patent documents (Ni, Samet, and Cavallucci, 2019). More than 80% of mankind technical knowledge is described in the patent literature (Souili, Cavallucci, and Rousselot, 2015a) and the World Intellectual Property Organization revealed that 90% to 95% of all the world's inventions are found in patent documents (Yeap, Loo, and Pang, 2003). In addition, patent documents are significant intellectual resources to protect the interests of individuals, organizations, and companies (Ni, Samet, and Cavallucci, 2019). They also provide valuable information to solve engineering problems and enhance innovativeness. In a word, innovative knowledge in patents tends to describe the leading edge of problems and their existing potential solutions.

Therefore, to efficiently and effectively use innovative knowledge contained in different patent documents is worth exploring further. Nevertheless, it has always been a significant challenge for engineers without a broad understanding of different domains knowledge to make full use of the inventive information contained in patent documents. Especially, exploring several patents by an expert rapidly turns to be an arduous task. Thus, to automate the entire process of innovative solutions retrieval from patent documents of different domains for innovatively solving target problems appears a significant and worthwhile challenge.

In recent years, various machine learning tools and techniques have been

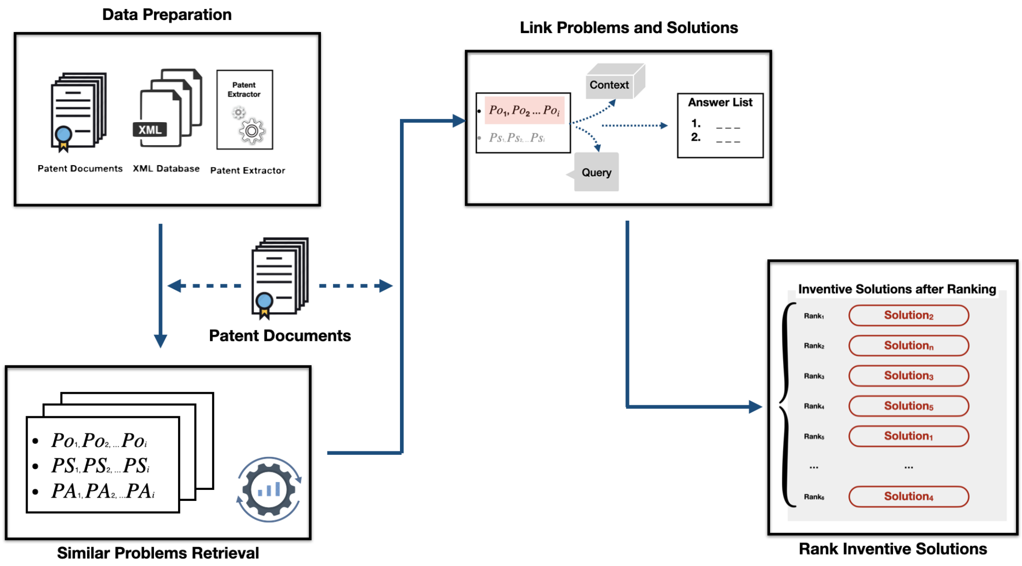


FIGURE 1.2: The global architecture of the thesis

developed in the analysis of patents and automation of the process of mining knowledge. Indeed, Natural Language Processing (NLP) techniques have witnessed a major leap forward. We especially use NLP techniques in this work. As illustrated in Fig. 1.2, the thesis is carried out under the framework of problem-solving contents extracted from patent documents using NLP techniques. It aims to assist engineers in solving complex and multidisciplinary problems through the subsequent process of similar problems retrieval to problem-solution match to innovative solutions ranking. Within this framework, a large number of patent documents from different domains are used to extract latent innovative solutions for the given real-world problems. We postulate that existing innovative knowledge contained in patent documents if brought to them appropriately could constitute a useful tool for engineers. Especially for those without a broad understanding of different domains and could thus contribute to their inventive R&D activities' performance.

In this thesis, we focus on developing an entirely automated solution retrieval process using relevant knowledge contained in patents from different domains to innovatively solve real-world target problems.

1.2 Motivation

The motivation driving this work is to facilitate and automate the entire inventive problem-solving process using patent documents based on NLP techniques compared to the difficult way to classically operate TRIZ theory.

As stated in Section 1.1, different domains patent documents contain rich (diverse) published information and the latest innovative knowledge. This innovative knowledge in patent documents always tends to present leading-edge problem-solving solutions. Furthermore, The innovative knowledge contained in patents could be defined as *problems* that it aims to solve (Souili, Cavallucci, and Rousselot, 2015a). The problem describes unsatisfactory features of existing methods or situations.

For instance, for the touch pen use case[1].

- *non-conductive materials like plastic could hamper users to operate the pen by wearing gloves, having very dry skin, or some situations in which the user does not make good conductive contact with the device to the touch screen.*

This problem could show up especially when the environment is cold. The patent, therefore, proposes *partial solutions* which provide improvements or changes to the defined problems. A partial solution could be:

- *replacing the inner moulding built by non-conductive material of touch pen with an ideally metallic material device so that the stylus tip operates even when held by an extremely good insulator.*

Therefore we are experiencing mining innovative problem-solving knowledge contained in patent documents. This can solve the target problems from different domains when these problems are similar enough.

However, there are already a large number of patent documents in the world and numerous new patent documents have been published every year. To verify this permanently growing quantity of existing patents manually is an impossible work. In addition, patent documents are spread all over a wide range of different domains. As an example, the United States Patent and Trademark Office (USPTO) dataset is a relevant sample of the overall 130 million patents produced so far. These patents are from eight domains. Obviously, no engineer can possess a broad understanding of all these different domains. Consequently, it constitutes a worthy goal for us to help engineers to cope with this challenge and let them fully benefit from patent documents when solving inventive problems. Besides, with the help of our work, engineers do not need to master the complexity of TRIZ theory and could manually explore innovative solutions from a broad scope of patent contents.

[1] reader may refer to this link for the full patent `https://patents.google.com/patent/US8847930B2/`

During our research, we notice that different domains patent documents contain similar semantic problems. When these problems are similar enough from a semantic viewpoint, their corresponding solutions proposed by different domains patent documents could be a candidate of innovative solutions for the target problem. The distance between their domains might also provide potentially more innovative solutions or inspirations for solving the target problem. Classically, most engineers fail to benefit from distant domains' knowledge since they are not aware of their mastering. Consequently, what motivates us is to make better use of this distant knowledge and propose a novel innovative knowledge retrieval process.

Moreover, in the context of automating the complex data mining process, we decided to automate the entire innovative solutions retrieval process to avoid complex cooperation works between different preparation phases. From data preparation to similar problems retrieval to problem-solution matching and innovative solutions ranking, these different steps cooperate to let the entire innovative solutions retrieval process be automated. Especially when a large number of patent documents are at the input of the process.

Furthermore, deep learning approaches have been quickly developing in recent years. Several state-of-the-art neural networks approaches have achieved promising results on different research fields and tasks, especially the NLP field. Due to their nature of design and structure, different neural network approaches feature different performances on various tasks. For instance, for predicting the target word with the background of long context information, Long Short Term Memory networks (LSTMs) perform better than Word2vec neural networks. It can better learn the longer context information around the target word due to its original design of forget and memory gates. On the other hand, patent documents are different from other types of text. They contain longer sentences compared to other documents. Besides, several complex expressions and specific vocabulary are numerous in sentences of patent documents. It makes context information analysis longer and difficult to achieve. All these limitations make the innovative solutions retrieval task from patent documents a significant aim. It also motivates us to explore different NLP techniques to analyze if we can find a relevant one for our challenge.

To conclude, the aforementioned challenges motivate us to propose an automatic innovative solutions retrieval process. Our proposed approaches combine data mining methods, semantic similarity computation technologies, and deep learning approaches in the NLP field to automate the innovative solutions retrieval from a large number of patent documents. Our work can eventually provide engineers without a broad understanding of different domain knowledge a new way to facilitate innovative inspiration from patent documents in given situations. On the other hand, to the best of our knowledge, our work is the first to make full use of the knowledge of different domains patent documents to automatically provide innovative solutions. It further facilitates the problem-solving research work in the TRIZ field and NLP field.

1.3 Contributions of the Thesis

Based on the motivation, in this thesis, the following contributions have been proposed in multiple research directions:

- **IDM-Similar model based on Word2vec neural networks**: A similar problem retrieval model called IDM-similar is proposed in this thesis. According to Word2vec neural networks (Mikolov et al., 2013b), the IDM-Similar model can extract similar problems from different domains patent documents. It obtains the sentence vector for each target problem in patent documents via Word2vec. Moreover, we first train the Word2vec model based on an open-source English Wikipedia dataset. It can thus learn semantic similarity among different words. Sentence representations can then be achieved via word representations. The cosine similarity metric is also combined to predict the similarity values between sentence pairs. The IDM-Similar model is eventually able to extract similar problem sentences from patent documents through the similarity computation.

- **SAM-IDM model based on Long Short Term Memory (LSTM) neural networks**: To let the model better learn long context information according to patent document features, a novel similar problem extraction model called SAM-IDM relying on LSTMs is proposed. It combines a Manhattan LSTMs model to figure out the semantic similarity comparison task among different sentences. In addition, an implementation of a pruning process is used to ensure a higher level of innovativeness and time efficiency. Compared to the IDM-Similar model, SAM-IDM illustrates a better performance on real-world U.S. patent documents.

- **IDM-Matching model based on XLNet neural networks**: To match problems and corresponding solutions in patent documents, a model called IDM-Matching is proposed in this thesis. It combines state-of-the-art neural networks called XLNet in the NLP field. Specially, we treat this task as a question answering system (Ravichandran and Hovy, 2002). We especially convert each problem into a query to make full use of XLNet neural networks and avoid the drawbacks of traditional lexico-syntactic pattern matching methods. This model aims to build the link between problems and solutions in patent documents to match similar problems from the SAM-IDM model with innovative solutions from different domains patent documents.

- **PatRIS model based on multiple-criteria decision analysis**: A model called PatRIS based on the multiple criteria decision analysis (MCDA) approach is proposed to rank latent innovative solutions. When a large number of patent documents are used to be the input of the SAM-IDM model, several similar problems might be generated from different domain patent documents. Several corresponding latent innovative solutions could be therefore achieved via the IDM-Matching model. To better rank these latent innovative solutions, the PatRIS model combines an MCDA approach called TOP-SIS. Furthermore, several patent innovativeness indicators and the similarity value are combined with different weights to build a solution innovativeness ranking system. This work aims to provide engineers with a way to reveal the most eligible innovative solutions when the number of latent innovative solutions is high.

- **A demonstrator named PatentSolver**: According to the aforementioned motivation in Section 1.2, a demonstrator named PatentSolver is proposed in this thesis. It contains the aforementioned models to automate the entire process. Several functions such as patent details presentation, patent number search, presentation of the similar problem list as well as the corresponding solutions list, and ranking of innovative solutions are all developed and assembled into PatentSolver. It is a software prototype destined for engineers. The future software will be based on PatentSovler but will process real-time patent data to let an industrial field benefit from our research works and further facilitate R&D activities.

1.4 Thesis Structure

Fig. 1.3 illustrates the structure of this thesis. It is organized with four main parts divided into eight chapters:

- **Part I: Introduction** provides a general introduction of the thesis. Moreover, a general introduction to the background, motivations, main contributions of the thesis, and the structure of the thesis are presented in this part.

- **Part II: NLP-related Theoretical Foundations** presents NLP-related theoretical foundations and research work corresponding to the thesis. A comprehensive review of existing research works is provided in this part. Furthermore, it especially presents representation learning approaches in the Natural Language Processing (NLP) field. Several state-of-the-art deep learning models are introduced in detail. We mainly present them in three directions: the use of representation learning approaches in the NLP field, semantic similarity computation approaches via deep learning approaches, and NLP applications. We also introduce the advantages and disadvantages of different types of approaches. In addition, we summarize the relationship between the aforementioned related works and the contributions of this thesis.

- **Part III: The Theory of Inventive Problem Solving** introduces the classical theory of inventive problem solving (TRIZ), including a literature review on TRIZ relevant to the scope of this thesis. In addition, the related knowledge about ontology-based knowledge modelling and their application status in the problem-solving field are also presented. The related literature of IDM-related knowledge deriving from TRIZ is also introduced.

- **Part IV: Contributions** demonstrate the contributions of this thesis. It consists of five chapters.

  – **Chapter 4** introduces two models that are developed for extracting similar problems from different domains patents. This chapter includes two different sentence semantic similarity comparison models according to Word2vec neural networks and LSTM neural networks. Within these models, different prediction performances are generated due to their natural mechanisms and learning ability on context information. Several case studies based on the real-world U.S. patent dataset are illustrated in detail.

  - **Chapter** 5 mainly illustrates a novel problem-solution matching model for building links between target problems and corresponding solutions in patent documents. State-of-the-art neural networks called XLNet is used to let the model be a question answering system. The model is thus able to match problems and corresponding solutions in the patent document by answering queries that are converted via target problems. A detailed case study on the real-world U.S. patent dataset illustrates the model's performance.
  - **Chapter** 6 mainly presents a model relying on the MCDA approach called TOP-SIS for ranking latent innovative solutions. We combined several chosen indicators of the corresponding patent innovativeness and similarity value as the latent innovativeness indicators of target solutions into the model. The model can eventually rank target solutions by ordering their latent innovativeness. It is also validated on the real-world U.S. patent dataset.
  - **Chapter** 7 introduces a software prototype we developed to automate the innovative solutions retrieval work from a large size of patent documents. It is named as PatentSovler. The software combines the aforementioned models to fully use the innovative knowledge contained in patent documents to provide the most possible innovative solutions for solving real-world problems. It can further facilitate R&D activities for companies.
  - **Chapter** 8 summarizes the thesis and outlines future perspectives.

- **Part V**: **Appendix** A, B, C, D, and E illustrate traditional TRIZ knowledge sources involved in Chapter 3. **Appendix** F presents several additional real solved use cases found in the literature that are provided by the SAM-IDM model in Chapter 4. These experimental results are investigated by experts to check their potential in creating an innovative solution. **Appendix** G gives the theoretical foundations and basic concepts of Multiple Criteria Decision Analysis (MCDA). As a subdiscipline of operation research, decision assistance is also introduced. In particular, we present a typical sequence of performing an MCDA approach in detail. This is combined into our PatRIS model in Chapter 6.

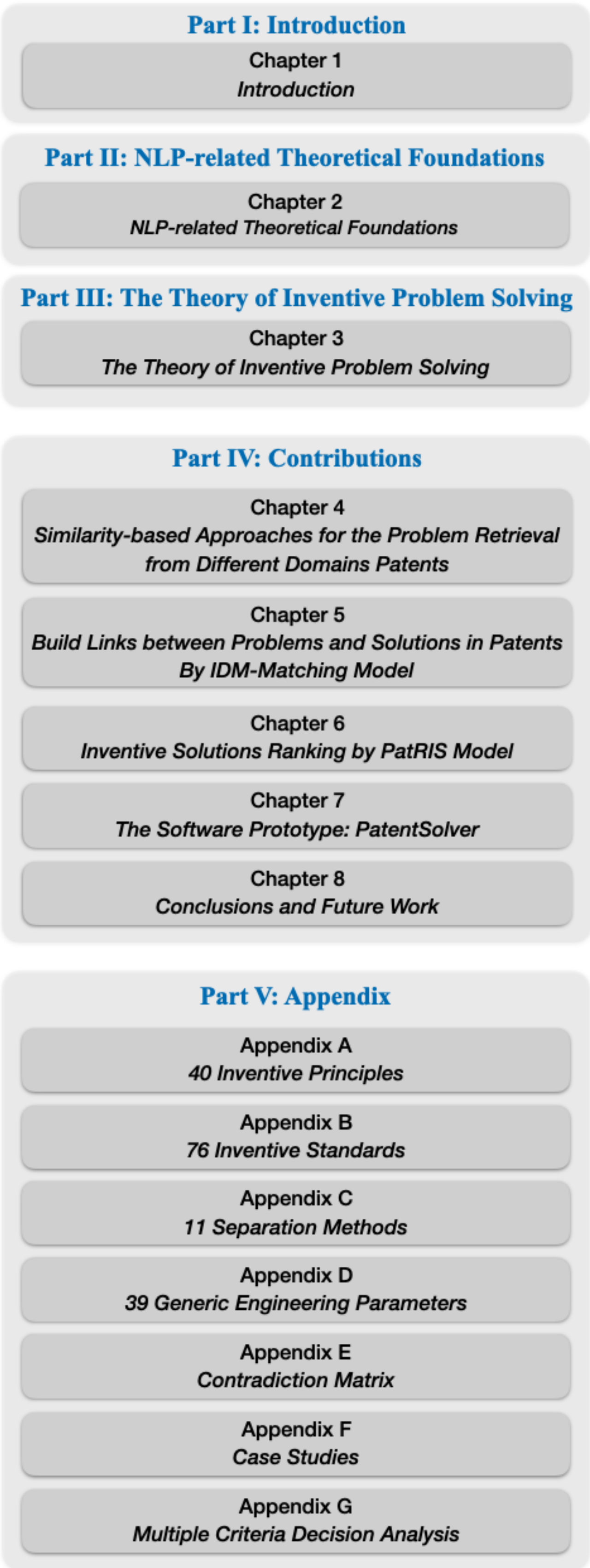


FIGURE 1.3: Structure of the thesis

Chapter 2

NLP-related Theoretical Foundations

This chapter presents related research works about the Natural Language Processing (NLP) field, including several typical approaches about the semantic textual similarity computation and NLP-related applications.

2.1 Representation Learning on Natural Language Processing

In order to avoid understanding the complex usage of the classical TRIZ and benefit from the innovative knowledge contained in patent documents with the technology of artificial intelligence, we especially choose NLP techniques in the thesis. Indeed, different NLP techniques with appropriate architectures are chosen to fit our research purposes, automation of innovative solutions from a large number of patent documents. In particular, sentences of patent documents usually consist of more tokens (words), which are far longer than the length that common sentences may contain. It also lets us explore the use of different NLP techniques to fit this unique feature of patent documents. Therefore, different deep learning approaches in the NLP field are used in the thesis to automate the inventive solutions retrieval from patent documents.

In detail, natural languages are the languages that people use every day, such as English, Chinese, and French. These languages evolve naturally with the development of human society. They are important tools for the record and transfer of human knowledge. Throughout human history, knowledge in the form of written language has accounted for more than 80% of the total human knowledge. In the computing application of the computer, according to statistics, only 10% is used for the mathematical calculation, less than 5%

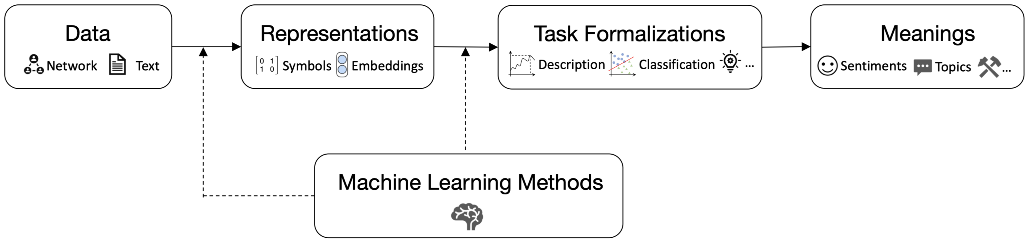


FIGURE 2.1: The generic process of NLP

for the process control, and the remaining 85% is used for linguistic information processing. Processing contains the process of understanding, transfer, and generation. NLP aims to let computers process the formulation, voice, and semantic meanings of natural languages. On the other hand, it is the processing for the input, output, recognition, analysis, understanding, and generation of the characteristic, phrase, sentence, paragraph, and chapter. As illustrated in Fig. 2.1 (Chen et al., 2021), the generic process of NLP is presented. Over the past years, since the huge advances in deep learning in the computer vision and speech recognition fields, deep learning has also been used in the NLP field. In general, deep learning involves multiple layers of neural networks. It achieves outputs from inputs by the sequential nonlinear variation, an end-to-end training from inputs to outputs. In this chapter, we especially introduce the related knowledge about NLP, deep learning, and its usage in the NLP field.

2.1.1 Natural Language Processing

Natural Language Processing (NLP), also known as computational linguistics, involves the engineering of computational models and programs to solve the practical problems of understanding human language (Otter, Medina, and Kalita, 2020).

Moreover, NLP involves teaching machines to interpret, classify, manipulate, and generate language. From the early use of handwritten rules and statistical techniques to the recent adoption of deep learning, the NLP domain has provided several tools to solve issues of machine understanding text, with applications in text generation, machine translation, question answering, and other tasks (Zhang et al., 2021).

The NLP field has also experienced rapid growth in recent years. Especially, as the most significant technology, representation learning contributes to the significant development of NLP (Bengio, Courville, and Vincent, 2013).

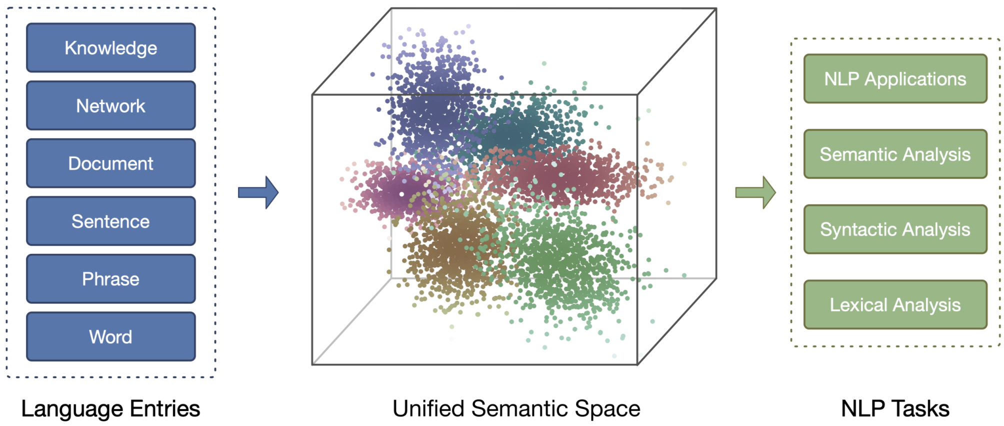


FIGURE 2.2: Distributed representation can provide unified semantic space for multi-grained language entries and for multiple NLP tasks

Over the past few years, with the growing computing power and large-scale text data, distributed representation trained with neural networks and large corpora has become the mainstream (Liu, Lin, and Sun, 2020). In detail, distributed representation is a representation of the observed data in such a way that they are modelled as being generated by the interactions of several hidden factors. A particular factor learned from configurations of other factors can often generalize well (Deng and Yu, 2013). As shown in Fig. 2.2 (Liu, Lin, and Sun, 2020), representation learning can facilitate knowledge transfer across multiple language entries, multiple NLP tasks, and multiple application domains.

In the timeline of representation learning development for NLP, from N-gram model (Brown et al., 1992), bag-of-words (Harris, 1954), distributed representation (Hinton et al., 1986), neural probabilistic language model (Bengio et al., 2003), to pre-trained language models (Edunov, Baevski, and Auli, 2019), these typical approaches continuously improve the performance of different NLP tasks. Distributed representations especially form the basis of deep learning. Therefore, in the thesis, we either combine them into our models or use them as comparative models. We introduce them in detail in the following sections.

- **N-gram Model** It predicts the next item in a sequence based on its previous n-1 items.

- **Bag-of-Words** It represents a sentence or a document as the bag of its words.

- **Distributed Representation** It represents items by a pattern of activation distributed over elements.
- **Neural Probabilistic Language Model** It learns a distributed representation of words for language modelling.
- **Pre-trained Language Model** It includes contextual word representation, the novel pre-training-fine-tuning pipeline, larger corpora, and deeper neural architectures.

2.1.2 N-gram Model

As one of the earliest word representation learning approaches, the N-gram model is used to predict a word from previous words in a sample of text (Brown et al., 1992). Furthermore, N-grams are sequences of characters or words extracted from the text (Majumder, Mitra, and Chaudhuri, 2002). It is coherent with the distributional hypothesis: linguistic items with similar distributions have similar meanings (Harris, 1954). It is the fundamental idea of several NLP models, from Word2vec (Mikolov et al., 2013b) to BERT (Devlin et al., 2018). N-grams can be classified under two categories: character-based and word-based.

- **Character N-gram** is a collection of n consecutive characters extracted from a word. The main motivation behind this approach is that similar words will contain a high percentage of N-grams in common. Typical values for n are 2 or 3. These values correspond to the use of bigrams or trigrams, respectively. For instance, for the word "car", it results in the generation of the bigrams as **c, ca, ar, r** and trigrams as ***c, *ca, car, ar*, r***.
- **Word N-grams** are sequences of n consecutive words extracted from a text. Word-level N-gram models are quite robust for statistical modelling of language as well as for information retrieval and are not very language-dependent. For instance, for the sentence "*Car is cleaned by Tony.*", a created vocabulary set by bi-gram is [*"car is", "is cleaned", ... , "by Tony", "by car", "Tony is"*].

The N-gram model is thus generally integrated into the document classification task. It is able to let the model consider the sequences of words instead of singular words (unigrams).

All Words

[“Hi”, “How”, “are”, “you”, “bye”, “see”, “later”]

“Hi”	$\longrightarrow$	[1, 0, 0, 0, 0, 0, 0]
“How are you”	$\longrightarrow$	[0, 1, 1, 1, 0, 0, 0]
“Bye”	$\longrightarrow$	[0, 0, 0, 0, 1, 0, 0]
“See you later”	$\longrightarrow$	[0, 0, 0, 1, 0, 1, 1]

FIGURE 2.3: An example of Bag-of-Words

2.1.3 Bag-of-Words

The Bag-of-Words model (BOW) is an orderless documentary representation, which is built on the distribution hypothesis (Harris, 1954). The distributional relation about elements' occurrence between the correlation of several aspects of meaning is revealed. BOW only concerns words, whether they occurred but not where they are. In this approach, each word count can be considered as a feature (Goldberg, 2017).

As a special n-gram model where $n = 1$, it ignores the text's syntax and word order and sees the text as a combination of several individual words. The occurrence of each word in the text is independent (Soumya George and Joseph, 2014). A bag-of-word vector is produced to represent texts. BOW first designs a vocabulary of words using every word in the corpus. Then it maps the text to it as a bag-of-word vector where co-occurrence words with the vocabulary are shown as "1" and the inverse as "0" (Ni, Samet, and Cavallucci, 2021). For instance, as illustrated in Fig. 2.3, the co-occurrence words in bag-of-words (All Words) are presented as "1" in the corresponding vector. After that, the text is converted into fixed-length vectors of numbers and it solves the issue that machine learning approaches fail to deal with the raw text directly.

2.1.4 Distributed Representation

Distributed representations for words were first proposed by (Hinton et al., 1986). Furthermore, distributed representations of words in a vector space can help to teach algorithms how to achieve better performance in NLP tasks by grouping similar words (Mikolov et al., 2013a).

The significant performance of distributed representations in the NLP field lets it become the paradigm of statistical language modelling (Elman, 1990; Mikolov et al., 2012). It is used in several NLP applications such as word representation, named entity recognition, word sense disambiguation, parsing, tagging, and machine translation (Collobert and Weston, 2008; Turney and Pantel, 2010; Turian, Ratinov, and Bengio, 2010; Collobert et al., 2011; Socher et al., 2011; Zou et al., 2013; Huang et al., 2012). Besides, representing phrases also becomes the new trend (Mitchell and Lapata, 2010; Yessenalina and Cardie, 2011; Grefenstette, 2013; Mikolov, Le, and Sutskever, 2013) in the NLP field. As the most typical distributed representation approach, Word2vec (Mikolov et al., 2013a), a simple and efficient distributed word representation, is mostly used in several NLP models. In detail, Word2vec is a two-layer neural network that can be trained by a given corpus to convert each unique word presentation in the corpus into a computable and structured vector in the space. Word vector is positioned in the vector space so that words sharing common contexts in the corpus are located in close proximity to one another in the space (Mikolov et al., 2013b). It allows words with similar meanings to obtain a similar representation. For instance, as illustrated in Fig. 2.4 (Mikolov, Le, and Sutskever, 2013), with the distributed word representations, five word vectors in English (left) and Spanish (right) are projected down to two dimensions. These concepts have similar geometric arrangements in both spaces. It reflects that distributed representations of words can present similarities in vector spaces of languages. For bag-of-words, its two major drawbacks are obvious. The ordering of the words is lost and the semantics of the words are also ignored. But distributed representation approaches can address these issues.

2.1.5 Neural Probabilistic Language Model

As one of the pioneer practices of distributed representation in NLP (Liu, Lin, and Sun, 2020), the neural probabilistic language model is to learn the joint probability function of sequences of words in a language (Bengio et al., 2003).

A computationally efficient probabilistic modelling approach is proposed

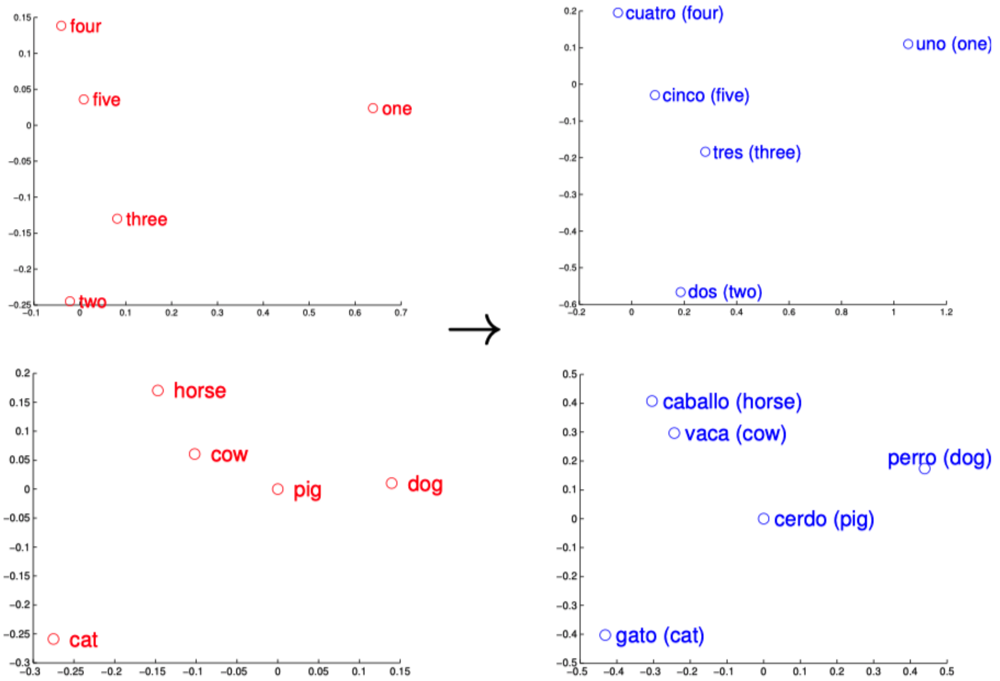


FIGURE 2.4: An example for the distributed representation of words

by (Bengio et al., 2003) to figure out the issue of the curse of dimensionality, which is that the joint distribution of a large number of discrete variables results in exponentially large parameters. It allows each training sentence to provide the model with an exponential number of semantically adjacent sentences. The model is able to learn a distributed representation of each word and a probability function for the sequence of words, which are represented in terms of these representations, and word embeddings (i.e., low-dimensional word vectors) are brought by it as learned parameters. Semantic meanings of words are indeed encoded by these vectors. Several typical models including Word2vec (Mikolov et al., 2013b), Glove (Pennington, Socher, and Manning, 2014), and fastText (Bojanowski et al., 2017) are inspired by neural probabilistic language model to embed words into distributed representations to optimize them as model parameters. It eventually makes it possible to take advantage of longer contexts and significantly improve the model's performance compared to n-gram models.

2.1.6 Pre-trained Language Models

In past few years, with the growing computing power and large scale text data, distributed representation trained with neural networks and large corpora have become the mainstream. Pre-trained language models based on the various neural networks have brought NLP to a new era (Qiu et al., 2020).

From Convolutional Neural Networks (CNNs) (Kalchbrenner, Grefenstette, and Blunsom, 2014; Kim, 2014; Gehring et al., 2017), Recurrent Neural Networks (RNNs) (Sutskever, Vinyals, and Le, 2014; Liu, Qiu, and Huang, 2016), graph-based neural networks (GNNs) (Socher et al., 2013; Tai, Socher, and Manning, 2015; Marcheggiani, Bastings, and Titov, 2018), to attention mechanisms (Bahdanau, Cho, and Bengio, 2014; Vaswani et al., 2017), an obvious advantage of these neural networks is the ability to alleviate the feature engineering issue. Compared to other non-neural NLP approaches, models based on neural networks approaches are capable of using low-dimensional and dense vectors (distributed representation) to implicitly represent the syntactic or semantic features of the language. It makes using a large size of an unlabelled corpus such as Wikipedia dataset to train models be possible. The novel paradigm of mainstream NLP research works becomes that using pre-trained models according to different large datasets and fine-tune then models on specific downstream tasks.

Overall, pre-trained language models can be classified into two major types: pre-trained word embedding models such as CBOW, Skip-gram (Mikolov et al., 2013b), Glove (Pennington, Socher, and Manning, 2014), and Fasttext (Joulin et al., 2016) as well as pre-trained contextual encoder models such as LSTMs (Fernando et al., 2018), ELMo (Peters et al., 2018), BERT (Devlin et al., 2018), GPT (Brown et al., 2020). These pre-trained models are able to transform the word representations learned in the pre-training stage into the downstream tasks. However, the drawbacks of different types of pre-trained language models are also obvious. Pre-trained word embedding models fail to capture higher-level concepts in a context like polysemous disambiguation, syntactic structures, semantic roles due to their fixed word vector representations cannot be changed with the different context information. The huge computational consumption and time cost of pre-trained contextual encoder models made them unaffordable for several tasks. Therefore, the tradeoff for using different models on different tasks is significant.

2.2 Semantic Similarity Computation

Semantic similarity computation involves understanding the meaning of words, phrases, sentences, or documents at some level (Otter, Medina, and Kalita, 2020). Estimating the semantic similarity between textual data is one of the most challenging and open research issues in the field of NLP. The variability of natural languages made it difficult to define rule-based approaches for determining semantic similarity measures. Therefore, different types of semantic similarity computation approaches have been proposed for addressing the issue over years. It can be mainly classified as follows:

- **Knowledge-based Semantic Similarity Computation Approaches** compute the semantic similarity between two terms based on information obtained from one or more underlying knowledge sources such as lexical databases. The underlying knowledge provides these approaches with a structured representation of terms connected by semantic relations. It can further provide an unambiguous semantic measure as to the actual meaning of the terms. TF-IDF (Jones, 1972) is a typical approach. Knowledge-based semantic approaches are computationally simple and can easily be extended to compute sentence-to-sentence similarity measures.

- **Corpus-based Semantic Similarity Computation Approaches** compute semantic similarity between terms via information retrieval from large corpora. Distributional hypothesis mentioned in chapter 2.1.4 states that "*similar words occur frequently*". Thus, various models were proposed to construct the vector representation of the textual data (Liu, Lin, and Sun, 2020). Several semantic similarity measurement approaches are then proposed to estimate the similarity between vectors (Mikolov et al., 2013b; Pennington, Socher, and Manning, 2014; Bojanowski et al., 2017; Devlin et al., 2018). In particular, the cosine similarity metric is widely used among these approaches. Moreover, as a typical word embedding method, the Word2vec model mentioned in chapter 2.1.4 is the most widely used pre-trained word embedding approach.

- **Deep Neural Networks-based Semantic Similarity Computation Approaches** further improve the performance of semantic similarity computation in NLP tasks and outperform most traditional approaches. Two fundamental operations, convolution and pooling, are mainly used to build deep neural networks. Convolution operations are used for feature extraction and pooling operations are used to eliminate features containing negative impacts. For instance, LSTM neural networks (Hochreiter and Schmidhuber, 1997) as a special recurrent neural network (RNN) (Mueller and Thyagarajan, 2016) and transformer-based models relying on attention mechanisms (Vaswani et al., 2017) are frequently used as typical deep neural network-based approaches to capture semantic similarity. Even though deep neural network-based approaches have always achieved state-of-the-art performance, the drawbacks of huge computational consumption and lack of interpretability are still requiring attention.

According to the above analysis, in the thesis, we favour different semantic similarity comparison approaches based on deep neural networks for our research aims. In detail, we either use them to capture similar problem sentences from different domains patent documents for target problem sentences or extract inventive solutions towards corresponding problems. We benefit from different approaches' advantages to fit our different tasks to make the entire inventive solutions retrieval process optimized. We introduce the use of typical approaches in detail in Chapter 4 and Chapter 5.

2.2.1 Overview of Semantic Similarity Computation Approaches

In recent years, several efforts have been invested in semantic similarity computation of the NLP field, including words (Mikolov et al., 2013b; Pawar and Mago, 2018), sentences (Wang, Mi, and Ittycheriah, 2016; Mueller and Thyagarajan, 2016), and documents (Mueller and Thyagarajan, 2016; Benedetti et al., 2019).

(Deerwester et al., 1990) proposed to use the latent semantic structure of documents to reduce the dimensionality of document vectors to improve relevant document detection. Over the years, the Bag-of-Words model (BOW) (Goldberg, 2017) and Term Frequency-Inverse Document Frequency (TF-IDF) (Rajaraman and Ullman, 2011) were the main stream's research approaches to the semantic similarity. Indeed, TF-IDF is a statistical approach for computing how important a word is to a document in a corpus (Rajaraman and Ullman, 2011). It computes the weight of each word according to the occurrences' percentage of each word in the document. Words containing higher word frequency (TF) weights imply a higher degree of relevance to the document. Lower inverse document frequency (IDF) weights are assigned to words that occur frequently throughout the corpus. For instance, stop words that fail to contain any meaning but occur in a high percentage of the text, such as "*the, which* ...", will be penalized when assigning weights. TF-IDF aims to find documents that are highly relevant to the query. Therefore, it has been used for similarity computation in several works. We also use it as the comparison approach in Section 4.3. With the further development of NLP, the idea that texts are similar if words are similar became another baseline for the similarity computation. In particular, considering not only words, but the full context meaning became the novel research focus in recent years. Especially, the fact that context information fails to be fully used becomes a crucial bottleneck to further improve the performance of the similarity computation in the NLP field.

To address this issue, several approaches have been proposed. Among them, the word representation with the low-dimensional vector is becoming an important basis for similarity computation in recent years. Several word embedding techniques have been proposed so far, including n-gram models, unsupervised learning (Mnih and Hinton, 2009), and neural network-based approaches. Among these approaches, Word2vec (Mikolov et al., 2013b) relying on neural network architecture outperforms others. It is a representative model with a two-layer neural network, which can be trained by a large-scale corpus to achieve a vector in the space for each unique word. It aims to reconstruct linguistic contexts of words. Besides, a classifier combined word alignment and saliency-weighted semantic graphical approaches have been proposed by (Kenter and De Rijke, 2015) to predict a semantic similarity score from word-level to text-level semantics. (Kusner et al., 2015) have introduced a novel distance function between text documents called Word Mover's Distance (WMD), which measures the dissimilarity between two text documents. (Ni, Samet, and Cavallucci, 2019) proposed to combine sentence vector produced by Word2vec model and cosine similarity metric to measure the similarity possibility between sentence pairs to find out similar problems from different domains patents.

BERT, a pre-trained model of deep bidirectional transformers for language understanding, became state-of-the-art approach recently. It obtained outperforming performance on sentence-pair regression tasks like semantic textual similarity (Devlin et al., 2018). However, its drawbacks of a massive computational overhead and memory intensiveness (Jiao et al., 2019) are also obvious. For example, finding the most similar pair in a collection of 10,000 sentences requires about 50 million inference computations (around 65 hours) with BERT. The construction of BERT makes it unsuitable for semantic similarity search as well as for unsupervised tasks like clustering (Reimers and Gurevych, 2019). A huge number of pairwise sentence comparisons in different domains patents discard using BERT in our work.

Nevertheless, as a special type of Recurrent Neural Networks (RNNs) (Mikolov et al., 2010), Long Short Term Memory networks (LSTMs) (Hochreiter and Schmidhuber, 1997) are more suitable for the work (Section 4.3) in the thesis due to its capacity of learning longer text information. Compared to RNNs and other neural network approaches, LSTMs can remember related information for a long period of time (Gasmi, Laval, and Bouras, 2019). It is designed to avoid the long-term dependency problem because of its special default design, especially the forget gate layer. Furthermore, (Mueller and Thyagarajan, 2016) proposed a model called Manhattan LSTM (MaLSTM) to the sentence similarity computation task. As a siamese adaptation of the Long Short-Term Memory (LSTM) network for labelled data composed of pairs of variable-length sequences, it can assess the semantic similarity between sentences. Compared to other neural network systems of greater complexity, its outperforming performance also inspires us to combine it into our model.

2.3 Applications of Natural Language Processing

NLP techniques have been used for various tasks. It can be mainly classified into six types: information retrieval, text classification, text generation, summarization, question answering, and machine translation. In the thesis, our similar problem retrieval work belongs to the application of information retrieval. We also build a problem-solution matching model by using a question answering system. We thus briefly introduce these two applications in this section.

- **Information Retrieval** An information retrieval system is designed to analyze, process, and store sources of information and retrieve those that match a particular user's requirements (Chowdhury, 2010). Deep learning models are generally used to match texts of queries to texts of documents in order to obtain relevance scores among them. Such models, therefore, focus on representing the interactions between the query and the individual words in the document. In addition, semantic relation measurements are always applied in information retrieval

(Lee, Kim, and Lee, 1993). In particular, (Ensan and Du, 2019) propose a novel semantic retrieval framework that uses semantic entity linking systems for forming a graph representation of documents and queries. It is capable of addressing the challenges of traditional keyword-based retrieval systems, such as the vocabulary gap between the query and document spaces. Besides, pre-trained language models are also recently wildly used in this field. For instance, two pre-trained contextualized language models (ELMo and BERT) are used by (MacAvaney et al., 2019) to rank ad-hoc documents. The proposed joint approach that incorporates BERT's classification vector into existing neural models achieved outperforming performance. In the thesis, we performed the semantic similarity measure to the information retrieval task, similar problems retrieval from patent documents (in Chapter 4).

- **Question Answering** Question Answering can be considered as an extension of search engines in the sense, that they aim at automatically supplying users with precise answers to questions posed in natural language, instead of simply returning a ranked list of relevant sources based on a set of keywords (Dimitrakis, Sgontzos, and Tzitzikas, 2020). Similar to summarization and information retrieval, question answering gathers relevant words, phrases, or sentences from a document. It coherently returns the information in response to a query. Neural network-related approaches have been used in the area in recent research works. (Wang et al., 2017) propose the gated self-matching networks, an end-to-end neural network model, for reading comprehension style question answering, which aims to answer questions from a given passage. Gated attention-based recurrent networks and self-matching attention mechanisms are used in this work to obtain representation for the question and passage, and then use the pointer-networks to locate answer boundaries. It achieved a state-of-the-art performance on the SQuAD dataset (Rajpurkar et al., 2016), a baseline dataset in the question-answering field. After that, Bidirectional Encoder Representation from Transformer (BERT) (Wang et al., 2017) updates state-of-the-art performance in question answering experiments on both SQuAD 1.1 and SQuAD 2.0 datasets. In fact, various deep neural networks based on the transformer mechanism have been continually updating the performance in this area. We also combined related techniques and a question answering system into our IDM-Matching model (in Chapter 5) in the thesis.

2.4 Summary

In this chapter, NLP-related theoretical foundations and approaches are introduced comprehensively. Especially, to exploit an automatic mechanism for similar problems retrieval and problem-solution matching systems, deep learning-related technologies on NLP and semantic similarity computation approaches are introduced in detail, including their basic approaches, concepts, and the current situation of their use, especially for applications of NLP. In the next chapter, we will comprehensively introduce the Theory of Inventive Problem Solving (TRIZ) and related knowledge.

Chapter 3

The Theory of Inventive Problem Solving

TRIZ, coming from the Russian acronym "teorija rezhenija izobreta- telskih zadach" (Shirwaiker and Okudan, 2008), is the Theory of Inventive Problem Solving (Altshuller, 1999). It has been developed by Altshuller in Russia from 1946 to 1985 by analyzing a hundred thousand patent documents, as shown in Fig. 3.1 (Cavallucci and Khomenko, 2007).

As a systematic methodology or set of techniques, TRIZ provides a logical approach to enhance creativity when in an innovation logic, more precisely in inventive problem solving. It has also spread over an ever-increasing quantity of countries across the world (35 to date). (Bae, 2005) especially emphasizes that *"TRlZ problem solving methods are especially suited for rapidly, identifying innovative solutions that are more both robust and economical than conventional methods"* for stating the creative problem solving. In addition, TRIZ relies on the fact that innovation is governed by certain repetitive patterns and it can be used in any field. On the other hand, it is able to support engineers to inventively solve problems by using the previous inventors' knowledge. Besides, with the advancement of TRIZ, several extensions of TRIZ and combination with other methods have been proposed. For instance, OTSM-TRIZ (Cavallucci and Khomenko, 2007) and IDM (Inventive Design Methodology) (Cavallucci, 2009), which are used to figure out complex industrial problems and to ease the inventive product design. The combinations of TRIZ with Quality Function Deployment (QFD) (Domb, 1998), Kano model (Ungvari, 1999), axiomatic design (Mann, 2002; Cavallucci, Rousselot, and Zanni, 2009), and six sigma (Slocum and Kermani, 2006) have also been proposed to increase its effectiveness.

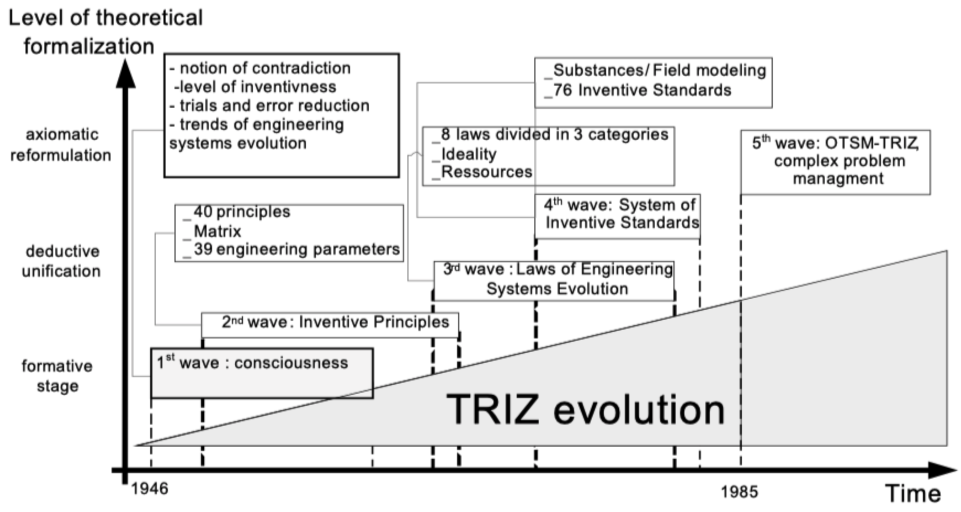


FIGURE 3.1: The evolution of TRIZ

3.1 Definition of TRIZ

Altshuller Institute for TRIZ Studies and European TRIZ Association have defined the TRIZ body of knowledge (Litvin et al., 2007). The frameworks and definitions associated with the methodology are numerous and consequently, newcomers encounter difficulties in rapidly grasping the essentials of it. To be more precise, "TRIZ is a human-oriented knowledge-based systematic methodology of inventive problem solving" (Savransky, 2000). The explanation to the definition is as follows:

- **Human-oriented** As the practice of TRIZ depends on the problem itself and the socio-economic environment, it is human beings, not machines, who set the direction of the heuristic.
- **Knowledge-based** Knowledge of generic problem-solving heuristics is extracted from thousands of patents in the engineering fields. TRIZ uses not only knowledge from the natural and engineering sciences, but also knowledge about the specific problem domain.
- **Systematic** TRIZ offers an effective application of existing solutions to new problems and the creative process is systematically structured.
- **Inventive problem solving** TRIZ aims to solve creative problems in which only the main contradiction needs to be solved.

In addition, with the definition of TRIZ, Altshuller proposed three primary findings which are:

- Problems and solutions could be shared among all industries.
- The pattern of technological evolution also repeats itself in different industries.
- The scientific effects used by an innovation go beyond the field in which it was developed.

3.2 Levels of Innovation

Alshuller introduced five levels of innovation by analyzing 400.000 patent documents (Shulyak, 1998). Indeed, not each invention contains an equal inventive value. We introduce these five levels of innovation as follows:

- **Level 1**: A simple improvement of the current technical system, no contradiction is resolved, the invention is the result of a compromise. Existing knowledge within the industry related to that system is sufficient.
- **Level 2**: Inventions including the solution of technical contradictions. It requires knowledge of various areas within an industry related to the system.
- **Level 3**: This is an invention that includes the resolution of physical contradictions. It requires knowledge from different industries.
- **Level 4**: This is novel technology development. It is developed by using breakthrough solutions that require knowledge from different scientific fields. This level is also an improvement of a technological system, but without addressing an existing technological issue. On the contrary, it improves functionality by replacing existing technology with a novel one. For instance, a mechanical system is replaced with a chemical system to perform that function.
- **Level 5**: It involves the discovery of novel phenomena. The novel phenomena can lead the existing technology to a higher level.

In the thesis, we mainly focus on the work addressing the target problem by using the level 3 innovation.

3.3 Contradiction

The concept of contradiction is one of the two significant parts of TRIZ (together with laws of evolution). It is applied in any TRIZ problem-solving process. TRIZ states that to obtain creative solutions, contradictions must be eliminated, without allowing for compromise or optimization, and introduces the principle of formulating and eliminating contradictions systematically. There are two typical types of contradictions classified by (Domb, 1997; Rousselot, Zanni-Merk, and Cavallucci, 2012).

- **Technical Contradiction** A technical contradiction describes the state of a system in which one action causes a useful effect, but also creates an undesirable effect at the same time. It is a typical engineering trade-off when something becomes better but another thing becomes worse. In fact, this occurs when it attempts to improve certain properties or functions of a system but results in the deterioration of other properties in the system (Yan, 2014). For instance, a better car brand can achieve a more comfortable situation during driving. Unfortunately, it also contributes to spending more money to afford it. Therefore, it becomes to achieve a trade-off between how comfortable the situation the consumer wants and the price that consumer can afford.

- **Physical Contradiction** A physical contradiction resolves the part of the technical contradiction centered on the parameter, which must contain two opposite values at the same time. This occurs when there are inconsistent requirements for the physical conditions of the same system. For instance, the big size of a laptop screen can provide a better watching experience but may make it too heavy to carry around. Therefore, the screen's size values (big screen for watching and small screen for space occupancy) present a physical contradiction.

3.4 The TRIZ Knowledge Sources

When dealing with technical contradictions, the TRIZ knowledge sources for solving inventive problems include forty inventive principles. Other TRIZ techniques like seventy-six inventive standards, and eleven separation methods for removing technical contradictions, provide problem-solving solutions and remove physical contradictions respectively. The full contents of

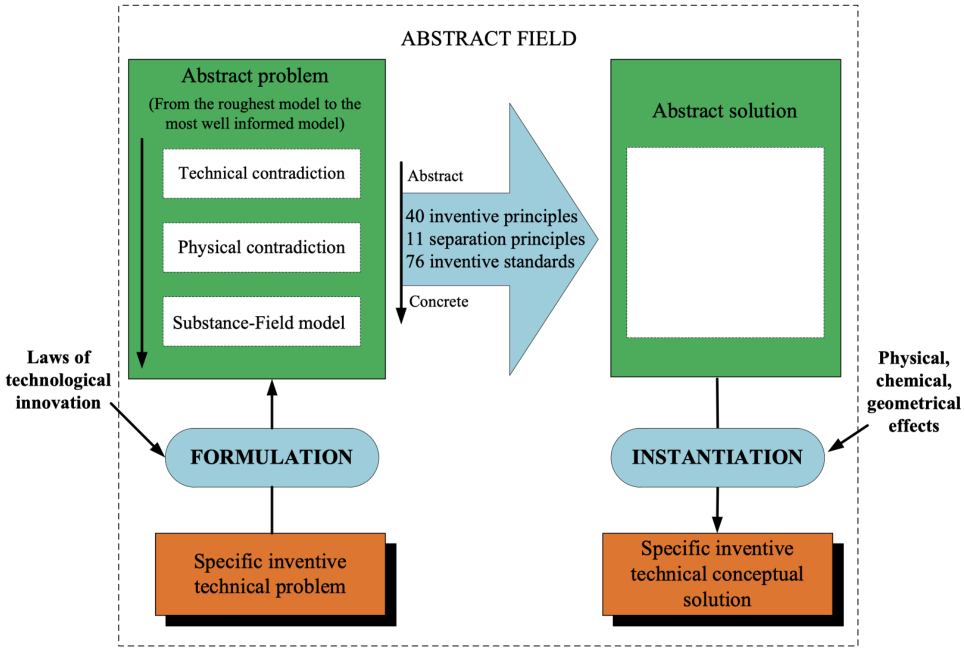


FIGURE 3.2: The processing of solving inventive problems via classical TRIZ

these techniques are presented in Appendix A, Appendix B, and Appendix C separately.

- **Inventive Principles** Forty inventive principles are a type of tool used to find out innovative and creative solutions for target problems. They derive from the research of TRIZ and patent analysis in order to solve the technical contradictions. Altshuller proposed thirty-nine generic engineering parameters (Appendix D), such as 'weight of moving objects' or 'speed', and created a contradiction matrix (Appendix E) to make the inventive principles (Appendix A) applicable in a systematic way.

- **Standard Solutions** Most inventions refer to conceptual modifications of physical systems. Therefore, there should be several common approaches for solving problems that apply to the entire group of similar inventive problems. These problems are similar when problems from different fields produce the same physical model. Thus, solving system problems do not always need to identify contradictions. Seventy-six standard solutions (Appendix B) are often applied to correct undesired interactions between two parts of a system.

With all of these TRIZ knowledge sources, a classical problem-solving process by using TRIZ can be illustrated in Fig. 3.2 (Yan, 2014). In detail, experts should first present the target problem in the form of contradiction by using different tools. After that, they should achieve abstract solutions by using different knowledge. Eventually, with the help of other domains knowledge base like physical, chemical, or geometrical fields, abstract solutions can be instantiated to achieve one or more concept solutions to be implemented in real cases.

3.5 IDM-related Knowledge

The Inventive Design Method (IDM) is based on the Theory of Inventive Problem Solving (TRIZ). It represents an extension of TRIZ and is perceived as more structured and less ambiguous. It is, therefore, easier to teach to others since it is more formally described. It aims at assisting companies and engineers to solve complex and multidisciplinary problems in creative ways.

Different from other ontologies, IDM ontology is generic and applicable in all fields (Bultey, De Bertrand De Beuvron, and Rousselot, 2007). Furthermore, (Cavallucci, Rousselot, and Zanni, 2010) proposed the main concepts of IDM that are problems, partial solutions, and contradictions including element parameters and values. In patents, problems normally describe unsatisfactory features of existing methods or situations. Partial solutions provide improvements or changes to the defined problems. Each problem may cause one or more contradictions the patent solves. Besides, partial solutions must be the simplest possible. Elements are components of the system and parameters qualify the element with certain specifics. Parameters are also qualified by values. The knowledge contained in these key concepts of patents usually has great value for engineers.

Thus, in this thesis, we try to maximize this knowledge, hidden in patents' unstructured text, to find out relevant potential inventive solutions to a given problem in order to assist creative R&D activities.

3.6 Implementation of TRIZ on the Problem Solving

Several models and extensions based on TRIZ have been proposed by researchers. A TRIZ-based patent knowledge management system called TPKMS is proposed by (Ding et al., 2017) to explore the possible inventive principles for solving problems. The structural information of patents, TRIZ theory, and 40 inventive principles are combined to help researchers explore the process of construction innovations. TPKMS requires that users have the corresponding domain knowledge to explore inventive solutions. Moreover, (Rahim et al., 2018) proposed an approach using the computational thinking model and TRIZ methodology to enhance patent innovativeness. Identifying patterns in the computational thinking model enables the generation of a possible solution. Based on the exploration of trend pattern recognition on the paper proposed, users could explore the inventive solutions among different domains.

Besides, a framework is proposed by (Cavallucci, Rousselot, and Zanni, 2011) which aims at extracting and representing the know-how of domain experts and populating an already constructed ontology of inventive design. (Yan, Zanni-Merk, and Rousselot, 2011) proposed a method to calculate the semantic distance between short texts and use it to fill the semantic gap between the parameter and the generalized one and to facilitate the use of inventive design techniques. A formal contradiction model applicable to inventive design is proposed by (Rousselot, Zanni-Merk, and Cavallucci, 2012) to promote the related software development. A method based on a synergy between the theory of inventive problem solving and case-based reasoning (Houssin et al., 2015) is proposed by (Negny et al., 2012) to support engineers in preliminary design. (Yan et al., 2013) also presents an inventive method to facilitate the use of the contradiction matrix, using a semantic similarity approach and case-based reasoning.

However, for the aforementioned approaches based on TRIZ, an obvious drawback is that users still need to master the complexity of TRIZ methodology to make full use of TRIZ potential. It becomes a significant challenge for most newcomers to use it. How to automatize the inventive solutions retrieval for users who do not possess TRIZ knowledge is the main purpose of this thesis.

3.7 TRIZ Improvements with AI Techniques

To our best knowledge, there are only a few works related to TRIZ improvements with the help of AI techniques. In this section, we comprehensively introduce these research works.

- For the understanding of AI in the environment of construction, (Hoch and Brad, 2020) proposed to combine TRIZ and Six Sigma tools in order to assist the application of selected AI technologies to the right business process or even business model in a collaboration.

- To address the reliability issue of neural networks' hidden layers, a quantum function called QuantumReLU (QReLU) according to the classical activation function ReLU is proposed by (Guehika, 2019), which is the use of TRIZ principle 35 (change of parameters) from 40 inventive principles (Appendix A) and seventy-six standards solutions (Appendix B).

- (Souili, Cavallucci, and Rousselot, 2015b) proposed to use classical NLP techniques that the finite state automate, a tool for the representation of linguistic phenomena, and XML tags to capture the IDM-related knowledge from patent documents. Furthermore, (Souili, Cavallucci, and Rousselot, 2015a) classify linguistic markers into super-markers (*improve*, *deteriorate*, etc.) and polyvalent markers (*allow, change, increase,* etc.) to identify problems and partial solutions in patent documents.

- Aiming to perform the trend analysis of TRIZ, (Yoon and Kim, 2011) proposed to combine the extraction of lexical binary relations in patent documents like 'adjective + noun' or 'verb + noun' forms and measuring semantic sentence similarity between the binary relations from patents and the binary relations in the rule base, through the natural language processing approach.

- (Park, Yoon, and Kim, 2013) proposed an approach to automatically identify the promising patents for technology transfers by adopting TRIZ evolution trends and Subject–Action–Object (SAO)-based text-mining techniques.

- For classifying patents into several categories of inventiveness, a novel framework based on computational methods is presented by (Li et al.,

2012) according to the level of invention (LOI) and use of artificial neural networks. It can eventually avoid the laborious manual effort required for assigning LOI to each patent.

- In order to automatically extract potential contradictions in patent documents and merge them into the TRIZ matrix, (Berdyugina and Cavallucci, 2021) applied the antonyms identification technique of NLP to facilitate the application of the TRIZ tool for practical problems.
- An extraction approach based on patent semantic space mapping through adopting the Doc2vec model is proposed by (Zhai, Li, and Cai, 2020) to identify the technical contradictions in patent documents.
- (Guarino et al., 2020; Guarino, Samet, and Cavallucci, 2020) proposed an end-to-end patent analysis algorithm called SummaTRIZ according to the BERT model to summarize the contradiction sentences in patent documents.

With the aforementioned works, we especially summarize them as Tab. 3.1. For these recent works, even if there are not numerous efforts in this field using AI technologies, several researchers have been managing to assist TRIZ improvements in different directions, like automate of the contradiction retrieval (Guarino, Samet, and Cavallucci, 2020; Berdyugina and Cavallucci, 2021; Guarino et al., 2020), with various NLP approaches. Compared to these works, nevertheless, we notice that few works aim to automate the innovative solutions retrieval from a large number of patent documents. Several works still need to use complex TRIZ approaches like TRIZ principles (Guehika, 2019) or Six Sigma tools (Hoch and Brad, 2020). Indeed, the work of (Souili, Cavallucci, and Rousselot, 2015a) inspires us to further explore the automation of TRIZ knowledge and facilitate innovative solutions retrieval among different domains patents.

3.8 Summary

In this chapter, as the background of our research, we mainly introduced the classical problem-solving theory, TRIZ. It includes the TRIZ background, the general knowledge about TRIZ. Indeed, TRIZ-related knowledge and research works inspire us to explore the use of the problem-solving ontology in our task, preparing inventive solutions from different domains patents. First of all, in this chapter, the definition of TRIZ, different levels of innovation, the

TABLE 3.1: TRIZ approaches with the assistance of AI techniques

Paper	TRIZ	AI Techniques	Purpose
Hoch and Brad, 2020	TRIZ and Six Sigma	No	Assistance
Guehika, 2019	TRIZ inventive principles and standards solutions	A quantum function QReLU based on ReLU activation function	Problem-solving
Souili, Cavallucci, and Rousselot, 2015b	IDM-related knowledge	Finite-state automate and XML tags	Information retrieval
Yoon and Kim, 2011	TRIZ	Lexical binary relations and semantic sentence similarity of NLP	Analysis
Park, Yoon, and Kim, 2013	TRIZ evolution trends	Subject-Action-Object (SAO)-based text-mining technique	Information Retrieval
Li et al., 2012	TRIZ	Artificial neural networks	Classification
Berdyugina and Cavallucci, 2021	TRIZ matrix	Antonyms identification technique of NLP	Information retrieval
Zhai, Li, and Cai, 2020	TRIZ contradiction	Doc2vec	Information retrieval
Guarino et al., 2020	TRIZ contradiction	BERT	Information retrieval

contradictions in the TRIZ problem solving, TRIZ knowledge sources, IDM-related knowledge deriving from TRIZ, and TRIZ implementations on the problem solving are introduced. In the end, we especially introduced recent TRIZ works using the assistance of AI techniques, and concluded the drawbacks of these works compared to this thesis. In the next chapter, we will introduce our similar problem retrieval work. Several aforementioned deep learning approaches in Section 2.2 will be applied to compute the similarity value of pair-wise sentences in the process of retrieving similar problems from different domains patents.

Chapter 4

Similarity-based Approaches for the Problem Retrieval from Different Domains Patents

As illustrated in Fig. 1.2 in Chapter 1, for preparing innovative solutions from different domains patents towards the target problem, the first step is to mine similar problems in different domains patents. Most importantly, our works are built on the following hypothesis:

Hypothesis 1 *The corresponding solutions of problems from different domains patents could represent potential innovative solutions for the target problem if these problems are similar enough.*

Indeed, five levels of inventiveness are proposed by (Altshuller, 1984) and introduced in Section 3.2. Level 1 and level 2 aiming to significantly improve the technical system within an industry are not considered in the thesis. Level 3 which is an invention containing a resolution of a physical contradiction focuses on the thesis. It requires different industrial knowledge.

Therefore, in this chapter, we especially introduce two models named IDM-Similar based on Word2vec neural networks in Section 4.2 and SAM-IDM based on LSTM neural networks in Section 4.3. They are mainly applied to extract similar problems from different domains patents. Moreover, SAM-IDM outperforms IDM-similar on the similar problem retrieval task. We especially present the compared experimental results in Section 4.4 then discuss and conclude all the introduced approaches in the last section.

4.1 IDM-related Knowledge Extraction

We firstly introduce the tool named Patent Extractor (Souili, Cavallucci, and Rousselot, 2015a). In the thesis, it is leveraged to retrieve IDM-related knowledge (Section 3.5) from patent documents. Indeed, lexico-syntactic patterns are used to design this tool. Problems, partial solutions, and parameters can be extracted from patent documents.

In detail, generic linguistic markers are combined into the tool to act as keywords in order to identify and retrieve IDM-related knowledge. The identification of concepts with IDM-related knowledge requires examining their context in patent documents to derive clue words or markers (Souili and Cavallucci, 2017). Patent Extractor learns therefore to choose linguistic markers to understand how patent documents present problems and partial solutions, through finding regularities between the information structure and morphosyntactic contained in patent documents. For instance, for extracting problems, the structure like "*A problem with ... is that ...*" or linguistic markers representing the negative sentiment such as "*damage, cause, harm*, and *serious* etc" are mainly used via Patent Extractor. We define IDM-related knowledge that is extracted by Patent Extractor as follows:

Definition 1 (IDM-related knowledge) *The retrieved IDM-related knowledge from a patent of the index i i.e., $P_i \in \mathcal{P}$ is a triplet (Po_i, Ps_i, Pa_i) in which Po_i, Ps_i ,and Pa_i are respectively the sets of problems, partial solutions, and parameters.*

From a single problem, we construct the bag with meaningful words. For the j-th problem Po_{ij} of patent P_i, the bag of meaningful words is $Po_{ij} = \{Po_{ij}^1, Po_{ij}^2, ..., Po_{ij}^{|P_{ij}|}\}$ where $Po_{ij}^{|P_{ij}|}$ is the $|P_{ij}|$-th word. Meaningless words like stop words, punctuations are removed.

Definition 2 (Candidate solutions) *Assuming j-th problem Po_{ij} from the patent i, the set of candidate patent solutions $\mathcal{C}_{Po_{ij}}$ is to the target problem. The latter set contains all partial solutions of a patent containing at least one problem Po_{lh} similar to Po_{ij} with regards to a similarity threshold σ set by the downstream task. Formally,*

$$\mathcal{C}_{Po_{ij}} = \{P_l = (Po_l, Ps_l, Pa_l) \in \mathcal{P} | \exists Po_{lh} \in Po_l, sim(Po_{ij}, Po_{lh}) \geq \sigma\} \tag{4.1}$$

Example 1 *Assuming the patent i US8847930B2*[1] *named "Electrically conductive touch pen". The IDM-related knowledge is as follows,*

[1]reader which may refer to this link for the full patent `https://patents.google.com/patent/US8847930B2/`

- *Po_{i1}: This would hamper a user's ability to operate the touch pen 10 with gloves. Pa_{i1}: glove.*
- *Po_{i2}: A problem with rubber containing carbon sufficient for conductivity is that it may leave black marks on substrates to which it comes into contact. Pa_{i2}: black mark.*
- *Ps_{i1}: The inner molding 29 is replaced by a former 39 that is ideally metallic. This alternative embodiment is designed to address the aforementioned problems attendant to a user wearing gloves. Pa_{i1}: metallic.*
- *Ps_{i2}: Coating the rubber, or selectively the rubber tip, with a very thin layer of Parylene. Pa_{i2}: parylene.*

We can see that, in patent i, the partial solution Ps_{i1} of the former 39 made by metallic can address the problem Po_{i1} of operating the touch pen with gloves. The partial solution Ps_{i2} of a very thin layer of Parylene can solve the problem Po_{i2} of black marks. Corresponding parameters describe the key element with certain specificity. Indeed, each patent contains several IDM-related knowledge. In this chapter, we aim to compute the similarity between pairwise problems (problem Po_{ij} and problem Po_{lh}) from different patents.

4.2 IDM-Similar Model based on Word2vec

We introduce the IDM-Similar model that is mainly used to mine similar problem sentences from patent documents in this section. Indeed, Word2vec neural networks (Mikolov et al., 2013a) and cosine similarity metric are used to compute similarity among problems in a wide range of domain patents. According to our postulate, this work aims to achieve similar problems from patents. Experiments illustrate that the IDM-Similar model is a promising alternative to the classical TRIZ. Engineers are thus able to associate their problems in a specific domain to solutions from another domain's patents.

4.2.1 IDM-Similar

IDM-Similar aims to find out similar problems to a given problem from the large-scale patent corpus in order to merge IDM-related knowledge. We first extract problems, partial solutions, and parameters from patent corpus via Patent Extractor. Then, we compute similarity values between these problems. Formally, the extracted IDM-related knowledge set $P_i = \{Po_i, Ps_i, Pa_i\}$.

Given the j-th problem $Po_{ij} = \{Po_{ij}^1, Po_{ij}^2, ..., Po_{ij}^{|P_{ij}|}\}$ in the i-th patent document where $|Po_{ij}|$ is the $|P_{ij}|$-th word in the j-th problem sentence, and we compute its similarity with other considered problems P.

- **Word Vector** Word2vec neural networks are used to compute the vector of each word in the training corpus. As a word embedding model, these two-layer neural networks can be trained via a large-scale corpus to achieve the vector in space for each unique word in the corpus. Word vectors are positioned in the vector space such that words sharing a common context in the corpus are close to each other in space (Mikolov et al., 2013b). The trained Word2vec neural networks simplify the processing of the chosen text as the vector operation in the n-dimensional space. Thus, the similarity in the vector space can represent the semantic similarity of the text. Moreover, the training process of Word2vec is unsupervised, and two-layer neural networks transform the text into the digital form that neural networks can understand. These shallow neural networks can run thus efficiently on the computer. As illustrated in Fig. 4.2. Continuous Bag-of-Words (CBOW) and Skip-gram (Mikolov et al., 2013b) are two sorts of model architectures that are used in Word2vec to produce the distributed representation of words. These two architectures are similar at the algorithmic-wise, but the main difference between them is that CBOW predicts the target word according to the context words around the initial word. On the contrary, Skip-gram predicts each context word via the target word. In our model, we apply Word2vec with Skip-gram due to the advantage of Skip-gram on infrequent words.

- **Sentence Vector** As shown in Fig. 4.1, Patent Extractor retrieves matched problem sentences from patent documents. Sentence vector $\overrightarrow{\mathbf{Po}}$ is then produced via calculating the average vector of all words contained in each sentence. The function is defined as:

$$\overrightarrow{\mathbf{Po}} = \frac{\sum_0^{|P_{ij}|} \overrightarrow{\mathbf{Po}_{ij}}}{|P_{ij}|} \tag{4.2}$$

- **Cosine Similarity** The cosine distance between the given problem sentence vector $\overrightarrow{\mathbf{Po}_L}$ and the vector of another problem sentence $\overrightarrow{\mathbf{Po}_J}$ is computed as:

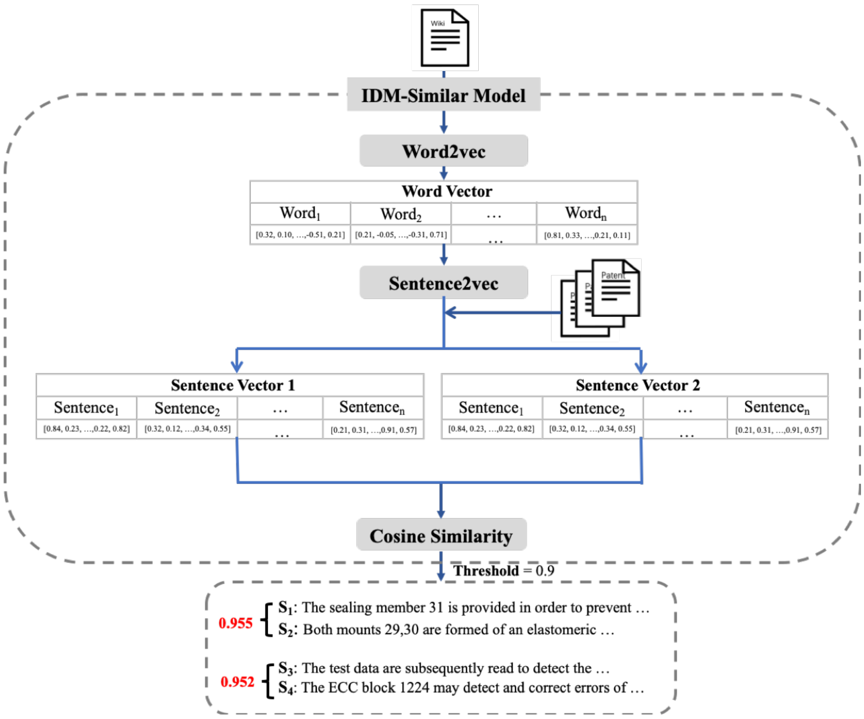


FIGURE 4.1: An overview of IDM-Similar

$$CosineDistance = \frac{\overrightarrow{\mathbf{Po}_L} \cdot \overrightarrow{\mathbf{Po}_J}}{\left|\overrightarrow{\mathbf{Po}_L}\right| \left|\overrightarrow{\mathbf{Po}_J}\right|} = \frac{\sum_{l,j=1}^{n} \mathbf{Po}_{il} \times \mathbf{Po}_{ij}}{\sqrt{\sum_{l=1}^{n} \left(\mathbf{Po}_{il}\right)^2} \times \sqrt{\sum_{j=1}^{n} \left(\mathbf{Po}_{ij}\right)^2}} \tag{4.3}$$

Next, the cosine similarity is defined as:

$$CosineSimilarity = 1 - CosineDistance = 1 - \frac{\overrightarrow{\mathbf{Po}_L} \cdot \overrightarrow{\mathbf{Po}_J}}{\left|\overrightarrow{\mathbf{Po}_L}\right| \left|\overrightarrow{\mathbf{Po}_J}\right|} \tag{4.4}$$

In general, the similarity between sentence pairs increases when their cosine similarity value is closer to 1.

In summary, as shown in Fig. 4.1, Word2vec neural networks are firstly used to achieve each word vector. The trained word vector model is then used to generate the sentence vector for each input problem sentence. We eventually apply the similarity computation approach to compute the cosine similarity among pairwise problem sentences to identify similar problems with values greater than a preset threshold along with their corresponding solutions and parameters.

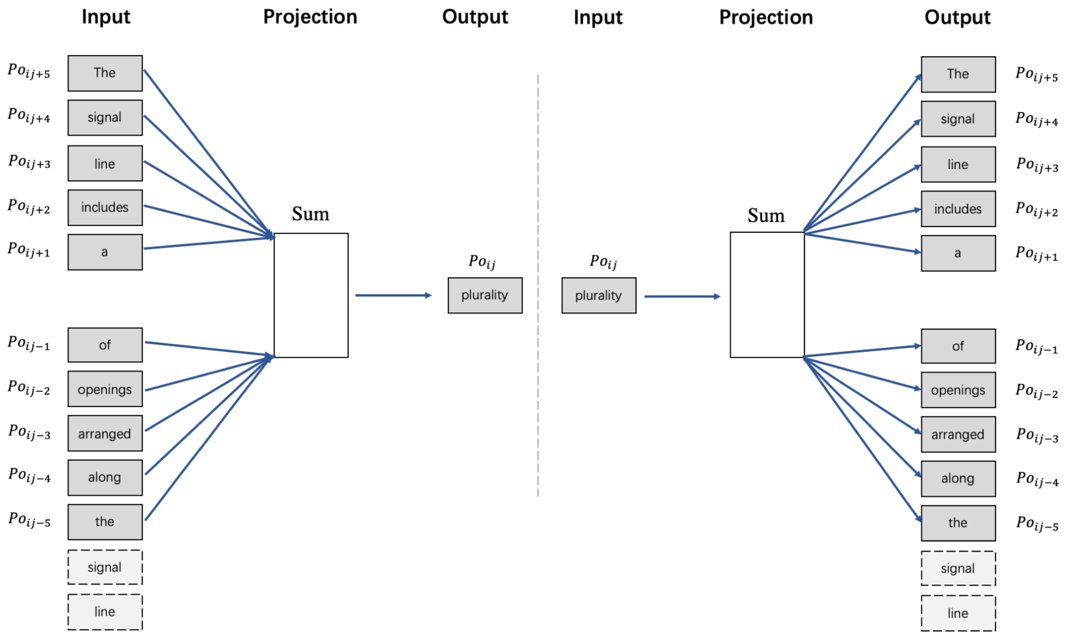


FIGURE 4.2: Architecture of Word2vec model: CBOW (left) and Skip-gram (right)

4.3 SAM-IDM Model based on LSTMs

We notice that for predicting similar problems from different domains patents accurately, the model's ability to learn long context information is significant. Specifically, patent documents contain much longer sentences compared to generic texts. In this section, in order to further improve this ability, we therefore introduce a novel problem retrieval model named SAM-IDM in detail. Similarity-based Approach for Merging IDM-related knowledge (SAM-IDM) aims at preparing inventive solutions by extracting similar problems from a large number of patent documents. SAM-IDM, as IDM-Similar, is based on our hypothesis.

The framework of SAM-IDM is illustrated in Fig. 4.3. At the first step, a period of several patent documents from USPTO are randomly chosen to build a patent database in XML format. Patent Extractor is then used to extract IDM-related knowledge (problems, partial solutions, and parameters) from input patents in the second step. At the third step, we design a reduction strategy to assign IDM-related knowledge into different groups according to different domains they belong to. It aims to only access inventive solutions with level 3 through retrieving latent inventive solutions from various industrial domains. This reduction strategy is also able to enhance the performance of the sequential similarity computation by decreasing the search

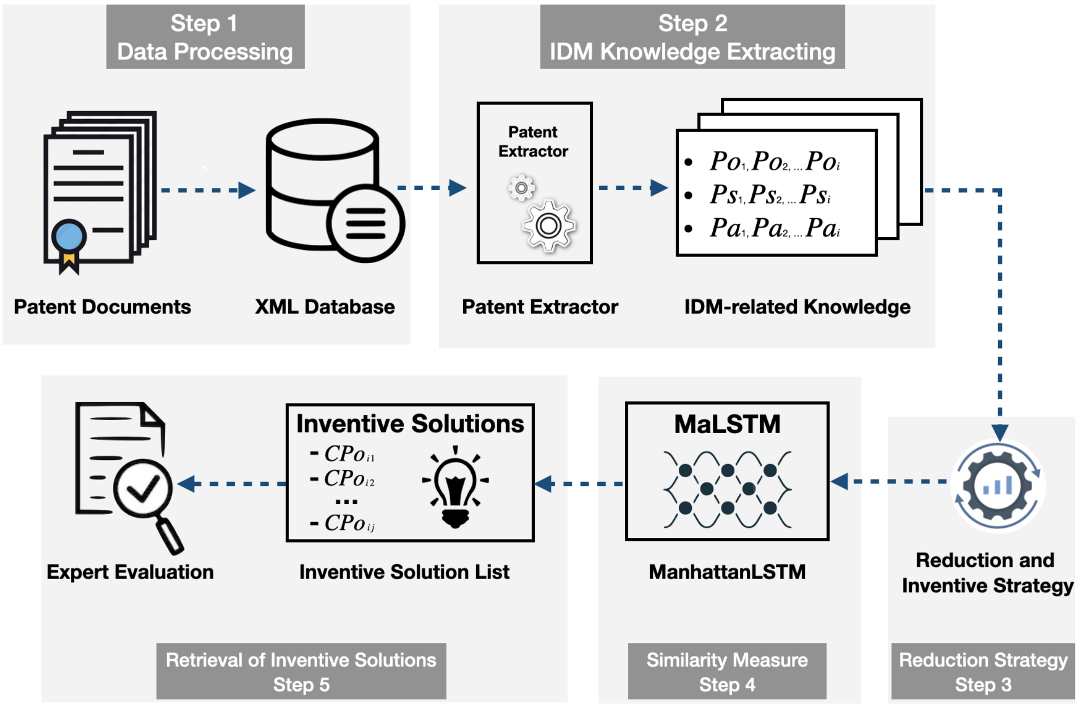


FIGURE 4.3: The framework of SAM-IDM

set. After that, in the fourth step, the trained MaLSTM (Mueller and Thyagarajan, 2016) is used to learn sentence semantic meanings in order to predict similarity among different domains problems. Several similar problems are eventually listed and ranked via their similarity scores. Corresponding partial solutions towards different domain problems can be seen as potential inventive solutions towards target problems. Different industrial experts evaluate the final results.

- **Reduction of IDM-related Knowledge Set** : At step 4 of SAM-IDM, a large number of problem sentences will be performed. This could lead to a significant efficiency issue for computing similarity among sentences. For instance, one of the problem sentences is chosen as the target problem. If we compare this target problem with all the remaining problem sentences, consequently computational consumption is not affordable. In practical usage, extensive computation is a waste of time for companies and engineers. Therefore, following the hypothesis, as illustrated in Fig. 4.5, we specifically divide IDM-related knowledge into several groups according to the different domains they belong to, so that the computational consumption among the same domain patents can be avoided. In detail, we compare the target problem Po_{ij} of patent

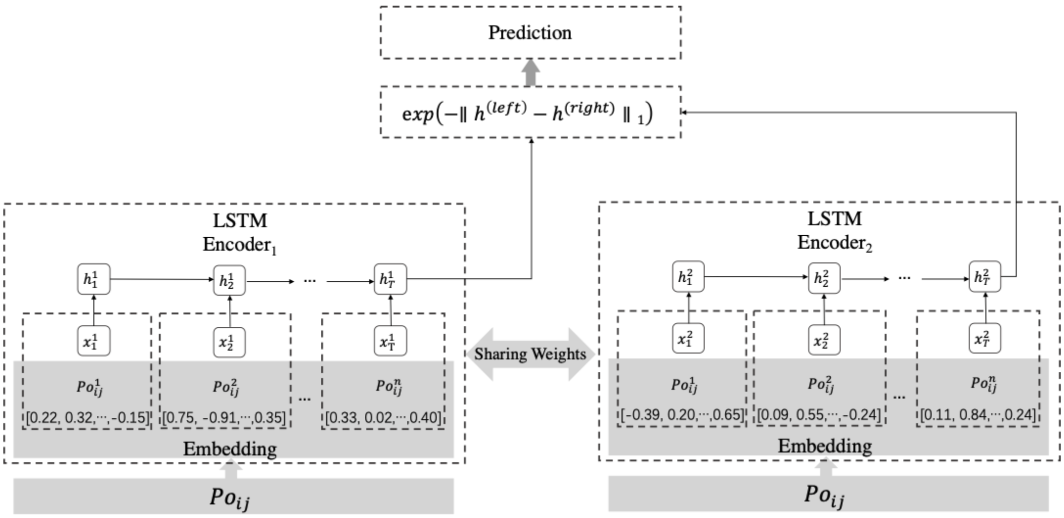


FIGURE 4.4: The structure of MaLSTM

i from $Domain_{d-1}$ with other problems which are from different domains like $Domain_d$. Since some domains tend to contain a much larger number of patents, the proportion of problems in this domain is consequently larger than other domains problems. As illustrated in Table 4.2, the physics domain contains a larger number of problems compared to other domains. In another word, performing a one-time comparison with the target problem in the physics domain to all of the problems in other domains will contribute the same time consumption as comparing once within the physics domain. Thus, avoiding this part of the comparison is also significant to the performance of the model.

- **Manhattan Long Short Term Memory Networks for Similarity Measure of Problem Sentence**: In SAM-IDM, Manhattan LSTM (MaLSTM) (Mueller and Thyagarajan, 2016) is applied to perform the semantic similarity computation among different problems. As shown in Fig. 4.4, MaLSTM consists of two identical LSTM networks (LSTM $Encoder_1$ and LSTM $Encoder_2$). Same weights are shared in these two LSTM networks to decrease the training time of the model, because of its siamese recurrent architecture. In MaLSTM, the identical sub-network LSTMs can learn representations of problem sentences Po_{ij} via sequences of word vectors x_T. Furthermore, the word embedding approach provides the semantic meaning to each word $Po^{\bullet}{}_{ij}$ of the problem sentence in a vector representation. Besides, in SAM-IDM, the embedding approach Word2vec that is trained by the open-source Wikipedia dataset

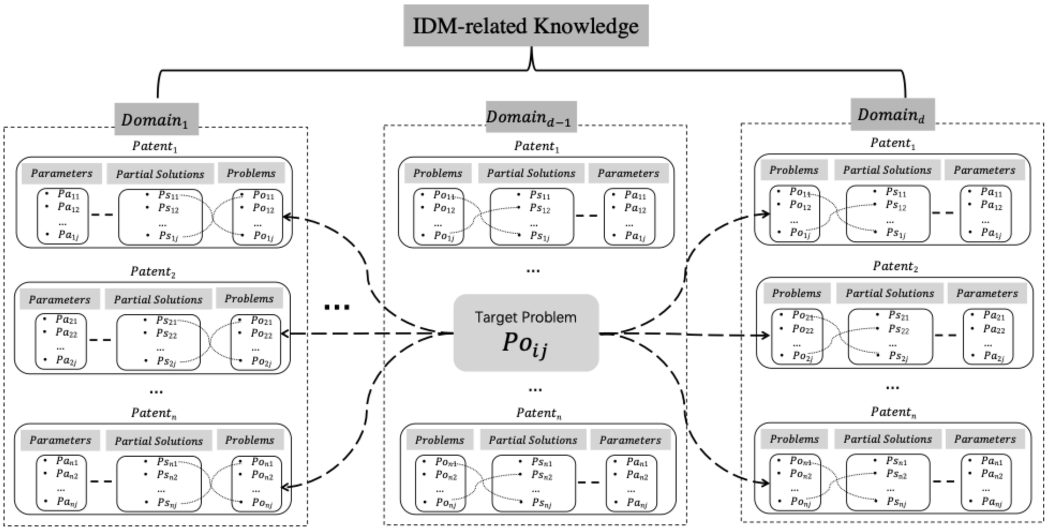


FIGURE 4.5: Reduction of IDM-related knowledge sets

in Section 4.4.4 is applied to achieve the embedding matrix of the given problem sentences. The hidden state at each sequence index is then updated by LSTMs through functions (3)-(8). After that, h_T outputs a hidden state encoding sentences' semantic meanings. We apply the labelled Quora dataset containing a large number of similar Quora question pairs to feed LSTMs in order to let it learn hidden semantic representations between labelled similar pairwise sentences. The similarity of the representation space is subsequently used to infer the underlying semantic similarity of the sentences. The Manhattan distance is eventually used to measure semantic similarity among problems Po_{ij} from different domains patents.

4.4 Experiments

In this section, we detail datasets involved in the thesis, computation settings, experimental settings of IDM-Similar and SAM-IDM, compared experimental results, and use cases.

4.4.1 Real-World Test Dataset

We choose real-world U.S. patents as our test dataset. Indeed, U.S. patents can be classified under three types: design patents, utility patents, and plant patents. Under the United States Patent and Trademark Office (USPTO), a

TABLE 4.1: Performance of Patent Extractor on U.S. utility patents

Problem	Partial Solution	Parameter	Category
4,574	17,971	29.264	8

TABLE 4.2: Distribution of problems in different domains

Problem \ Domain	HN	PO	C	T	FC	ME	P	E
4,574	652	414	370	26	70	245	1,558	1,239

utility patent is granted to any person who invents or discovers any new and useful process, machine, article of manufacture, or combination of substances, or any new and useful improvement thereof. A design patent is granted to any person who invents a new, initial and ornamental design for an article of manufacture. A plant patent is granted to anyone who invents or discovers and asexually propagates any unique and new variety of plant. However, 90% of U.S. patents are utility patents, which protect the practical or functional aspects of an invention, compared to other types of patents, which are usually somewhat similar in the field of invention. Therefore, a utility patent dataset[2] containing 6,161 patent documents is chosen as the test dataset to evaluate the performance of IDM-Similar and SAM-IDM on the real-world patent dataset.

In detail, as illustrated in Table 4.1, 4,574 problems are extracted from 6,161 patent documents by Patent Extractor. These problems are classified into eight industrial domains: fixed constructions (FC), human necessities (HN), textiles (T), physics (P), mechanical engineering (ME), chemistry (C), and electricity (E). We illustrate the distribution in Table 4.2.

4.4.2 Evaluation Metric

Finding the gold-standard ground truth for evaluating the similarity between different sentences has always been an open problem and challenge, especially for verifying the similarity between different domains problem sentences. It is always inherently subjective to assess the semantic similarity between pairwise sentences. So far, there is no work to address this issue. We, therefore, choose Stanford Natural Language Inference (SNLI) corpus, a benchmark labelled dataset (Toutanova et al., 2015), to evaluate our models.

[2] https://bulkdata.uspto.gov/data/patent/grant/redbook/fulltext/2017/

In addition, to illustrate the objectivity of the patent dataset, a small size of manual labelled data sample with 1,121 similar patent pairwise sentences [3] is used as another validation dataset. This labelled data sample is performed with the help of two industrial experts. Besides, four experts from the mechanics, engineering, physics, and chemistry domains are invited to cross evaluate the final experimental results on the real-world test dataset (U.S. patents).

4.4.3 Computation Settings

To compare the performance of different approaches in terms of computational consumption, we evaluate them using a labelled sample dataset, as illustrated in Table 4.4. MaLSTM in SAM-IDM is the most computational intensive approach in terms of computation time and memory occupation for a computer with a 4-core CPU and 16GB RAM. Nevertheless, it is still acceptable compared to other approaches. Moreover, the total computation time of MaLSTM on 2.8 million pairwise problems from 8 different domains is around 15 hours.

4.4.4 Experimental Settings

IDM-Similar

For achieving an ideal word vector model, we train our Word2vec neural networks with the clean English version of Wikipedia dataset [4]. Regular text is contained in the training dataset but it removes tables and links to foreign language versions. Besides, citations, footnotes, and markup are also removed and hypertext links are converted to the regular text. Optimization of the efficiency and accuracy for training the model is also performed. Table 4.3 illustrates the optimal parameters of Word2vec.

In detail, for parameters of Word2vec, *size* defines the dimension number of the created vectors. *size* presents hence that a 100-dimensional vector from the training phase is received by each document. More dimensions tend to slow training speed and overfitting issues could arise when the model performs on the small size dataset. The number of words that is included as context words around the target word is indicated by the *window*. *min_count* discards those words of training corpus when their frequency is smaller than

[3] https://drive.google.com/file/d/1JvrUuO4by_FzvyP-5gxKQAuyyQc9cadY/view
[4] http://mattmahoney.net/dc/textdata.html

TABLE 4.3: Parameters of the Word2vec model

Parameter	size	window	min_count	negative	sample	hs
Value	100	5	5	3	0.001	1

the threshold of frequency. This can filter out those extremely rare or misspelt words in the corpus. The hierarchical softmax is used as the loss function when we set $negative > 0$ and $hs = 1$. Besides, we set 3 noise words in the model with 3 of $negative$. The threshold of sampling is presented via $sample$. Words with higher frequency in the training corpus are randomly down-sampled. In addition, as introduced in section 4.2.1, pairwise sentences are more similar when their similarity values are closer to 1. But the final similarity threshold depends on the downstream task. By carrying out several tests on our task, We experimentally fix the similarity threshold at 0.9.

SAM-IDM

Parameter Setting: In SAM-IDM , we train the model on the open-source labelled Quora dataset[5]. The grid search determining the optimal parameters is also used in this work. For MaLSTM, the identical configuration with the same parameters and weights are shared in two identical LSTM neural networks. Parameter updates are mirrored in both sub-networks. To capture and learn semantic similarity of input pairwise sentences, a pre-trained 100-dimensional Word2vec model (Ni, Samet, and Cavallucci, 2019) is initially used to achieve word embeddings as inputs to LSTMs. More dimensions usually present slower training and can lead to the overfitting issue. Therefore, for LSTMs, we choose batch size among {500, **1000**, 1500, 2000} and epochs among {25, **50**, 75, 100} since a large number of training dataset. In addition, various parameters are set, including the dense units number as 50, validation split as 0.1, rate of drop dense as 0.25, rate of drop as 0.17, LSTM layers number as 50, and ReLU as the activation function. Furthermore, the Quora dataset contains 403,459 labelled similar pairwise sentences. 363,114 pairwise sentences are chosen as the training dataset and the validation dataset contains the remaining 40,345 pairwise sentences. As this dataset official notice states, the true meaning of a sentence is difficult to be determined with certainty. It is a "noisy" process for human labelling, and reasonable people will have different opinions. This large number of labelled

[5] https://www.kaggle.com/c/quora-question-pairs/data

TABLE 4.4: Performance of the computation consumption

	BOW	TF-IDF	Word2vec	MaLSTM
Computation Time	0.61s	0.63s	1.42s	6.25s*
Memory Consumption	8.34MB	4.72MB	3.21MB	10.24MB*

* MaLSTM is the most computation consumption model among them since its natural structure of using diverse hidden units to encode various characteristics of each sentence.

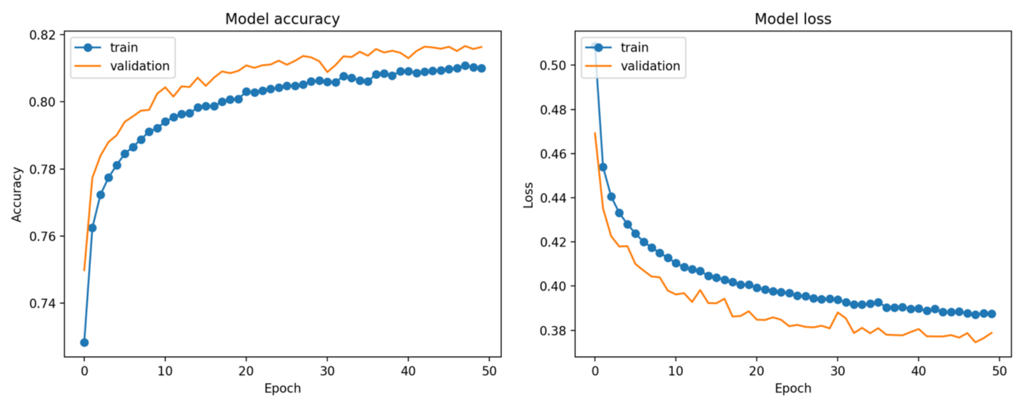


FIGURE 4.6: The illustration of learning curves

datasets might therefore include several mislabelling. Therefore, when training the model, we do not only avoid the overfitting issue but also manage to achieve the trade-off between underfitting and goodfitting to avoid learning numerous wrong features in incorrectly labelled datasets even under goodfitting conditions. It aims to let the model generalize well to future patent datasets. Therefore, as illustrated in Fig. 4.6, we stop training when a small gap between the training loss and validation loss remains. Indeed, they all decrease to stability. Besides, we set the maximum sequence length as {20, 30, 40, **50**} since sentences contained in patents usually have more tokens. Dropout rates of the LSTM encoder and dense layers are set to 0.17 and 0.25 respectively, to prevent overfitting of neural networks. The optimal parameters are highlighted in bold.

4.4.5 Experimental Results

In this section, we illustrate the performance of IDM-Similar and SAM-IDM on the labelled datasets. As illustrated in Table 4.5, different approaches are compared. MaLSTM in SAM-IDM generally outperforms BOW, TF-IDF, and Word2vec in IDM-Similar on the labelled datasets with different thresholds. We notice that, when the similarity threshold is 0.8, MaLSTM is capable

TABLE 4.5: Various approaches' experimental results on the labelled SNLI dataset (left) and the labelled patent pairwise sentence data sample (right) with different threshold values

Precision	Threshold											
	0.6		0.65		0.7		0.75		0.8		0.85		0.9	
	SNLI	Patent	SNLI	Patent	SNLI	Patent	SNLI	Patent	SNLI	Patent	SNLI	Patent	SNLI	Patent
BOW	62.45%	73.77%	60.24%	72.25%	60.04%	68.06%	61.24%	61.99%	62.25%	59.94%	61.24%	44.51%	62.05%	35.05%
TF-IDF	62.05%	72.70%	62.25%	67.88%	61.85%	66.63%	60.84%	51.65%	60.04%	49.86%	61.45%	42.01%	61.85%	26.27%
Word2vec$^{IDM-Similar}$4.2	43.78%	71.36%	45.38%	70.47%	47.19%	72.79%	49.60%	67.26%	56.22%	61.73%	61.85%	58.70%	62.45%	71.28%
MaLSTM$^{SAM-IDM}$4.3	70.28%	70.12%	71.49%	75.38%	73.69%	75.28%	72.49%	76.89%	**81.73%**	**77.88%**	76.51%	72.26%	77.11%	72.52%

of achieving promising experimental results like initial research results in (Mueller and Thyagarajan, 2016). For gaining the trade-off on both SNLI and Patent datasets, we eventually set the similarity thresholds of SAM-IDM as 0.8 and IDM-Similar as 0.9, according to our experimental results.

In addition, for SAM-IDM, 2.8 million pairwise problem matches in different domains are retained out of 10 million pairs of problem matches for 4,574 problems due to the reduction strategy. It eventually avoids two-third of the additional computational consumption for step 4. MaLSTM eventually extracts 327 similar pairwise problems from 2.8 million pairwise problems in 8 domains. With the expert evaluation in step 5, SAM-IDM achieves 78.59% precision on the chosen U.S. patent dataset. We separately introduce several use cases from IDM-Similar and SAM-IDM in the next section in detail.

4.4.6 Case Study

IDM-Similar

We illustrate use cases of IDM-Similar on extracting similar problems from different domains patents in this section. The performance of IDM-Similar can be assessed by two case studies among chemistry and mechanics domains as well as computer and physics domains.

1. Chemistry/Mechanics: US9537152: "Collector for bipolar lithium-ion secondary batteries" and US9532691: "Vacuum cleaner with the motor between separation stages" are two U.S. patents from chemistry and mechanics domains respectively. As illustrated in Fig. 4.7, IDM-Similar retrieves a pairwise similar problems: "*The sealing member 31 is provided in order to prevent contact between the current collectors 11 adjacent to each other inside the battery and prevent a short circuit caused by slight unevenness at edge portions of the single-cell layers 19 in the power generation element 21.*" and "*Both mounts 29,30 are formed of an elastomeric material and act to isolate the second dirt-separation*

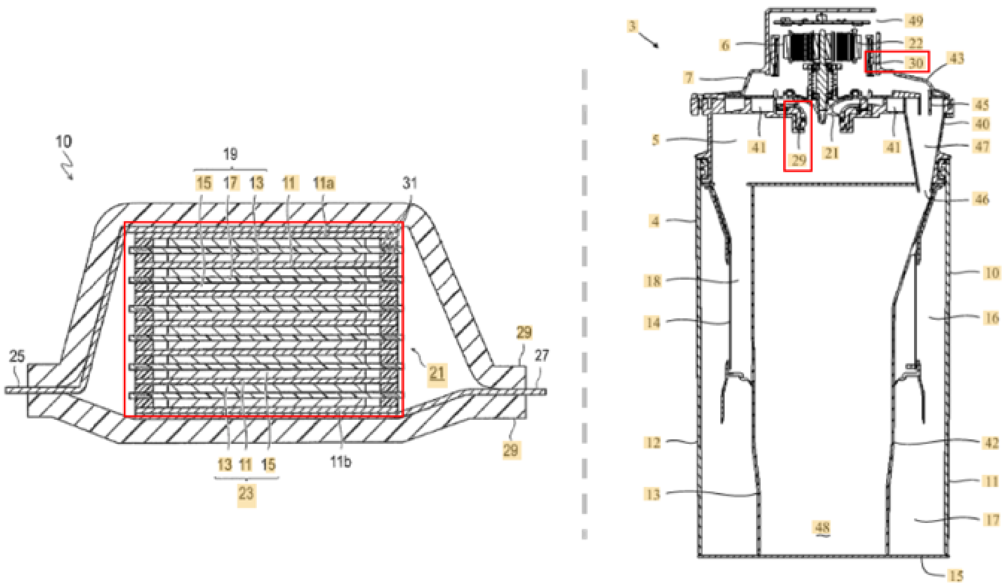


FIGURE 4.7: Diagrams of the sealing member (left) and the elastomeric material (right)

stage 7 and thus the remainder of the dirt separator 3 from the vibration generated by the vacuum motor 6.” After analyzing entire patents, experts think that these problems can be linked. It is possible to address the short circuit issue in the US9537152 patent with the corresponding solution of the elastomeric material in the US9532691 patent and vice versa.

2. Computer/Physics: "Hybrid-HDD with improved data retention (US9536619)" and "Semiconductor device and method of fabricating the same (US9536897)" are two U.S. patents from computer and physics fields respectively. As illustrated in Fig. 4.8, IDM-Similar extracts these two similar problems from different domains: "*The test data are subsequently read to detect the possibility of data retention errors that may occur when reading the associated user data.*" and *“The ECC block 1224 may detect and correct errors of data which are read out from the memory device 1210.”* After evaluating entire patents, we consider that adding the ECC block of the US9536897 patent into the left device is possible to address the data retention error mentioned in the US9536619 patent.

SAM-IDM

With the further improvement of our similar problems retrieval work by SAM-IDM, we manage to detail a typical use case that is extracted by SAM-IDM from real-world U.S. patents.

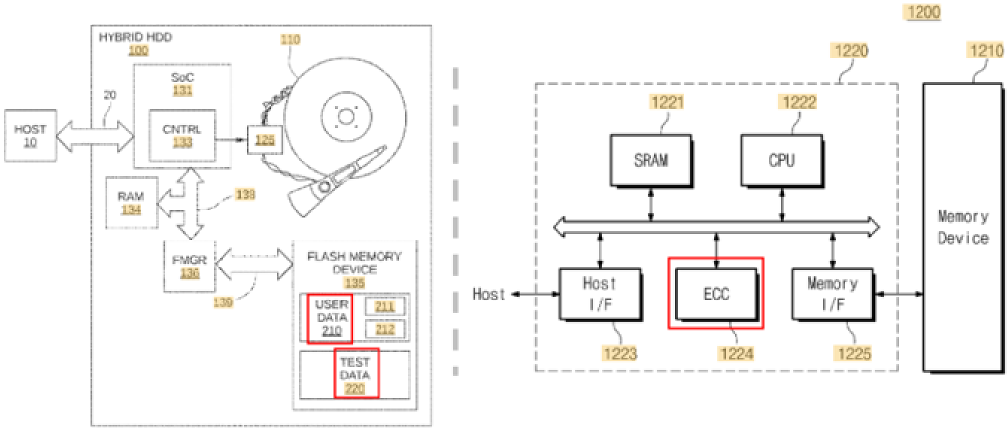


FIGURE 4.8: Diagrams of the hybrid HDD (left) and the memory systems (right)

Human Necessities / Electricity: US9532821: *"The locking mechanism may prevent the first screw member and the second screw member from pulling out of the first internal screw guide and the second internal screw guide."* US9536950: "*Such a structure of the channel region CH may contribute to preventing a short channel effect from occurring in the transistor TR.*"

These two similar problems come from the domains of human necessity and electricity. SAM-IDM defines the similarity value of 0.89 to them. As illustrated in Fig. 4.9, the US9532821 patent provides a bi-directional fixating transvertebral (BDFT) screw device. Inventors illustrate, in the patent, multiple device embodiments that combine the following dual functions in a single, independent structure: a) an intervertebral shelf spacer, which can be filled with osseointegration material to maintain disc height. b) a bidirectional fixation/fusion transforaminal screw device. This patent proposes a novel bi-directional fixed transvertebral (BDFT) screw/retainer device containing a mechanism of locking the screw in position by a vertical half brace. It can lock two adjacent screws in position and prevent retraction by inserting the vertical half brace into a novel indentation in the upper and lower part of the screw cassette that aligns the axial midpoint of the upper surface of the retainer between two adjacent internalized retainer screw guides/screws. These brackets can be easily snapped into the recesses of the cage and removed with a bracket tool. The primary function of this mechanism is applicable to any device requiring a screw locking mechanism, for example lumbar plates, and other orthopedic/medical devices requiring a screw locking mechanism.

In detail, problem "*The locking mechanism may prevent the first screw member*

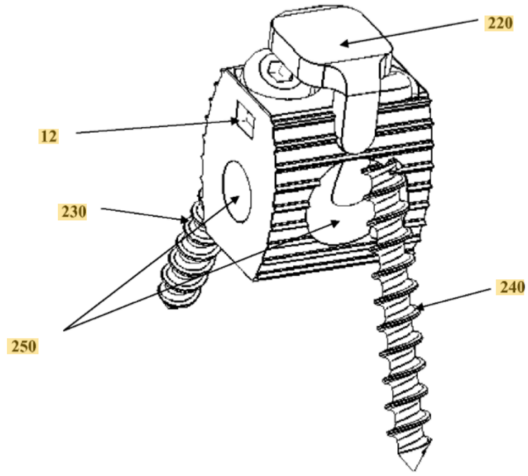


FIGURE 4.9: Frontal perspective view of the posterior lumbar elliptical design of the vertebral cage/BDFT construct (frontal isometric)

and the second screw member from pulling out of the first internal screw guide and the second internal screw guide" is as illustrated in Fig. 4.10. Screw member 30 and screw member 40 will be locked in their final position by their final rotation when the screw head is flush with the surface of the cage 10. The narrowing of the internal screw guides 190, 192 can be used as a preliminary screw locking mechanism by hugging the top of the screw/screw head interface (e.g., at its junction with the screw head). One vertical half bracket 120 covers the inside of the first two screws 130, 140 (or portions thereof) and the other vertical half bracket 120 covers the inside of the third and fourth screws 150, 160 (or portions thereof). When the bracket is caught and/or locked in the recess 194 of the cage, the screws of all four screws can be prevented from backing out or pulling out. These types of locking mechanisms are effective in avoiding the pulling out issue.

As illustrated in Fig. 4.11, a semiconductor device with a domain effect transistor and manufacturing approach is proposed by the US9536950 patent. The semiconductor device can be used to provide high-reliability electronic devices. From the patent, a strain relaxation buffer layer provided on a substrate that contains silicon germanium and a semiconductor pattern provided at the strain relaxation buffer layer that includes a source region is included by this novel semiconductor. As the problem "*Such a structure of the channel region CH may contribute to preventing a short channel effect from occurring in the transistor TR.*" mentioned in Fig. 4.12, the transistor is formed to have a gate-all-around structure.

The channel region CH is a nanowire structure with a width ranging from

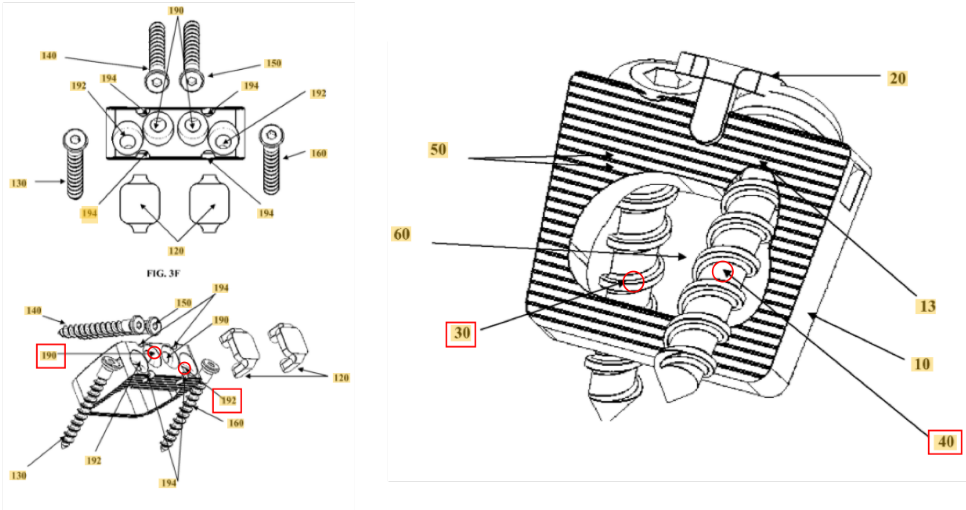


FIGURE 4.10: The illustration of the locking mechanism

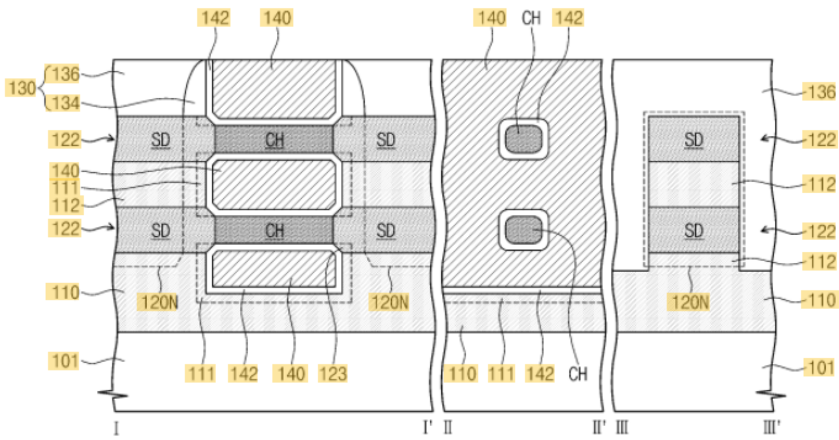


FIGURE 4.11: The illustration of a semiconductor device with a domain effect transistor.

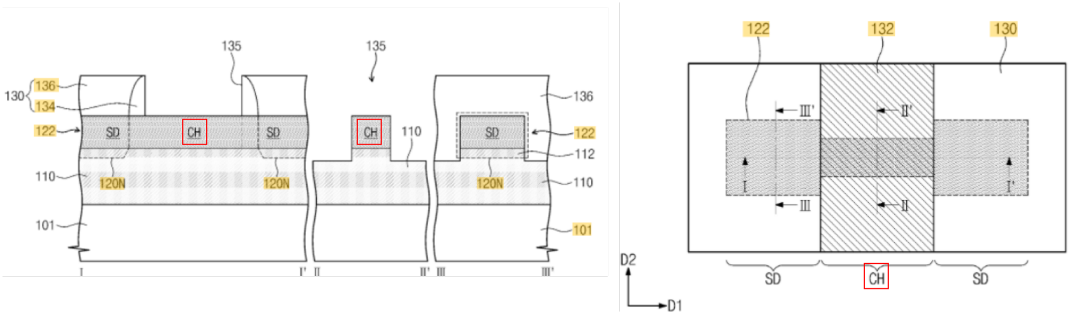


FIGURE 4.12: The illustration of a transistor.

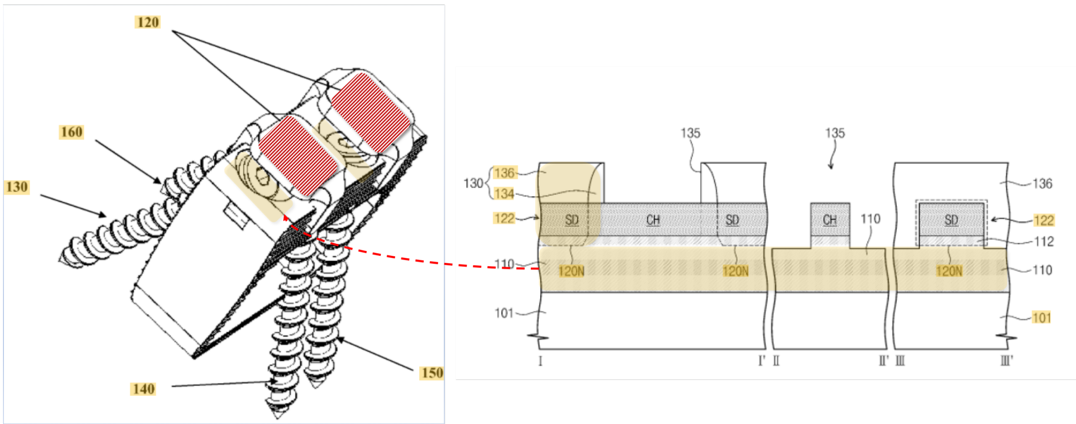


FIGURE 4.13: A supposition application solution.

a few nanometers to tens of nanometers. To avoid short channel effects, the strain relaxation buffer layer (SRB) 110 and the semiconductor layer 120 are formed sequentially on the substrate 101, as shown in Figure/reffig:7. The strain relaxation buffer layer 110 can have a recessed region adjacent to the channel region and the gate electrode extends into the recessed region to better avoid the short channel effect.

With experts' evaluation and analysis, the solution of the strain relaxed layer (patent US9536950) from the electricity domain might be a latent innovative solution to address the pulling out issue (patent US9532981) from the human necessities domain. A bi-directional fixating transvertebral (BDFT) screw device in patent US9532981 is designed for holding positioning bone plates. Nevertheless, once a screw retracts with patient motion, an acute vascular injury may result. In addition, removal of the plate is difficult for redoing the procedure and may result in complications such as screw fracture. It therefore requires additional components, such as the member pictured in Figure 120, to cover the screw head in order to maintain this instrument.

According to the problem contained in patent US9536950 that SAM-IDM

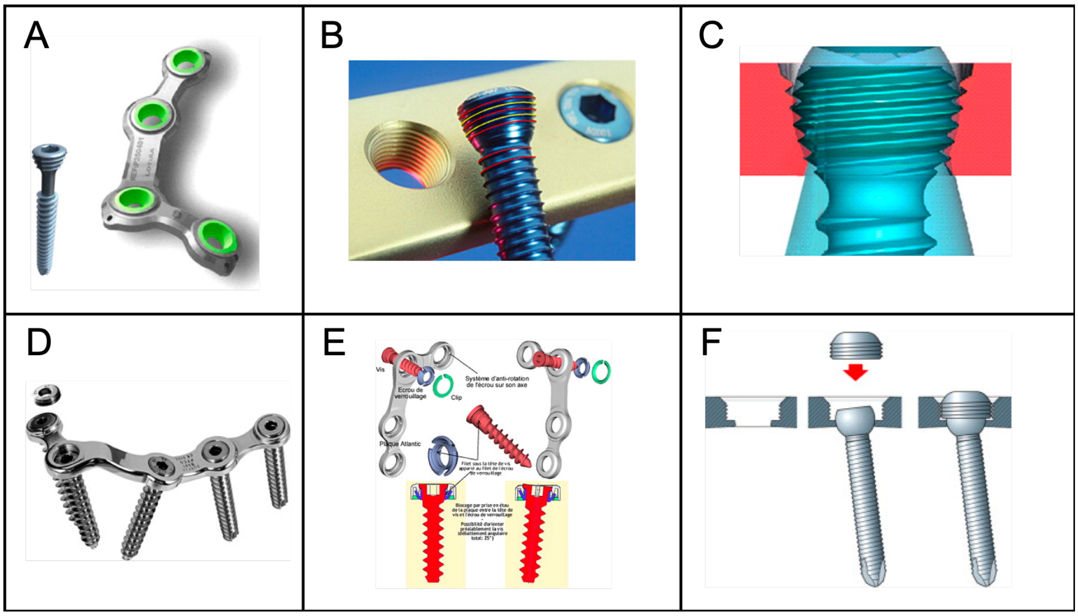


FIGURE 4.14: The illustration of different locking mechanisms for the bone.

extracted, its corresponding solution could be considered as a potential innovative solution for addressing the pulling out issue of patent US9532821.As illustrated in Fig. 4.13, head covers 120 (left) aim to prevent screws from pulling out. We assume that we probably remove these headcovers (red shadow) so that a precious piece of space can be saved in the bone. To prevent the issue of screws' pulling out, we could use a similar design like the strain relaxed buffer layer and the gate-all-around structure (yellow shadow, right) in patent US9536950 to replace headcovers (red shadow, left). In addition, if the material using at the strain relaxed buffer layer could be used as a new material to address the issue of pulling-out issue, the weight of the device could be reduced. It can improve the comfort of the patients. This will provide more imagination to doctors to better cure patients.

Fig. 4.14, (Cronier et al., 2010) introduces several types of locking mechanisms for orthopaedics. These products have been in the market. We notice that the locking mechanism (in patent US9532821) is similar to the type D and type F. In a Surfix system (type D), the locking is obtained by means of a locking nut. The screw has a flat head and is locked in the cavity using a lock nut screwed onto the plate thickness. the Zimmer system (type F) includes a lock nut that covers the spherical head of the screw and can be locked with a clearance of up to 15°. These two kinds of locking nuts are the solution to pulling out screws and their design is similar to the head cover of patent US9532821. Furthermore, the conical and self-tapping screw headlocks (type

FIGURE 4.15: Interlocking plate fixation.

A) is with the selected angulation in a polyaryletherketones (PEEK) insert set in the plate. This product design is similar to the innovative solution of the strain relaxed buffer layer (patent US9536950). The polyaryletherketones part (green part) of type A is almost the same as our innovative solution deriving from SAM-IDM. This esisting product design strongly illustrates the effectiveness of our SAM-IDM model.

Besides, as introduced by (Perren, 2002), the foreign body effect can decrease the resistance to infection. In traditional plate fixation, it will generate necrosis of the cortical bone due to plate compression. On the contrary, the design of locking plate fixation fails to result in osteonecrosis as illustrated in Fig. 4.15. We thus assume that our latent innovative solution is capable of addressing or decreasing the necrosis issue if we can remove the cap 120 in Fig. 4.13 for decreasing the locking mechanism weight.

In conclusion, the performance of SAM-IDM and its promising ability can be illustrated in this detailed case study. Besides, several additional representative cases are listed in Appendix F to illustrate the ability of SAM-IDM further.

4.5 Summary

This chapter introduces two models for mining similar problems from patent documents. IDM-Similar model based on Word2vec neural networks is firstly used to retrieve similar problems contained in patent documents. Nevertheless, we notice that patent sentences are normally longer than generic sentences. Indeed, the expected model needs to be able to learn more context information contained in the sentence. We thus apply LSTMs to build a novel model called SAM-IDM to retrieve similar problems.

With similar problems from different domains patents, there still is another concern about how to automatically match target problems with latent innovative solutions from different domains patents. To address this issue, a problem-solution matching model using XLNet neural networks is proposed in the next chapter.

Chapter 5

Build Links between Problems and Solutions in Patents By IDM-Matching Model

In this chapter, we address how to associate the solution containing the patent to a given problem when we establish similarity between problems in Chapter 4. Indeed, while achieving similar problems from different domains patents, how to automatically match solutions to these similar problems becomes a novel challenge. In order to further improve our previous works, we thus introduce the IDM-Matching model for building links between problems and partial solutions. This work aims to provide concrete innovative solutions to the target problem while TRIZ approaches can only give vague inventive principles.

5.1 Methodology

As illustrated in Fig. 5.1, with the help of Patent Extractor, IDM-Matching firstly retrieves problems of patent documents to prepare a list of related problems. These problems are then converted into queries. After that, a Question Answering system with pre-trained XLNet neural networks(Yang et al., 2019) answers these queries to achieve an answer list. After the filtering mechanism, we extract corresponding solutions. Links between target problems and corresponding solutions are therefore established in patent documents.

Formally, the i-th patent document contains the IDM-related knowledge set $P_i = \{Po_i, Ps_i\}$, which contains several problems Po_i and partial solutions Ps_i. For j-th given problem $Po_{ij} = \{(Po_{ij}^1, z_1); (Po_{ij}^2, z_2); ...(Po_{ij}^{|P_{ij}|}, |z_t|)\}$ in the

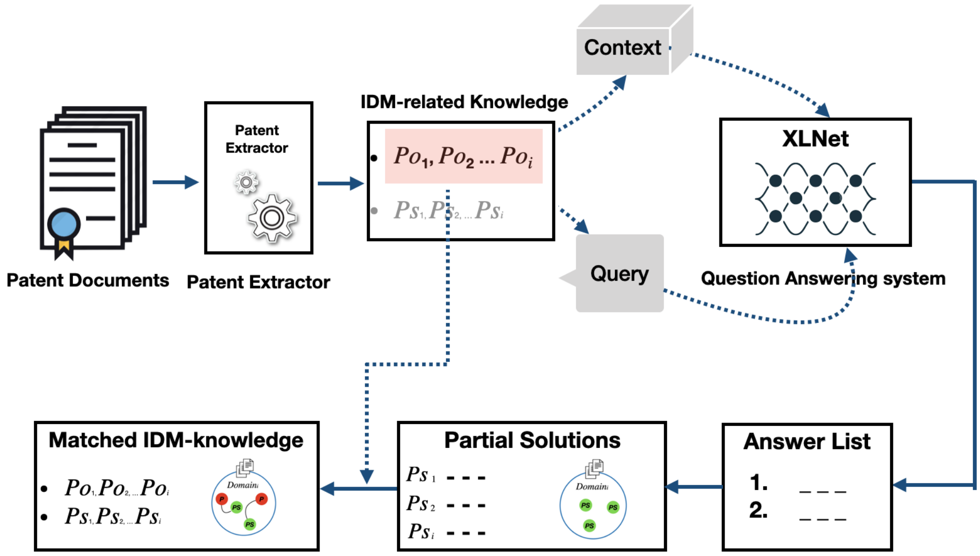


FIGURE 5.1: The framework of IDM-Matching

i-th patent document, $|Po_{ij}|$ is the $|P_{ij}|$-th word that is located in the $|z_t|$-th position. We will detail IDM-Matching in the next.

5.1.1 Pack Problem

We aim at building connections between problems Po and partial solutions Ps by the Question Answering system.

Indeed, Patent Extractor is firstly leveraged to extract problems and partial solutions in patent documents. From previous works, we notice that Patent Extractor usually fails to capture precise partial solutions towards the given problem. Therefore, for the i-th patent document, a problem database without partial solutions is built. We convert each single problem Po_{ij} in i-th patent into a query with a fixed format, for example "*What is the solution for the problem that* ___ ?". With this sort of conversion, problems can be considered as queries in the Question Answering system. In addition, "*What is the solution for the problem that*" is obvious to let the model learn that the target problem is after "*problem that*" and the purpose is finding the corresponding solution towards the given problem. This sort of design can successfully pack the target problem into a query and the model can also be trained which is a problem when some problem sentences are lacking apparent negative tokens. In addition, "*solution*" in the packing head can help the model locate the corresponding solution sentence when the related context text contains the keyword "*solution*".

5.1.2 Define Context Information

In the Question Answering system, context information plays an important role to retrieve the answer for the target query. However, more noisy information also tends to be contained in the longer context information. Redundant context information could also lead to a greater computational cost.

By examining patent documents, we notice that the corresponding partial solution of the target problem always appears near the paragraph containing the target problem, due to the natural structure of the patent document. Besides, several apparent partial solutions are located next to the target problem described by the patent. Therefore, for such a case, we may find the corresponding solution to the target problem in the same paragraph. For these reasons, in this task, we define the context information as three paragraphs of context text, including the same paragraph containing the target problem, the previous single paragraph, and the following single paragraph for the IDM-Matching model.

5.1.3 XLNet Model

A state-of-the-art natural language model called XLNet (Yang et al., 2019) is used in the IDM-Matching model.

As a pre-trained permutation language modelling, XLNet can address the shortcomings of the traditional autoregressive language modelling like GPT (Radford et al., 2019), ELMO (Peters et al., 2018) etc. These models fail to learn both the forward and backward context information to predict the target word. In addition, XLNet is able to address the shortcoming of the artificial symbols like [MASK] used in BERT (Devlin et al., 2018). These symbols appear in the pre-training but are absent in the real text during the fine-tuning process, resulting in a pretrain-finetune discrepancy.

In fact, XLNet extracts the bidirectional context through the permutation according to the dependency rule. Indeed, take the problem sentence of a patent as an example. For the given length ($T = 3$) problem $Po_{ij} = \{(Po_{ij}^1, z_1); (Po_{ij}^2, z_2); (Po_{ij}^3, z_3)\}$ in the i-th patent document, $T!$ different orders are generated to perform a valid autoregressive fatorization for tokens that the problem sentence Po_{ij} contained:

$$p(\mathbf{Po}) = p(Po_{ij}^1)p(Po_{ij}^2|Po_{ij}^1)p(Po_{ij}^3|Po_{ij}^2 Po_{ij}^1) \Rightarrow 1 \rightarrow 2 \rightarrow 3 \tag{5.1}$$

$$p(\mathbf{Po}) = p(Po_{ij}^1)p(Po_{ij}^2|Po_{ij}^1 Po_{ij}^3)p(Po_{ij}^3|Po_{ij}^1) \Rightarrow 1 \rightarrow 3 \rightarrow 2 \tag{5.2}$$

$$p(\mathbf{Po}) = p(Po_{ij}^1|Po_{ij}^2)p(Po_{ij}^2)p(Po_{ij}^3|Po_{ij}^1Po_{ij}^2) \Rightarrow 2 \rightarrow 1 \rightarrow 3 \tag{5.3}$$

$$p(\mathbf{Po}) = p(Po_{ij}^1|Po_{ij}^2Po_{ij}^3)p(Po_{ij}^2)pPo_{ij}^3|Po_{ij}^3) \Rightarrow 2 \rightarrow 3 \rightarrow 1 \tag{5.4}$$

$$p(\mathbf{Po}) = p(Po_{ij}^1|Po_{ij}^3)p(Po_{ij}^2|Po_{ij}^1Po_{ij}^3)p(Po_{ij}^3) \Rightarrow 3 \rightarrow 1 \rightarrow 2 \tag{5.5}$$

where $(Po_{ij}^2|Po_{ij}^1Po_{ij}^3)$ presents the possibility p of the second word Po_{ij}^2 with the first word Po_{ij}^1 and third word Po_{ij}^3. This designed mechanism lets XLNet is able to capture bidirectional context information.

For predicting the target word $Po_{ij}^{|P_{ij}|}$with its position z_t, the function is presented as follows:

$$p_\theta\left(Po_{z_t} = Po_{ij}^{|P_{ij}|}|\mathbf{Po}_{z_{<t}}\right) = \frac{\exp\left(e(Po_{ij}^{|P_{ij}|})^T g_\theta\left(\mathbf{Po}_{z_{<t}}, z_t\right)\right)}{\sum_{x'}\exp\left(e\left(Po_{ij}^{|P_{ij}|'}\right)^T g_\theta\left(\mathbf{Po}_{z_{<t}}, z_t\right)\right)} \tag{5.6}$$

where $\mathbf{Po}_{z_{<t}}$ represents previous words of the target word $Po_{ij}^{|P_{ij}|}$.

Moreover, as an unsupervised language representation learning model, XLNet also benefits from the advantages of Transformer-XL (Dai et al., 2019), so that it presents better performance in language tasks involving long context. Therefore, we combine it into our IDM-Matching model to capture corresponding solutions considering given problems.

5.2 Experiments

The experimental settings are introduced in this section.

Dataset and Evaluation Metric: In this work, an open-source Stanford Question Answering Dataset (SQuAD 2.0)[1] is leveraged to fine-tune and evaluate our model, since it is a significant benchmark in the Question Answering system. As a reading comprehension dataset, SQuAD consists of crowd-sourced questions on a set of Wikipedia articles. Each question is answered by a paragraph or span of text in the corresponding passage. The SQuAD contains 100,000 questions and corresponding labelled answers. In addition, due to the lack of a labelled patent dataset, we choose 50 U.S. patents as the test dataset to verify the performance of IDM-Matching on the real-world

[1] https://rajpurkar.github.io/SQuAD-explorer/

patent dataset manually. These patents were issued on 03, January 2017 via the United States Patent and Trademark Office (USPTO)[2]. EM (Exact Match) and F1 scores are leveraged as evaluation metrics.

Parameter and Computer Settings: In this work, we leverage the SQuAD dataset to tune our IDM-Matching model and the grid search is used to determine the optimal parameters. A pre-trained XLNet-base-case model from Huggingface [3], containing 12-layer, 768-hidden, 12-heads, 110M parameters, is used as our Question Answering system. For tuning the model, we choose epochs among {3, 4, **5**, 6}, learning rate among $\{2e^{-5}, \mathbf{3}e^{-5}, 4e^{-5}\}$, batch size among {**5**, 10, 15}, and others by default. The optimal parameters are highlighted in bold. In addition, fine-tuning the model took 15 hours on 1 Tesla P100 GPU.

Overall Results: IDM-Matching is eventually evaluated on the evaluation dataset. It achieves 70.99% EM and 72% F1.

5.3 Case Study

We illustrate a real-world use case on the U.S. patent document in this section, to illustrate the practical performance of our IDM-Matching.

"**Electrically conductive touch pen**"(US8847930) is a U.S. patent in the field of physics. As illustrated in Fig. 5.2, it proposes a multi-function writing device that can be used for the physical marking on the conventional writing surface, for the digital marking on the computer digital display, or as other input means in combination with a computer digital display. The invention has an internal cartridge that can be expanded through a hole in the stylus tip. The stylus tip extends out of a sleeve formed of a conductive elastic material. The sleeve extends upward along the rigid axis of the device so that it contacts sufficient ground. The stylus tip is coated with a protective material that adjusts the coefficient of friction and prevents carbon build-up on the touch screen. A sufficient contact patch is achieved to simulate a human finger, thus overcoming the false positives associated with common touch screen logic.

As illustrated in Fig. 5.3, Patent Extractor retrieves several problems (red circles) from the patent. We choose seven correct problems (red circles with yellow borders) as inputs for IDM-Matching and convert them into 7 queries. We list questions, answers that IDM-Matching retrieved, correct answers,

[2]https://bulkdata.uspto.gov/data/patent/grant/redbook/fulltext/2017/
[3]https://huggingface.co/transformers

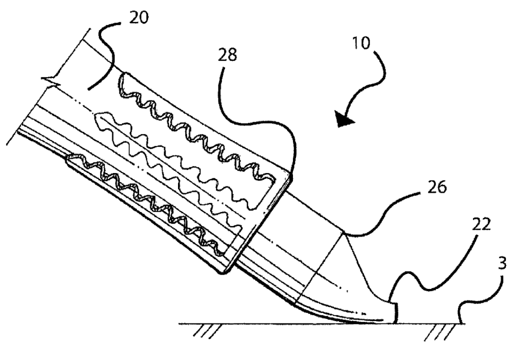


FIGURE 5.2: An overview of the invention

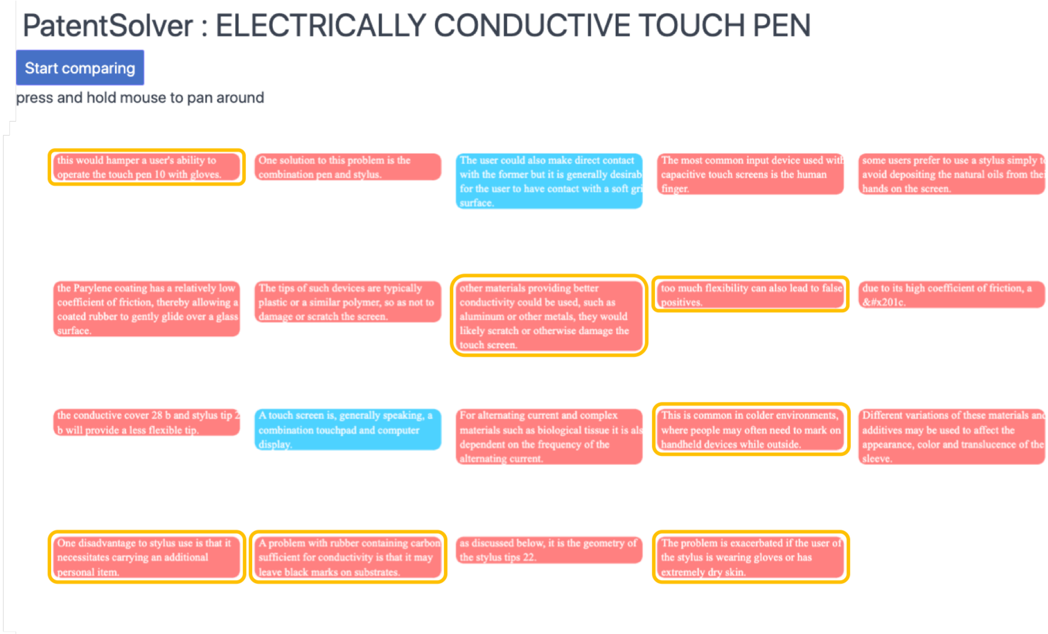


FIGURE 5.3: The extracted problems

and related context information are as follows:

- 1. **[Question]**: What is the solution for the problem that this would hamper a user's ability to operate the touch pen 10 with gloves?
  | **Answer List** |
  1). ***The inner moulding 29 is replaced by a former 39 that is ideally metallic. This alternative embodiment is designed to address the aforementioned problems attendant to a user wearing gloves***, having very dry skin, or situations in which the user does not make good conductive contact with the touch pen 10. In such cases, the conductive cover 28 needs to be in good electrical contact with a volume of metal V (m3) of conductivity.

2). conductive cover 28 needs to be in good electrical contact with a volume of metal V (m3) of conductivity a (Siemens per meter S/m).
Analysis: From the first answer (in boldface), we know that the ideally metallic *former 39* can be used to address the issue of disabling using the touch pen with gloves.

- 2. [**Question**]: What is the solution for the problem that too much flexibility can also lead to false positives?
  | **Answer List** |
  1).The larger the air cavity 32, the more flexible the stylus tip 22 will become. However, too much flexibility can also lead to false positives. As shown, the former 39 comprises an extension 41 of various sizes.
  2). The larger the air cavity 32, the more flexible the stylus tip 22 will become. However, too much flexibility can also lead to false positives. As shown, the former 39 comprises an extension 41 of various sizes. ***The size of this extension directly controls the size of the air cavity 32. In some embodiments, this extension may be a controllable feature of the touch pen 10.***
  Analysis: Answer 2 mentioned that the size of the extension might be the reason for the target issue.

- 3. [**Question**]: What is the solution for the problem that one disadvantage to stylus use is that it necessitates carrying an additional personal item?
  | **Answer List** |
  1). ***combination pen and stylus***.
  2). ***combination pen and stylus***. A touch screen.
  3). ***the combination pen and stylus***. A touch screen is, generally speaking, a combination of a touch pad and computer display.
  Context:
  One solution to this problem is the combination pen and stylus.
  Analysis: First three answers all mentioned *combination pen and stylus*. From the context information, we can know it is correct and direct.

- 4. [**Question**]: What is the solution for the problem that a problem with rubber containing carbon sufficient for conductivity is that it may leave black marks on substrates to which it comes into contact?
  | **Answer List** |
  1).***coating the rubber, or selectively the rubber tip, with a very thin layer of Parylene***.

2). ***selectively the rubber tip, with a very thin layer of Parylene***. This conformal coating.

3). very thin layer.

Context:

These problems can be solved by coating the rubber, or selectively the rubber tip, with a very thin layer of Parylene.

Analysis: From the context information, the first two answers all mentioned key solutions for the given problem.

- 5. [**Question**]: What is the solution for the problem that the problem is exacerbated if the user of the stylus is wearing gloves or has extremely dry skin?

  | **Answer List** |

  1). ***films made from graphene (carbon nanotubes), or other suitable material***. Conductive materials that touch or are in very close proximity to this type of touch screen alter the electrostatic field of the screen, thereby creating a registerable change in capacitance.

  Context:

  Traditional plastic or polymer-based styli are not effective in marking on capacitive touch screens because they are not sufficiently conductive. The problem is exacerbated if the user of the stylus is wearing gloves or has extremely dry skin.

  Correct Answer:

  Capacitive touch screens generally comprise a flat insulative transparent sheet such as glass having an inside portion coated with a transparent conductor such as indium tin oxide (ITO), ***films made from graphene (carbon nanotubes), or other suitable material***.

  Analysis: Question 5 mentioned the issue that the stylus cannot be used well with the fingers wearing gloves or dry skin. In fact, the plastic or polymer-based caused the problem, but this significant information did not appear in the question. However, IDM-Matching still managed to learn the relevant information and achieve the right answer for films made of graphene (carbon nanotubes), or other suitable materials.

- 6. [**Question**]: What is the solution for the problem that this is common in colder environments, where people may often need to mark on handheld devices while outside?

  | **Answer List** |

  1). Capacitive touch screens are quickly replacing resistive touch screens.

2). sink or source of electrons.

3). films made from graphene (***carbon nanotubes***), or other suitable material. Conductive materials that touch or are in very close proximity to this type of touch screen alter the electrostatic field of the screen, thereby creating a registerable change in capacitance.

4). sink or source of electrons sometimes called a "ground.

Correct Answer:
One solution that enables a stylus to be used with a capacitive touch screen is the use of conductive rubber or a similar conductive elastomeric material.

Context:
One solution that enables a stylus to be used with a capacitive touch screen is the use of ***conductive rubber*** or a similar conductive elastomeric material. ***Conductive rubber*** is a rarer and more expensive form of rubber that ***contains*** suspended graphite carbon, ***carbon nanotubes***, nickel or silver particles.

Analysis: Question 6 mentions an issue where people need to mark their handheld devices when they are out in the cold. This situation can lead to the insulation of the stylus from the human body. The list of answers presented in IDM-Matching does not indicate a precise answer. The correct answer is to use conductive rubber or a similar conductive elastic material. However, we see that carbon nanotubes appear in our list of answers. Indeed, conductive rubber contains carbon nanotubes. This hints that IDM-Matching still learns the important information to make a connection between the problem and the corresponding solution.

- 7. [**Question**]: What is the solution for the problem that other materials providing better conductivity could be used, such as aluminum or other metals, they would likely scratch or otherwise damage the touch screen?

  | **Answer List** |

  1). ***Conductive materials*** that touch or are in very close proximity to this type of touch screen.

  2). films made from graphene (carbon nanotubes), or other suitable material. ***Conductive materials*** that touch or are in very close proximity.

  3). films made from graphene.

4). ions—cations.

5). conductive materials such as biological tissue, these charged carriers could be predominantly ions—cations and/or anions.

Correct Answer:

One solution that enables a stylus to be used with a capacitive touch screen is the use of ***conductive rubber or a similar conductive elastomeric material***.

Analysis: As shown in the correct answer, the first two answers all mentioned the key information to address the given problem.

In summary, from this detailed case study, we note that IDM-Matching can retrieve precious corresponding solutions towards given problems. It can facilitate engineers facing a large number of patent documents by automating the connection established between the problem and the corresponding partial solution.

5.4 Summary

In this chapter, we propose a problem-solution matching model called IDM-Matching for building links between problems and corresponding solutions in patent documents. IDM-Matching is capable of automating the solution retrieval and matching it with the corresponding problem in patents. With the help of IDM-Matching, it can facilitate engineers to find innovative details contained in patent documents to accelerate R&D activities. In addition, this model can further improve the innovative solutions retrieval for the given problems through associating with different domain similar problems in patent documents in Chapter 4. More importantly, with this work, engineers without extensive knowledge of different domains are capable of fully leveraging the inventive knowledge from various patent documents to facilitate their innovative deign inspirations. Final experimental results on the real-world patent dataset illustrate the performance of IDM-Matching. In particular, a detailed case study presents the usage of IDM-Matching in reality.

Nevertheless, we still notice that there is an issue that IDM-Matching might provide numerous latent innovative solutions towards the target problem when the number of input patents is numerous. It generates an obstacle for engineers to choose the best ones. To address this issue, we propose a way in the next chapter to rank latent innovative solutions by using the multiple criteria decision analysis approach.

Chapter 6

Inventive Solutions Ranking by PatRIS Model

For the innovative solution mining, IDM-Similar, SAM-IDM, and IDM-Matching models have been introduced in Chapter 4 and Chapter 5 to prepare similar problems and build links between problems and corresponding solutions from different domain patents.

Unfortunately, a missing step remains which is addressing the issue of ranking several potential innovative solutions when the large size of input patents, especially several solutions with the same similarity value. Thus, in this chapter, we propose an inventive solutions ranking model named PatRIS by mainly using the multiple criteria decision analysis approach to rank latent inventive solutions according to their potential inventiveness.

6.1 Patent Inventiveness Evaluation Approaches

In recent research works, there have been few methods to assess the inventiveness of solutions. However, several approaches have been proposed to assess the value or inventiveness of patents.

Abrams et al. (Abrams, Akcigit, and Grennan, 2013) presented that the relationship between patent value and citations forms an inverted-U. Moreover, citations accumulate two types of patent innovation efforts: strategic and productive. In particular, productive innovation implies that the innovator makes an early contribution to the field and then earns a large profit. This leads to a positive relationship between patent value and forward citations. In addition, patent citations are considered to reflect the technical and economic importance of the innovation (Block et al., 2013). Besides, several research works on patent inventiveness based on specific patent fields have been proposed. For example, innovation patterns in energy technologies are described by Huenteler et al. (Huenteler et al., 2016). They analyzed patent citation networks for wind power and solar PV. Park et al. (Park, Yoon, and Kim, 2013) proposed, relying on the empirical data, a framework that relies on the patent index to support corporate mergers and acquisitions. Patent scope, grant lag, patent family size, different indexes, and citations are used by Squicciarini et al. (Squicciarini, Dernis, and Criscuolo, 2013) to capture the economic and technical value of patents. It helps to define and measure the quality of patents. They also note that different indicators such as grant lags and forward citations may contain different effects on patents in different countries. In addition, various patent features including renewal, grant decision, and opposition are explored by several works (Van Zeebroeck, 2011; Nagaoka, Motohashi, and Goto, 2010; Guan and Gao, 2009).

We notice that these research efforts have been mainly focusing on assessing the specific domain's patent inventiveness or patent value (Abrams, Akcigit, and Grennan, 2013; Block et al., 2013; Park, Yoon, and Kim, 2013; Squicciarini, Dernis, and Criscuolo, 2013; Van Zeebroeck, 2011; Nagaoka, Motohashi, and Goto, 2010; Guan and Gao, 2009). Nevertheless, to the best of our knowledge, no other works are aiming to assess the inventiveness of solutions in patents. Moreover, no such work is capable of leveraging different domain patents to extract innovative solutions to the given problem. Indeed, those works on patent indicators have inspired us to build our approach to rank latent innovative solutions.

6.2 Typical Patent Inventiveness Indicators

Our work begins with a collection of several patent inventiveness indicators to build our inventive solution ranking methodology, as illustrated in Table 6.1. *The earliest priority date* (first application of the patent in the world) is recommended as one of the reflections of patent inventiveness (Dernis and Khan, 2004; Silverberg and Verspagen, 2007). *Patent citation* is another important indicator for patent innovation. We can track the flow of knowledge among different patents by this indicator. It can be classified as two sorts of citations, *cited-forward citations* and *cited-backward citations*. *Cited-forward citations* are citations received subsequently by patents. It is able to leverage it to measure the technological impact of inventions (Petruzzelli, Rotolo, and Albino, 2015; Miller, Fern, and Cardinal, 2007). A *cited-backward citation* is a patent referenced during the patent application process. It can be used to track the creative knowledge spillovers in the technology (Noailly and

TABLE 6.1: Patent Indicators and Explanations

Patent Indicator	Explanation
Number of Inventors (NI)	The number of inventors involved in the patent.
Cited-Forward Citations with no Family (CFCNF)	Forward Citations that are not family-to-family cites.
Cited-Forward Citations with Family (CFCF)	Forward Citations that are family-to-family cites.
Cited-Backward Citations with no Family (CBCNF)	Backward Citations that are not family-to-family cites.
Cited-Backward Citations with Family (CBCF)	Backward Citations that are family-to-family cites.
Number of IPC Classes (NIPCC)	The number of technical classes.
Priority Date (PD)	The earliest filing date in a family of patent applications.
Family Size (FS)	The number of countries in which the same invention is patented.

Shestalova, 2017). In particular, the value of the patent is considered as correlated with the number and quality of its forward citations (Khanna, Guler, and Nerkar, 2016). Besides, patents that are cited more times than average are more likely to be renewed (Hall, Jaffe, and Trajtenberg, 2005). Moreover, when a patent in the patent family (a set of patents performed in various countries to protect a single invention) cites a patent rather than the patent itself, this is seen as a family citation since the family cites other patents. Therefore, we classify citations as in-family or non-in-family. *Number of IPC classes* is the number of technical classes covered by the patent. (Lerner, 1994) introduced its positive correlation with the company's market value. However, there is limited evidence on the correlation between the inventiveness of patents and *the number of IPC classes* (Lanjouw and Schankerman, 2004; Harhoff, Scherer, and Vopel, 2003). *Number of inventors* may indicate the research cost behind the invention. Moreover, (Guellec and La Potterie, 2007; Arora, Fosfuri, and Gambardella, 2004) think that there is a correlation between the number of inventors listed in the patent and the economical and technological value of patents. In addition, the more resources involved, the more research-intensive and expensive the project will be in terms of research. *The size of patent families* is proxy by the number of patent offices in which a given invention is protected (Squicciarini, Dernis, and Criscuolo, 2013). Because international patent applications are highly expensive than domestic applications. It implies that they contain higher expectations of the return on their patents (Nagaoka, Motohashi, and Goto, 2010).

In this work, *the number of IPC classes* is not chosen as a patent inventiveness indicator since several opposing research opinions. In addition, for *family size*, different patent definitions may have an impact on the 25% patent families with complex structures and lead to different family compositions, which have an impact on family size as the proxy of patent value (Martínez, 2011). In addition, *priority date* of the patent is not suitable for this work.

In this work, we aim to rank different solutions according to their inventiveness. The priority dates of the corresponding patents are not in the same time level of comparison. In addition, solutions of the patents with earlier priority dates do not present more inventiveness than others. *Family size* and *priority date* are therefore not chosen in this work.

In general, five patent indicators, *cited-backward citations with family, cited-backward citations with no family, cited-forward citations with family, cited-forward citations with no family*, and *the number of inventors* are leveraged to be indicators in this work.

6.3 Patent Ranking Inventive Solutions: PatRIS

As illustrated in Fig. 6.1, an inventive solutions ranking model called PatRIS is proposed. In detail, different domain patents P_{at} firstly flow to the SAM-IDM model based on LSTM neural networks to achieve similar pairwise problems set $\mathcal{P}$ for the target problem P_{target} and corresponding similarity values (SV). After that, IDM-Matching based on XLNet neural networks retrieves solutions $\mathcal{S}$ for given similar problems according to the context information of patent documents. We see these solutions from different domain patents as potential inventive solutions for the target problem.

However, when a large number of patents are input, several inventive solutions with the same similarity value might be also generated. It leads to the obstacle of ranking them only according to corresponding similarity values. As shown in Table 6.2, three latent inventive solutions with the same similarity value of 0.86 are generated from the real-world U.S. patent sample. The hypothesis that inventiveness of potential solutions is correlated to the inventiveness of corresponding patents provides a resolution to address the concern. Patent inventiveness indicators in Section 6.2 and the similarity value indicator SV are leveraged by PatRIS to build the inventive solution ranking model according to the multiple-criteria decision analysis approach. It can eventually rank these inventive solutions according to patent features and semantic similarity.

Moreover, as a sub-discipline of the operation research, Multiple Criteria Decision Analysis (MCDA) can explicitly evaluate multiple criteria in decision making to help understand the inherent trade-off (Greene et al., 2011). PatRIS based on the MCDA approach named TOPSIS contains the first five patent indicators in Table 6.1 and semantic similarity value (SV) as the sixth

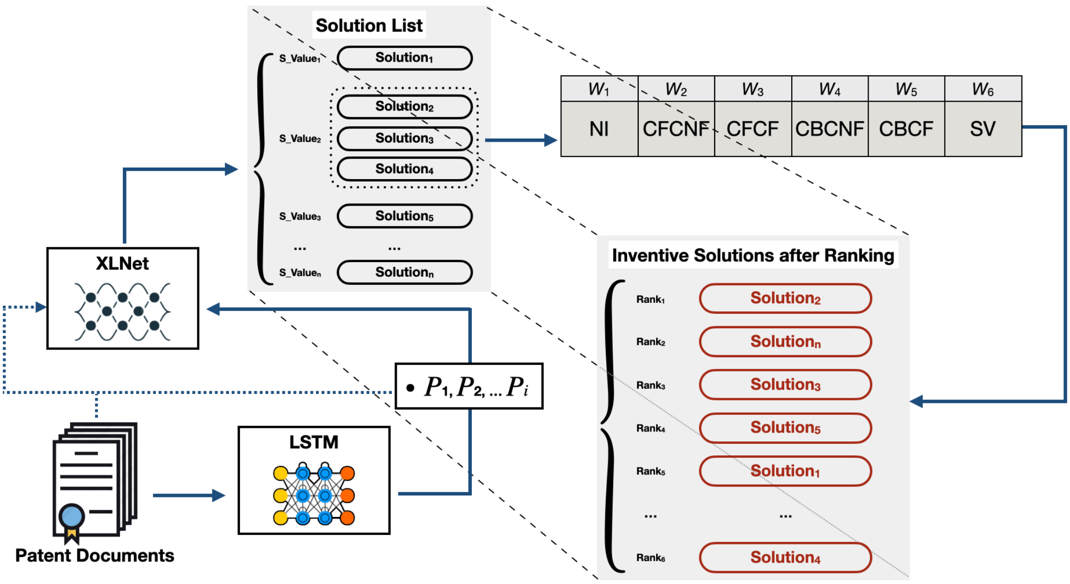


FIGURE 6.1: The framework of PatRIS

TABLE 6.2: Inventive Solutions from Different Domains

Target Problem	Similar Problems	Patent Number	Similarity Value	Domain	Inventive Solutions
Patent Number: US9534284 Domain: C the second metal layer is not provided	the web page is not captured normally	US9535571	0.86	G	associated with a broadcasting application according to an exemplary embodiment of the present invention
	if the wfe instruction is not intended for agent discovery purposes	US9535772	0.83	G	are accessible to the agent as well as the client, such as designated locations in a memory 104
	the scope of the first aspect is not limited to these examples	US9537403	0.86	H	a slow DAC 930, the gear shift can be made gradual. Alternatively, by using a fast DAC, the gear shift can
	the message is not received in step	US9537998	0.86	H	step 301. Alternatively, when the message is transmitted, the mobile

criterion to build a ranking system. It aims to rank mined solutions according to their latent inventiveness through combining patent indicators and semantic similarity. We assume that the solution is more inventive when the corresponding patent is ranked as one of the best creative supports and the corresponding problem contains high similarity with the target problem. It can be formulated as the function 6.1. It is designed to maximize the inventiveness of solutions under the selected indicators to rank them.

$$\begin{aligned} max f(x) &= (F_1(x), F_2(x), F_3(x), F_4(x), F_5(x), F_6(x)) \\ F_1(x) &= NI(x_1, x_2, ..., x_n) \\ F_2(x) &= CFCNF(x_1, x_2, ..., x_n) \\ F_3(x) &= CFCF(x_1, x_2, ..., x_n) \\ F_4(x) &= CBCNF(x_1, x_2, ..., x_n) \\ F_5(x) &= CBCF(x_1, x_2, ..., x_n) \\ F_6(x) &= SV(x_{sv1}, x_{sv2}, ..., x_{svn}) \end{aligned} \tag{6.1}$$

subjected to linear constraints:

$$x_i \geq 0, i = 1, ..., n \tag{6.2}$$

$$x_{sv_i} \in [0, 1], i = 1, ..., n \tag{6.3}$$

In (6.1)-(6.3), $NI(x)$, $CFCNF(x)$, $CFCF(x)$, $CBCNF(x)$, $CBCF(x)$, and $SV(x_{sv})$ are a count of the number of inventors, cited-forward citations with no family, cited-forward citations with family, cited-backward citations with no family, cited-backward citations with family, and semantic similarity respectively. x stands for the vector of the count variable, 0 presents the lower bound of the i-th count variable. The linear constraints come from the consideration of patent cost expenditure (number of inventors), peer recognition (number of backward citations), relevant knowledge references (number of forward citations), and semantic distance (similarity value). As illustrated in Table 6.3, a real-world sample of U.S. patents, the count distribution of indicators has been listed. The higher the number of inventors, the higher the cost invested in the patent. A high number of forward citations present in the invention contain a high technical impact. A high number of backward citations presents that the innovation tends to cite a large number of and range of scientific publications. A high similarity value indicates a short

TABLE 6.3: A sample of the indicator detail

Patent Number	NI	CFCNF	CFCF	CBCNF	CBCF	SV
US9535571	1	1	22	18	0	0.86
US9537403	3	3	14	15	0	0.86
US9535772	2	0	2	4	0	0.83
US9537998	2	0	2	9	0	0.86

distance between the corresponding problem of solution and the target problem. These six criteria are therefore positively related with the corresponding values x_{ij}. PatRIS ranks the mined solutions from the most innovative to the least innovative. Different weights w_j are also assigned to these criteria.

As illustrated in formula 6.4, we normalize each value so that all attributes are set in the same range. j-th feature $F_j = \{x_1, x_2, ...x_i\}$, $i \in \{0, n\}$ derives from six criteria, and x_{ij} is the value of the i-th solution under the j-th feature. After normalization, values x_{ij} of patent indicators reaching 1 indicate that the corresponding patents are more innovative in the j-th feature F_j than other patents. When $x_{sv_{ij}}$ is closer to 1, the solution might be more innovative. Attribute weights are also applied to the corresponding values.

$$Normalization(x_{ij}, F) = \frac{x_{ij}}{sum(F)} \tag{6.4}$$

TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) with other two typical types of MCDA approaches, namely Weighted Sum approach and Weighted Product approach, are listed below.

1. Weighted Sum (WS):

$$Score_i = \sum_i x_{ij} \times w_j \tag{6.5}$$

2. Weighted Product (WP):

$$Score_i = \Pi_i {x_{ij}}^{w_j} \tag{6.6}$$

3. TOPSIS: It aims at selecting the alternative that contains the longest geometric distance from the negative ideal solution and the shortest geometric distance from the positive ideal solution. Thus, the optimal goal is to approach the best alternative and stay away from the worst alternative. The main steps are as follows:

Step1: Realize the normalized matrix.

$$\bar{x_{ij}} = \frac{x_{ij}}{\sqrt{\sum_i {x_{ij}}^2}} \tag{6.7}$$

Step2: Assign weights to each value in order to compute the weighted normalized matrix.

$$xw_{ij} = \bar{x}_{ij} \times w_j \tag{6.8}$$

Step3: Compute the ideal best to mark the attribute as containing the positive impact.

$$idealbest_{positive_j} = Max(xw_{ij}) \tag{6.9}$$

$$idealworst_{positive_j} = Min(xw_{ij}) \tag{6.10}$$

Step4: Compute the Euclidean distance of $idealbest_{positive}$ and $idealworst_{positive}$, respectively.

$$Best_i = \sqrt{\sum_j (xw_{ij} - idealbest_{positive_j})^2} \tag{6.11}$$

$$Worst_i = \sqrt{\sum_j (xw_{ij} - idealworst_{positive_j})^2} \tag{6.12}$$

Step5: Compute $Score_i$ and use it to rank inventive solutions.

$$Score_i = Worst_i / (Best_i + Worst_i) \tag{6.13}$$

In general, as shown in Table 6.4, the final ranking of inventive solutions according to three types of MCDA methods are presented. Besides, algorithm 1 presents the generic idea of PatRIS according to TOPSIS towards ranking inventive solutions.

6.4 Experimental Settings

As we introduced in Chapter 4, 6,161 U.S. patent documents are used as the input of SAM-IDM based on LSTM neural networks to generate the dataset of problem sentences for each patent. It eventually generates 327 pairs of similar problems from 8 different domain patents. After that, IDM-Matching based on XLNet neural networks (In Chapter 5) is used to prepare potential

Algorithm 1 PatRIS

Input: P_{target}: target problem; P_{at}: input patents; F: patent feature; th: similarity threshold;

Output: ranking

$\mathcal{P} \leftarrow \text{LSTM}(P_{at}, P_{target}, th)$

$\triangleright$Extract input Patents P_{at} to achieve similar problems $\mathcal{P} = \{P_1, P_2, ..., P_n\}$ with chosen similarity threshold th for the target problem P_{target}

$\mathcal{S} \leftarrow \text{XLNet}(\,P_{at}, \mathcal{P})$

$\triangleright$XLNet retrieves corresponding solutions $\mathcal{S} = \{S_1, S_2, ..., S_n\}$ of $\mathcal{P}$ for P_{target} from P_{at}

for each $S_i \in \mathcal{S}$ **do**

$\quad \bar{x_{ij}} \leftarrow \frac{x_{ij}}{\sqrt{\sum_i {x_{ij}}^2}}$ normalization of each value x_{ij} under F_j

$\quad xw_{ij} \leftarrow \bar{x}_{ij} \times w_j$ assign weights w_j to achieve weighted normalized matrix

$\quad idealbest_{positive_j} \leftarrow Max(xw_{ij})$

$\quad idealworst_{positive_j} \leftarrow Min(xw_{ij})$

$\quad Best_i \leftarrow \sqrt{\sum_j (xw_{ij} - idealbest_{positive_j})^2}$

$\quad Worst_i \leftarrow \sqrt{\sum_j (xw_{ij} - idealworst_{positive_j})^2}$

$\quad Score_i \leftarrow Worst_i / (Best_i + Worst_i)$ achieve ranking $Score_i$ for S_i

end for

return ranking($Score_i$, S_i)

TABLE 6.4: Ranking of Different Decision Approaches

Patent Number	Inventive Solutions	NI	CFCNF	CFCF	CBCNF	CBCF	SV	Rank(WS)	Rank(WP)	Rank(PatRIS)
US9535571	associated with a broadcasting application according to an exemplary embodiment of the present invention	1	1	22	18	0	0.86	3	2	3
US9537403	a slow DAC 930, the gear shift can be made gradual. Alternatively, by using a fast DAC, the gear shift can	3	3	14	15	0	0.86	4	1	4
US9535772	are accessible to the agent as well as the client, such as designated locations in a memory 104	2	0	2	4	0	0.83	2	4	2
US9537998	step 301. Alternatively, when the message is transmitted, the mobile	2	0	2	9	0	0.86	1	3	1

innovative solutions for them. In order to rank these latent innovative solutions, we assign different weights to the criteria of PatRIS to increase the weight of the most valuable features, such as SV and CFCNF. The optimized weights w_* are selected as {NI: 0.1, CFCNF: 0.3, CFCF: 0.1, CBCNF: 0.1, CBCF: 0.1, SV: 0.3} through several comparisons with the different weights distribution. PatRIS ultimately ranks these potential innovative solutions according to their inventiveness to let engineers realize the most likely potential innovative solutions.

6.5 Experimental Analysis

The target problem in Table 6.2 is chosen as an example. The full target problem sentence is "*When the second metal layer 410 is not provided, the first metal layer 310 may be dissolved in a solvent to thus be removed.*". Indeed, patent US9534284 mentions "*A metal having corrosion resistance, such as stainless steel, may have an oxide film on the surface thereof to protect the metal.*". The solution that the patent proposed for the target problem is thus to replace layer 310 in Fig. 6.2 by using stainless steel or other metals containing corrosion resistance.

As shown in Table 6.4, PatRIS and WeigtedSum illustrate the same ranking. Indeed, Widianta et al. (Widianta et al., 2018) mentioned that the TOPSIS approach tends to outperform other MCDA approaches. In fact, in this case, the ranking results according to PatRIS based on TOPSIS illustrate the same performance. In detail, the first inventive solution "*Alternatively, when the message is transmitted, the mobile terminal determines whether a received or transmitted message for a call of the message exists in step 343.*" from U.S. patent US9537998 introduced a mobile terminal device. Its role is to determine whether the message is received. Therefore, to address the target issue, designing a device like the mobile terminal in patent US9537998 is probably able to detect solvents to prevent the dissolution issue or to detect the condition of the oxide film of metals for further protection. It could be a potential innovative solution for the target problem. The second inventive solution from U.S. patent US9535772 "*In alternative embodiments, the client may communicate intentions to the agent via any number of WFE communication registers 108 and/or any number of other storage components that are accessible to the agent as well as the client, such as designated locations in a memory 104.*" is designed to leverage a communication register or storage components to solve the problem of accessing the agent. For our target problem, the inventive solution of the communication register is similar to the first inventive solution. We can design an electrical device to detect the condition of the oxide film and thus prevent the dissolution issue. In fact, the first two latent inventive solutions belong to the same sort of solution. PatRIS does rank them as the neighbouring solutions. The third inventive solution "*FIG. 10 is a diagram illustrating the controlling display of a terminal icon associated with a broadcasting application according to an exemplary embodiment of the present invention.*" from U.S. patent US9535571 introduces a controlling display of the terminal icon, as shown in Fig. 6.3. Further details are shown in Fig. 6.3, (a) part is the terminal displaying screen. The updated message can be illustrated in the screen from (b) part when the graphical object changes condition. We, therefore, hypothesize that the latent inventive solution could be adding a screen device to detect the condition of layer 310 and then perform further protective measures. The fourth inventive solution in U.S. patent US9537403 "*As mentioned above for FIG. 9, by using a slow DAC 930, the gear shift can be made gradual.*" introduces a slow Digital-to-analogueue Converter (DAC) device, as illustrated in Fig. 6.3. With details of the patent, we notice that it makes the shifting progressive by converting the output of the counter to an analogueue signal, forming a compensating signal. We assume therefore that designing a device like DAC device could be used to detect the solvent or layer 310 through sending an analogue signal or to slow down the dissolution of the metal layer in the solvent, and then leave the time for further protective measures.

In general, the mobile terminal, the alternate embodiment, and the controlling display introduced in the first three inventive solutions are directly derived from the initial solution sentences. They could provide the direction of inventive solutions. The slow DAC device in the fourth inventive solution converts the output to an analogue signal. In fact, the analogue signal is not directly linked to the oxide film-related solutions. Nevertheless, it still can provide an innovative problem-solving idea using a device to alert or prevent its related problems. Therefore, the fourth inventive solution is seen as less of a direct reminder to the target problem than the first three inventive solutions. This also coincides with the ranking results of PatRIS according to TOPSIS. It implies that the multiple criteria decision analysis method is suitable for ranking potential inventive solutions based on the combination of corresponding patent inventiveness indicators and the similarity indicator.

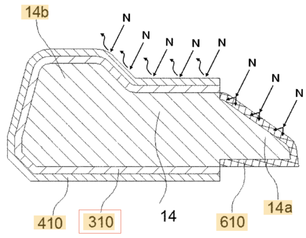
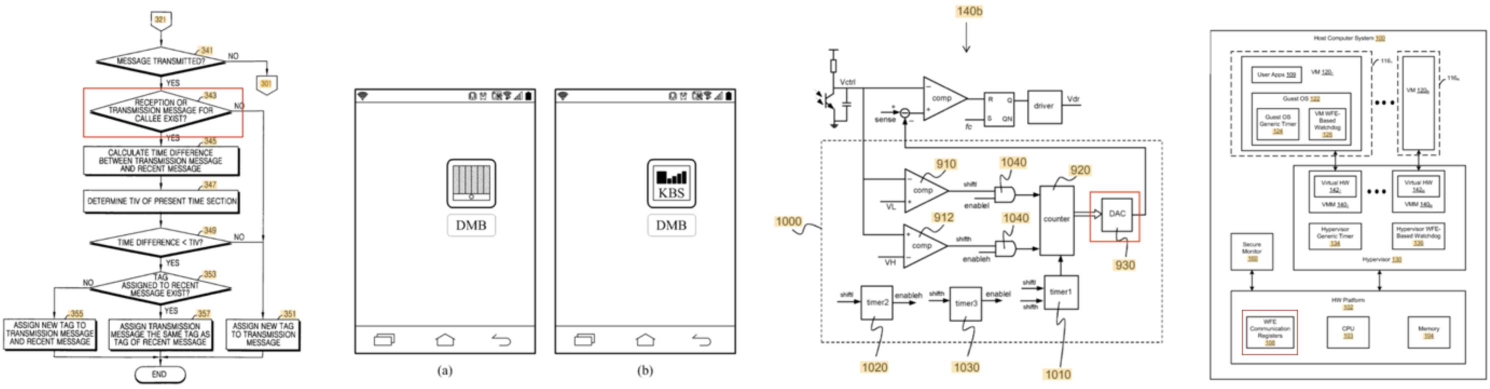


FIGURE 6.2: The illustration figure of patent US9534284



US9537998 US9535771 US9537403 US9535772

FIGURE 6.3: The illustration figures of patents

6.6 Summary

In this chapter, a model named PatRIS based on a multiple criteria decision analysis approach called TOPSIS is proposed. It is mainly used to rank latent inventive solutions from the IDM-Matching model in Chapter 5.

We assume that solutions' inventiveness is associated with their corresponding patents and the similarity level of corresponding problems for the target problem. PatRIS is eventually able to rank these solutions via several chosen indicators. This work further facilitates engineers to find the most likely potential inventive solutions hidden in patent documents of different domains for the target problem to accelerate R&D activities. More importantly, it avoids the failure of ranking when solutions with the same similarity value. In particular, the final case analysis on real-world U.S. patents illustrates the performance of PatRIS and presents its potential perspective in the real world.

Chapter 7

The Software Prototype: PatentSolver

From Chapter 4 to Chapter 6, we have introduced sequential steps to extract the most possible potential inventive solutions from different domain patents. In Chapter 4, two similar problem retrieval models named IDM-Similar and SAM-IDM, which are based on Word2vec neural networks and LSTM neural networks, are developed. In Chapter 5, according to XLNet neural networks, a novel problem-solution matching model named IDM-Matching is proposed to build connections between problems and solutions in patent documents. In Chapter 6, an inventive solutions ranking model named PatRIS based on the multiple criteria decision analysis approach is proposed to eventually rank latent inventive solutions.

To realize and automate the aforementioned works, we have developed a software prototype named PatentSovler for providing potential inventive solutions from a large size of real-world U.S. patent documents. This software prototype uses both data crawling technologies and natural language processing approaches to provide the given patent statistical details, analyze problem sentences, provide similar problems and corresponding solutions, and ranking of inventive solutions. In this chapter, we first introduce the development environment and tools for the software prototype. We then introduce its core functionalities by the following sequence: similar problem retrieval, inventive solutions matching, inventive solutions ranking, and patent statistical details collecting.

7.1 Software Development Environment and Tools

During the development of PatentSolver, several software and tools were used. They are introduced as follows:

- **Python (Python 3.8)**[1] Python is an interpreted high-level general-purpose programming language. It is designed with an emphasis on the readability of codes, using significant indentation. Its language constructs an object-oriented approach that aims to help programmers write clear, logical codes for small and large scale projects. Besides, it supports multiple programming paradigms, including structured (particularly, procedural), object-oriented, and functional programming. Its comprehensive standard libraries and various third-party libraries make the work of the thesis easier to achieve than using other programming languages.

- **PyCharm**[2] It is an integrated development environment (IDE) for computer programming. With PyCharm, in this work, we can access the command line, connect to the database, create a virtual environment, and manage the version control system all in one place. We especially develop our models and applications via this IDE.

- **Streamlit**[3] Streamlit is an open-source app framework for Machine Learning and Data Science teams. It can be used to build an application conveniently to share our developed models and analysis. It also performs well with several tools related to data science, such as PyTorch, Keras, and Pandas. In this thesis, we use it to create our software prototype.

- **Docker**[4] Docker is the source code for the Core Docker project. It is an infrastructure management platform that is required for running as well as the deployment of software. The main motive behind using Docker in the development is that containers of an OS system level are used as an abstraction layer on top of the application and deployment operations.

In the next sections, we introduce key functions and main Graphical User Interfaces (GUI) of PatentSolver.

7.2 Similar Problems Retrieval GUI

We introduce the function of similar problems retrieval in this section.

[1] https://www.python.org/
[2] https://www.jetbrains.com/pycharm/
[3] https://streamlit.io/
[4] https://www.docker.com/

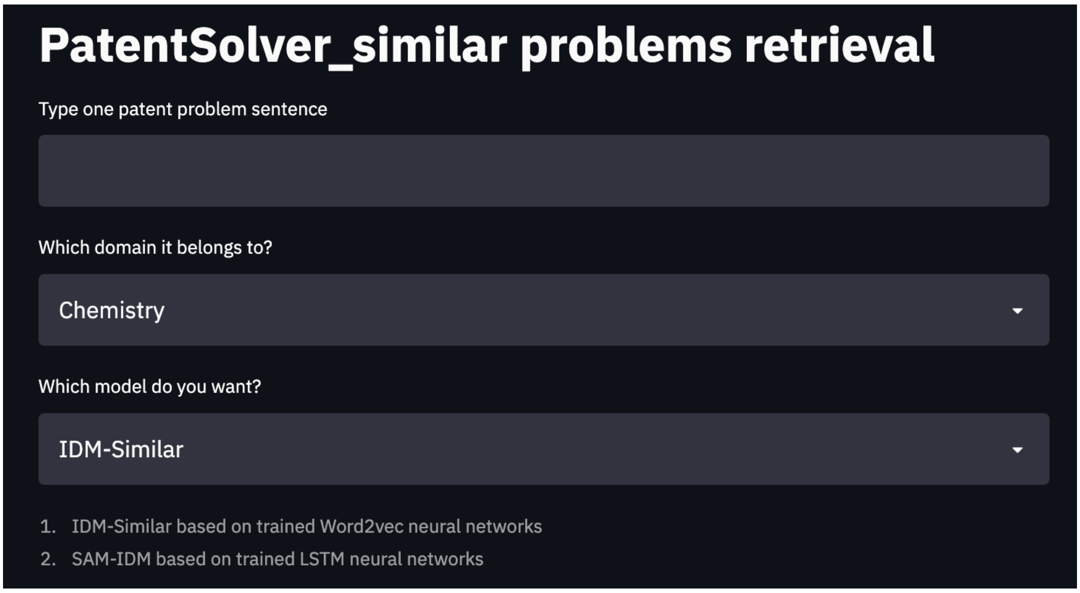


FIGURE 7.1: The welcome page of similar problem retrieval

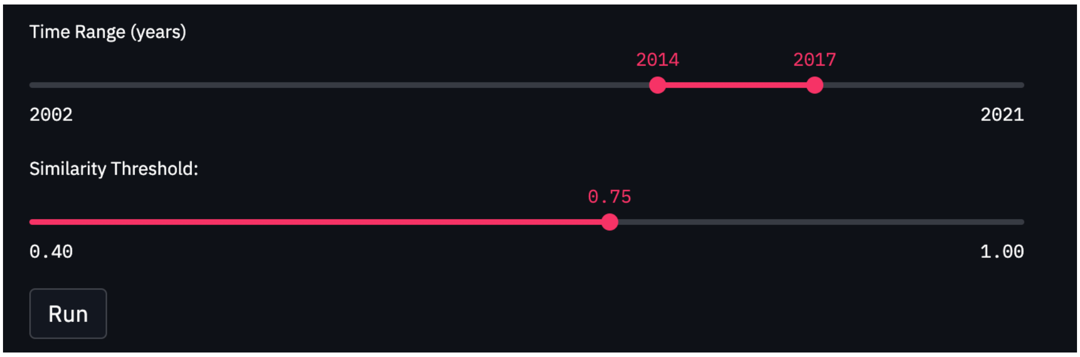


FIGURE 7.2: The parameter restriction page of similar problem retrieval

As shown in Fig. 7.1, the running of PatentSolver starts with the choice of the target problem sentence. Then, we provide a function to let users choose the domain that the target problem users typed belongs to. It contains eight different U.S. patent domains we introduced in Section 4.4.1 to reduce the comparison consumption that is introduced in Section 4.3. After that, we provide two trained models called IDM-Similar in Section 4.2 and SAM-IDM in Section 4.3 separately according to two different neural networks, Word2vec and LSTMs. This function could help users compare different problem retrieval results from our two models.

In the parameter constraint interface, as illustrated in Fig. 7.2, PatentSovler provides two functions for users. The first function is *Time Range*. It aims to let users choose a time range for the compared patents. The longer time distance suggests that more years of the patent problem sentences are going to be used for the comparison with the target sentences. However, it may

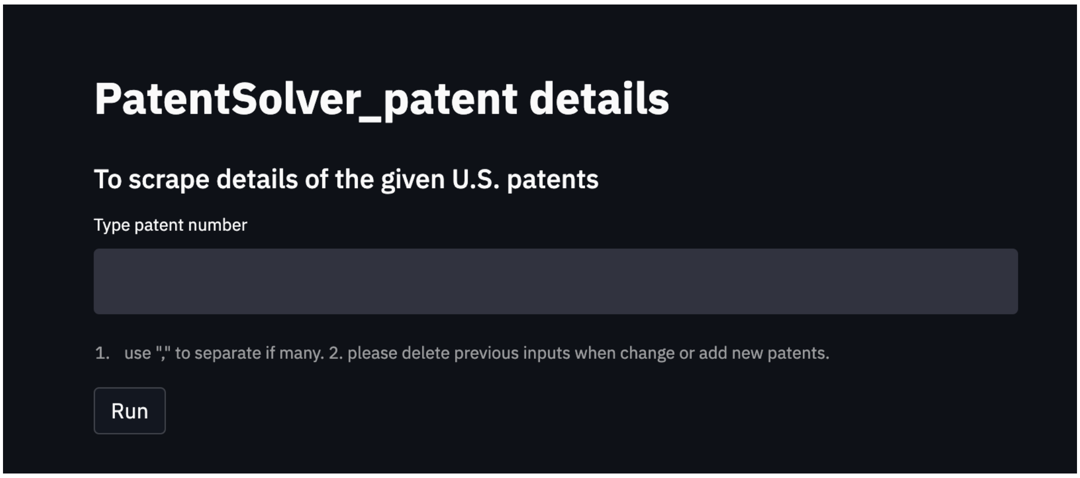


FIGURE 7.3: The welcome page of patent details

contribute to a greater waiting time to achieve results due to more computational consumption. The function of *Similarity Threshold* lets users be able to choose different similarity thresholds to evaluate the generated results from models. Indeed, a higher threshold will usually provide more precious similarity comparison results but also a lower recall rate. A lower threshold may generate lower precision but a higher recall rate. Users can modify the similarity threshold in this function to achieve a suitable threshold for their downstream tasks. Besides, since the different nature of IDM-Similar and SAM-IDM models, we separately set two different pre-chosen similarity thresholds to provide the reference for users. At the bottom of the surface, users may eventually click on the *Run* button to run the software for receiving the list of similar problem sentences for the target one.

7.3 Patent Details Statistic GUI

We introduce the function of the U.S. patent details scraper in this section.

As shown in Fig. 7.3, the function of the patent details scraper firstly provides a typing box for users to input the patent numbers they aim at. We especially optimize the input pattern to avoid imperceptible typing errors letting software crush. Indeed, we optimized two parts as follows:

- Space between two input patent numbers is not sensitive for the software. Thus, PatentSovler is not going to crush when users leave a space between input patent numbers after "," due to different users' typing behaviours. For instance, *"US10106875,US10106833"* and *"US10106875, US10106833"*, both of these inputs can be accepted by our software

TABLE 7.1: The items of patent details

Item	Explanation
patent_number	The target patent number
pub_date	The publication date of the patent
priority_date	The priority date of the patent
grant_date	The grant date of the patent
inventor_name	The inventors of patent
count_inventor_name	The statistical number of inventors
assignee_name	The original assignees to patent
count_assignee_name	The statistical number of original assignees
assignee_name_current	The current assignees to patent
count_assignee_name_current	The statistical number of current assignees
forward_cite_no_family	The forward citations that are not family-to-family cites
count_forward_cite_no_family	The statistical number of forward citations with no family cites
forward_cite_yes_family	The forward citations that are family-to-family cites
count_forward_cite_yes_family	The statistical number of forward citations with family cites
backward_cite_no_family	The backward citations that are not family-to-family cites
count_backward_cite_no_family	The statistical number of backward citations with no family cites
backward_cite_yes_family	The backward citations that are family-to-family cites
count_backward_cite_yes_family	The statistical number of backward citations with family cites
patent_link	The link to specific Google patent website

since our optimization behind the software. Otherwise, the second type could let software crush.

- For U.S. patent numbers, there are two ways to present them, with domain marks like B2 of US10106875B2 and without domain marks like US10106833, as illustrated in Fig. 7.4. In PatentSovler, these two expressions of patent number have been well accepted without the software crush. It improves the robustness of software and provides more convenience to users.

With these two aforementioned optimized functions and our model behind PatentSolver, PatentSolver is eventually able to retrieve details of U.S. patents. As shown in Fig. 7.4, we especially input two U.S. patents randomly, US10106875B2 and US10106833, as an example. After *Run* the software, as shown in Table. 7.1, the details of given patents with 19 items are displayed. In particular, the patent link can guide users to the corresponding web page on the Google patent website[5]. Furthermore, some of these patent features are used in the ranking of inventive solutions. *Download* function is eventually capable of letting users download the result as a CSV file.

[5] https://patents.google.com/

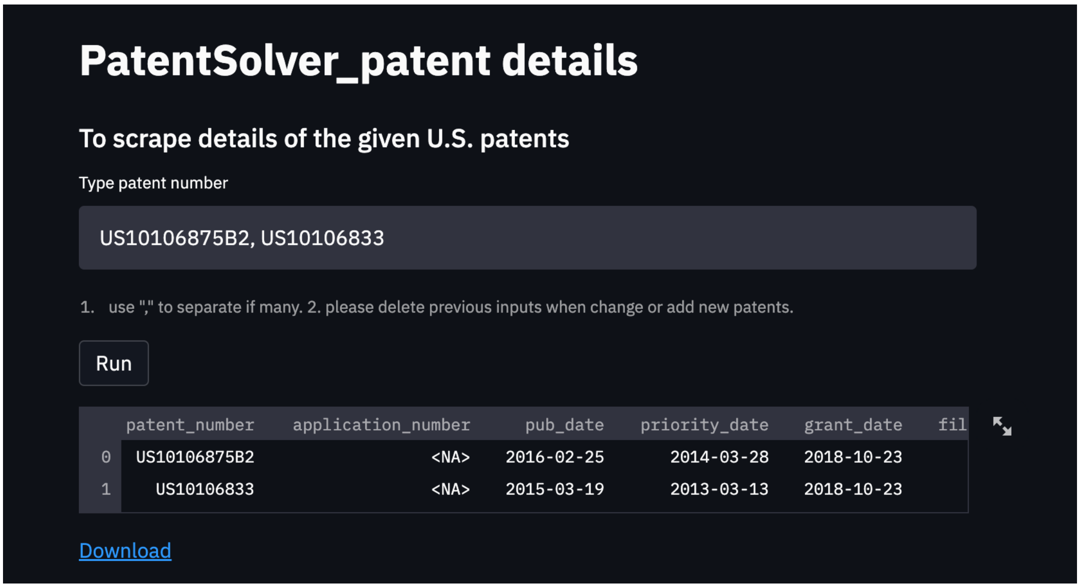


FIGURE 7.4: The page of patent details

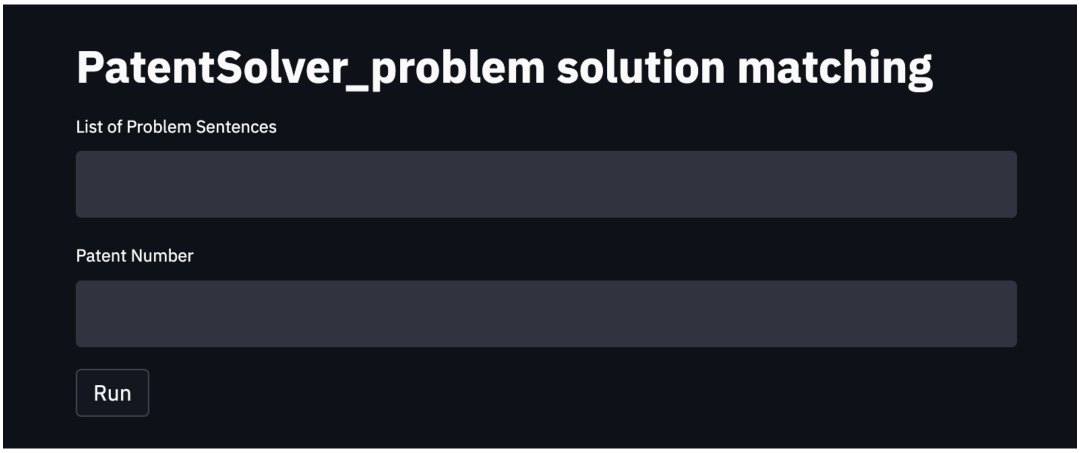


FIGURE 7.5: The page of problem-solution matching

7.4 Inventive Solutions Matching GUI

We introduce the function of problem-solution matching in U.S. patents in this section.

With the function of Section 7.2, a list of similar problem sentences from different domains patents corresponding to the target problem will be generated. To provide corresponding solutions to these problems, we build the function of inventive solutions matching, as illustrated in Fig. 7.5. Users firstly input the list of similar problems that are generated by the function of similar problems retrieval in Section 7.2. After that, clicking on the *Run* button can achieve the list of corresponding solutions linked to problems.

7.5 Inventive Solutions Rankings GUI

We introduce the function of latent inventive solutions ranking from U.S. patents in this section.

With results from the function of inventive solutions matching in Section 7.4, latent inventive solutions for the target problem is achieved. In order to rank them according to their inventiveness, we provide a function of inventive solutions ranking based on the PatRIS model in Chapter 6, as shown in Fig. 7.6. First of all, users input the target problem in the *Target Problem* box. They type the list of inventive solutions that are generated by the function of inventive solutions matching in Section 7.4. After that, six different inventiveness measuring features are provided to users to choose from. Various combinations with different chosen features may provide users with different ranking results. Users can eventually *Run* the software to achieve the final ranked inventive solutions linked to the target problem.

7.6 Summary

This chapter introduces a software prototype named PatentSolver. We have developed it to illustrate our thesis contributions and assist industrial engineers. The software enables the entire inventive solutions retrieval process from U.S. patent documents by starting with similar problem retrieval for the given problem. In addition, we also developed a function to scrape patent details features. This could provide users with more detailed information about patents. Overall, with PatentSolver, the entire process of the inventive solutions retrieval is automatic and provided to engineers to further facilitate R&D activities.

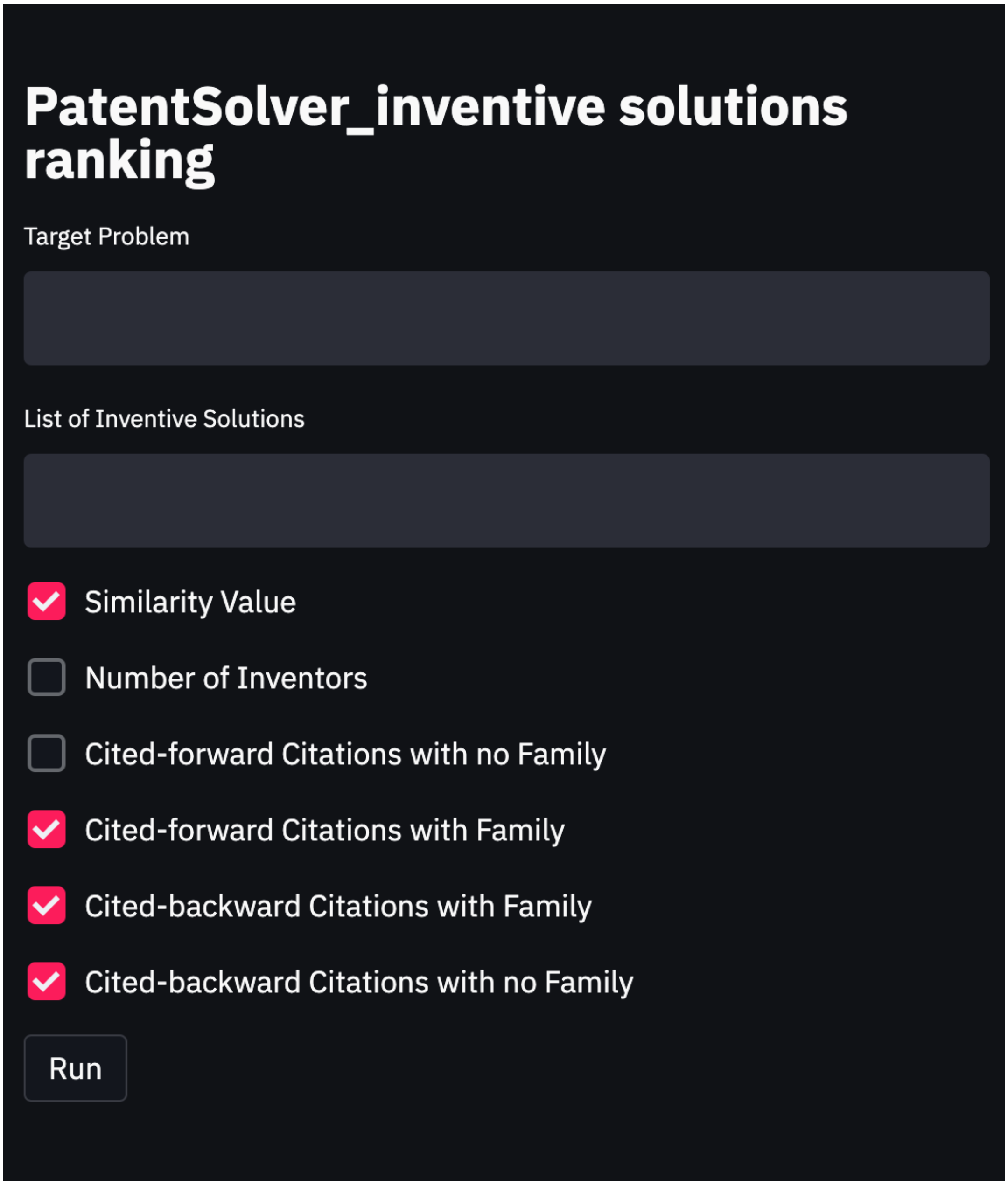


FIGURE 7.6: The page of inventive solutions ranking

Chapter 8

Conclusions and Future Work

In the era of Industry 4.0, digitization touches almost all departments of a company. Nevertheless, R&D is an exception, as little has changed from the advent of numerical simulations to the use of various tools derived from brainstorming or more obsolete approaches (such as TRIZ). It requires years of experience to benefit these for an industrial organization. This becomes a crucial obstacle for companies to further improve their R&D activities. For several years, the use of Artificial Intelligence (AI) technologies in the context of assisting industrial use has been drawing more attention. Therefore, the goal of a form of R&D aided via AI is emerging and has relevance in the global industrial digitization movement. In this thesis, we pursue our objective of inventive problem solving, to offer engineers an automatic resolution to explore innovative knowledge in patent documents. Our work made it possible to avoid the use of TRIZ theory. Its lack of formalization and complex principles imply huge obstacles to implementing it, even for engineers, to understand it (Dubois et al., 2005).

More importantly, in recent years, the demand for creative engineering solutions has been growing rapidly against the background of increasing pressure on innovation in engineering (Shirwaiker and Okudan, 2008; Jardim-Goncalves et al., 2011; Smirnov et al., 2013). Exploring broader knowledge fields to achieve innovative inspirations has become a significant alternative to brace complex challenges during the resolution of industrial problems. Compared to scientific documents, we notice that patent documents play a more significant role to represent the latest inventive knowledge in various domains. On the other hand, as a significant part of the product development process, innovative ability plays a key role to survive in the fierce competition among companies (Hao et al., 2019; Renjith, Park, and Kremer, 2020; Kusiak, 2016). Although engineers have realized the significance of knowledge exchange for successful product creation and development(Whiteside et al., 2009), creative knowledge remains intrinsically linked to the people who use it (Girodon et al., 2015). Obviously, having a broad understanding of different domain knowledge is impossible for the most engineers.

To address the aforementioned challenges, in this thesis, we developed an entirely automatic process to improve the flexibility and performance of classical TRIZ theory on the inventive problem-solving task by using NLP techniques. Besides, full use of a large number of published U.S. patent documents allows us to make use of the huge knowledge corpus in different domains. Moreover, different types of NLP techniques, including deep learning approaches, are combined into our models to benefit from the advancement of AI techniques to better facilitate the inventive problem-solving task. Therefore, our work represents the first step to help engineers who are without a broad knowledge understanding to refer to the innovative knowledge contained in various domains patents to obtain innovative inspirations.

8.1 Contributions

In this section, the contributions of the thesis are recalled. For each part of the contribution, the achieved results and general conclusions are illustrated.

In Chapter 4, two similar problem retrieval models called IDM-Similar and SAM-IDM are proposed. The similar problem retrieval task is the core component of the entire inventive solutions retrieval process. They are developed according to semantic similarity computation techniques. Indeed, Word2vec neural networks and LSTM neural networks are separately used on the IDM-Similar model and SAM-IDM model. Furthermore, with the base of the IDM-Similar model according to Word2vec neural networks, SAM-IDM further improves the performance of sentence semantic similarity computation. The specificity of LSTM neural networks that SAM-IDM uses has made the model learn longer context information in order to improve the ability of similar sentence computing. Especially, for sentences in patent documents, they tend to be longer than generic sentences due to technical expressions of patent documents. SAM-IDM also improves the precision of the prediction compared to the IDM-Similar model. We especially present it by comparing the experiments. Various case studies on IDM-Similar and SAM-IDM are presented to illustrate the usage of the IDM-Similar model and SAM-IDM model. Moreover, a locking mechanism design case study that has been evaluated by published medical papers strongly shows the performance of our model. This semantic similarity computation-related work enables the similar problem retrieval task from different domains patents. It also provides convenience to the next task of problem-solution matching.

In Chapter 5, a problem-solution matching model named IDM-Matching is proposed. Within the model, a Question Answering (QA) system is designed. We firstly pack problems as queries by the fixed interrogative format to achieve the queries for the QA system. A state-of-the-art pre-trained language model called XLNet is then leveraged to predict answers to queries according to context information around queries. These answers are seen as corresponding solutions to target problems. With IDM-Matching, we can build links between problems and solutions in patent documents. In addition, a real-world case study from the U.S. patent about electrically conductive touch pen is used to present our model's performance in detail. Its experimental results eventually illustrate the practical usage in the problem-solution matching task in patent documents.

Chapter 6 is devoted to ranking latent inventive solutions for the target problem. In this chapter, we proposed a model called PatRIS according to the multiple criteria decision analysis approach. Within the model, several patent inventiveness indicators and the similarity indicator with different weights are chosen as ranking features. PatRIS aims to address the ranking issue when latent inventive solutions numbers explode. Several case studies based on real-world U.S. patent documents and experimental analysis are used to illustrate the performance of our model. In general, PatRIS can provide a major step for engineers to find the most possible inventive solutions to the given problems.

Chapter 7 presents PatentSolver, which is a software prototype that combines problem retrieval (IDM-Similar and SAM-IDM), problem-solution matching (IDM-Matching), inventive solutions ranking (PatRIS), on top of the function of patent details scraper. This prototype aims to implement all the proposed approaches on real-world U.S. patent corpus, for automating the inventive solutions retrieval task. It enables the entire inventive solutions retrieval process. Besides all implemented contributions derived from the previous Chapters, we implement patent details scraper. Patent details scraper in PatentSolver is designed to collect patent details and statistical results.

8.2 Perspectives

The contributions summarized in the previous section may induce potential future research. In this section, we detail the following perspectives:

- The first future work is the evolution of the Patent Extractor (Souili, Cavallucci, and Rousselot, 2015a) we used in Chapter 4 for extracting problem sentences of patent documents. A more precise and accurate problem extraction tool can improve the performance of the problem-solution matching task and inventive solution ranking task. Patent Extractor is designed according to lexico-syntactic patterns. Within it, generic linguistic markers are used as keywords to identify and extract problem sentences, such as linguistic markers expressing negative notions. But we notice that, in our use, the precision of identifying problems is not stable in several cases. To deal with this issue, supervised machine learning approaches could be used to improve it. Manual labelling and evaluation of more problem sentences via initial Patent Extractor could be a direction to build labelled datasets. With these labelled datasets, we may fine-tune state-of-the-art language models like models based on BERT (Devlin et al., 2018) or GPT-3 (Brown et al., 2020), to further improve the performance of problem sentences retrieval.

- The second perspective is to improve the performance of the similar problem retrieval task. In Chapter 4, Word2vec neural networks and LSTM neural networks are separately used to build similarity computation models to retrieve similar problems from different domain patents. The learning ability of models on long context information is the main challenge to predict similar problems. Especially, numerous long sentences with rare unique words may exist in patent documents due to their own nature. It generates a crucial obstacle to most models to better learn semantic representations. Thus, according to the progress of NLP techniques, in future work, we aim to involve a state-of-the-art language model based on Transformer (Vaswani et al., 2017) techniques, which is trained on bigger datasets for a similar problem retrieval task. Its attention-mechanism should be able to let the model learn longer context information and more accurately remember which parts of information are important to predict similar problems. For instance, the model could behave like humans on reading the text. Humans always focus on the word they read but at the same time, their minds still hold the significant keywords of the text in memory to provide context. This ability is significant for our model to extract similar problems from different domains patents.

- The third future work is the reduction of the running time for the similarity computation. In Chapter 4, we introduced our reduction strategy to reduce the huge comparison time consumption. However, for building a real-time inventive solutions retrieval software on a large size of the patent dataset, it is still far from the expectations of users. Therefore, in future work, we aim to build a summarization model to summarize the main problem the patent manages to solve. In other words, this allows us to model each patent as one main problem. It further reduces the computation time when we compare the target problem with different domains patents' problems.

- The fourth perspective is to improve the capability of the system to extract more reliable inventive solutions for target problems. In Chapter 6, we introduced our inventive solutions ranking model according to the metric of its patent's corresponding inventiveness indicators and similarity value. However, how to ensure that each latent inventive solution is full of inventiveness is a crucial concern for all users. Since finding the "gold-standard" ground truth of verifying the inventiveness among solutions is always an open problem, no work has a complete effective evaluation approach. In order to handle this challenge, in future work, we aim to combine multiple knowledge sources such as scientific papers, professional blogs, and scientific news to build the knowledge graph. This could be seen as a hybrid evaluation metric.

Appendix A

40 Inventive Principles

The 40 Inventive Principles (IP) are used with the contradiction matrix to solve technical contradictions. This list with the so-called "sub-principles" that intend to help clarify the meaning of the principles was taken from the TRIZ Journal [1].

- Inventive Principle 1: Segmentation
  - Divide an object into independent parts.
  - Make an object easy to disassemble.
  - Increase the degree of fragmentation or segmentation.
- Inventive Principle 2: Taking out
  - Separate an interfering part or property from an object, or single out the only necessary part (or property) of an object.
- Inventive Principle 3: Local quality
  - Change an object's structure from uniform to non-uniform, change an external environment (or external influence) from uniform to non-uniform.
  - Make each part of an object function in conditions most suitable for its operation.
  - Make each part of an object fulfill a different and useful function.
- Inventive Principle 4: Asymmetry
  - Change the shape of an object from symmetrical to asymmetrical.
  - If an object is asymmetrical, increase its degree of asymmetry.
- Inventive Principle 5: Merging

[1] https://the-trizjournal.com/

  - Bring closer together (or merge) identical or similar objects, assemble identical or similar parts to perform parallel operations.
  - Make operations contiguous or parallel; bring them together in time.

- Inventive Principle 6: Universality
  - Make a part or object perform multiple functions; eliminate the need for other parts.

- Inventive Principle 7: “Nested doll"
  - Place one object inside another; place each object, in turn, inside the other.
  - Make one part pass through a cavity in the other.

- Inventive Principle 8: Anti-weight
  - To compensate for the weight of an object, merge it with other objects that provide lift.
  - To compensate for the weight of an object, make it interact with the environment (e.g. use aerodynamic, hydrodynamic, buoyancy and other forces).

- Inventive Principle 9: Preliminary anti-action
  - If it will be necessary to do an action with both harmful and useful effects, this action should be replaced with anti-actions to control harmful effects.
  - Create beforehand stresses in an object that will oppose known undesirable working stresses later on.

- Inventive Principle 10: Preliminary action
  - Perform, before it is needed, the required change of an object (either fully or partially).
  - Pre-arrange objects such that they can come into action from the most convenient place and without losing time for their delivery.

- Inventive Principle 11: Beforehand cushioning
  - Prepare emergency means beforehand to compensate for the relatively low reliability of an object.

- Inventive Principle 12: Equipotentiality
  - In a potential field, limit position changes (e.g. change operating conditions to eliminate the need to raise or lower objects in a gravity field).
- Inventive Principle 13: "The other way round"
  - Invert the action(s) used to solve the problem (e.g. instead of cooling an object, heat it).
  - Make movable parts (or the external environment) fixed, and fixed parts movable).
  - Turn the object (or process) 'upside down'.
- Inventive Principle 14: Spheroidality - Curvature
  - Instead of using rectilinear parts, surfaces, or forms, use curvilinear ones; move from flat surfaces to spherical ones; from parts shaped as a cube (parallelepiped) to ball-shaped structures.
  - Use rollers, balls, spirals, domes.
  - Go from linear to rotary motion, use centrifugal forces.
- Inventive Principle 15: Dynamics
  - Allow (or design) the characteristics of an object, external environment, or process to change to be optimal or to find an optimal operating condition.
  - Divide an object into parts capable of movement relative to each other.
  - If an object (or process) is rigid or inflexible, make it movable or adaptive.
- Inventive Principle 16: Partial or excessive actions
  - If 100 percent of an object is hard to achieve using a given solution method then, by using 'slightly less' or 'slightly more' of the same method, the problem may be considerably easier to solve.
- Inventive Principle 17: Another dimension
  - To move an object in two- or three-dimensional space.

  - Use a multi-story arrangement of objects instead of a single-story arrangement.
  - Tilt or re-orient the object, lay it on its side.
  - Use 'another side' of a given area.

- Inventive Principle 18: Mechanical vibration
  - Cause an object to oscillate or vibrate.
  - Increase its frequency (even up to the ultrasonic).
  - Use an object's resonant frequency.
  - Use piezoelectric vibrators instead of mechanical ones.
  - Use combined ultrasonic and electromagnetic field oscillations.

- Inventive Principle 19: Periodic action
  - Instead of continuous action, use periodic or pulsating actions.
  - If an action is already periodic, change the periodic magnitude or frequency.
  - Use pauses between impulses to perform a different action.

- Inventive Principle 20: Continuity of useful action
  - Carry on work continuously; make all parts of an object work at full load, all the time.
  - Eliminate all idle or intermittent actions or work.

- Inventive Principle 21: Skipping
  - Conduct a process , or certain stages (e.g. destructible, harmful or hazardous operations) at high speed.

- Inventive Principle 22: "Blessing in disguise" or "Turn Lemons into Lemonade"
  - Use harmful factors (particularly, harmful effects of the environment or surroundings) to achieve a positive effect.
  - Eliminate the primary harmful action by adding it to another harmful action to resolve the problem.
  - Amplify a harmful factor to such a degree that it is no longer harmful.

- Inventive Principle 23: Feedback
  - Introduce feedback (referring back, cross-checking) to improve a process or action.
  - If feedback is already used, change its magnitude or influence.
- Inventive Principle 24: 'Intermediary'
  - Use an intermediary carrier article or intermediary process.
  - Merge one object temporarily with another (which can be easily removed).
- Inventive Principle 25: Self-service
  - Make an object serve itself by performing auxiliary helpful functions.
  - Use waste resources, energy, or substances.
- Inventive Principle 26: Copying
  - Instead of an unavailable, expensive, fragile object, use simpler and inexpensive copies.
  - Replace an object, or process with optical copies.
  - If visible optical copies are already used, move to infrared or ultraviolet copies.
- Inventive Principle 27: Cheap short-living objects
  - Replace an inexpensive object with a multiple of inexpensive objects, comprising certain qualities (such as service life, for instance).
- Inventive Principle 28: Mechanics substitution
  - Replace a mechanical means with a sensory (optical, acoustic, taste or smell) means.
  - Use electric, magnetic and electromagnetic fields to interact with the object.
  - Change from static to movable fields, from unstructured fields to those having structure.
  - Use fields in conjunction with field-activated (e.g. ferromagnetic) particles.

- Inventive Principle 29: Pneumatics and hydraulics
  - Use gas and liquid parts of an object instead of solid parts (e.g. inflatable, filled with liquids, air cushion, hydrostatic, hydro-reactive).
- Inventive Principle 30: Flexible shells and thin films
  - Use flexible shells and thin films instead of three dimensional structures.
  - Isolate the object from the external environment using flexible shells and thin films.
- Inventive Principle 31: Porous materials
  - Make an object porous or add porous elements (inserts, coatings, etc.).
  - If an object is already porous, use the pores to introduce a useful substance or function.
- Inventive Principle 32: Color changes
  - Change the color of an object or its external environment.
  - Change the transparency of an object or its external environment.
- Inventive Principle 33: Homogeneity
  - Make objects interacting with a given object of the same material (or material with identical properties).
- Inventive Principle 34: Discarding and recovering
  - Make portions of an object that have fulfilled their functions go away (discard by dissolving, evaporating, etc.) or modify these directly during operation.
  - Conversely, restore consumable parts of an object directly in operation.
- Inventive Principle 35: Change of physical and chemical parameters
  - Change the object's aggregate state.
  - Change concentration or consistency of the object.
  - Change the degree of flexibility of the object.
  - Change the temperature of the object or environment.

- Inventive Principle 36: Phase transitions
  - Use phenomena occurring during phase transitions (e.g. volume changes, loss or absorption of heat, etc.).
- Inventive Principle 37: Thermal expansion
  - Use thermal expansion (or contraction) of materials.
  - If thermal expansion is being used, use multiple materials with different coefficients of thermal expansion.
- Inventive Principle 38: Strong oxidants
  - Replace common air with oxygen-enriched air.
  - Replace enriched air with pure oxygen.
  - Expose air or oxygen to ionizing radiation.
  - Use ionized oxygen.
  - Replace ozonized (or ionized) oxygen with ozone.
- Inventive Principle 39: Inert atmosphere
  - Use inert gases instead of usual ones.
  - Add neutral parts or additives to the object.
- Inventive Principle 40: Composite materials
  - Change from uniform to composite (multiple) materials.

Appendix B

76 Inventive Standards

B.1 Some Extra Precision on Inventive Standards

The 76 Inventive Standard Solutions (IS) are divided into five classes with various sub-classes that are used depending on the type of engineering problem they solve. The five classes are:

- Class 1: Building and Destruction of Substance-Field Models (13 IS)
- Class 2: Development of Substance-Field Models (23 IS)
- Class 3 Transition to Super-system and Micro level (6 IS)
- Class 4: Standards for Detection and Measuring (17 IS)
- Class 5: Standards on Application of Standards (17 IS)

This appendix describes the classes and the sub-classes.

B.1.1 Class 1: Building and Destruction of Substance-Field Models

Class 1 aims to solve problems by building or destroying the Su-Field Models if they are incomplete or have harmful functions. Class 1 contains two sub-classes containing 13 IS:

Sub-class 1.1 Building of Su-Fields (if incomplete) (8 IS)

The major recommendations from this sub-class are:

- Make the Su-Field complete.
- Make it minimally workable by introducing an internal additive.

- Make it minimally workable by introducing an external additive.
- Use minimal - maximal mode (add more and remove the extras; add less and enhance locally).

Sub-class 1.2 Destruction of Su-Field (harms) (5 IS)

The major recommendations from this sub-class are:

- Introduce a third substance between the given two substances.
- Introduce a third substance from the super-system.
- Introduce a third substance that is a modification of one of the given two substances.
- Introduce a sacrificial substance.
- Introduce a field that counteracts the harmful field.

B.1.2 Class 2: Development of Substance-Field Models

This class is used to improve the efficiency of engineered systems by introducing small modifications. It provides conceptual solutions on how to improve and develop the system. The main recommendations in this class are:

- Use of chain Su-Fields
- Use of double Su-Fields
- Segmentation (including porosity increase)
- Dynamisation
- Rhythm coordination
- Use of magnetic substances.

Class 2 contains 4 sub-classes and 23 IS:

Sub-class 2.1 Transition to complex Su-Field Models (2 IS)

Sub-class 2.2 Evolution of Su-Field Models (6 IS)

Sub-class 2.3 Evolution of rhythms (3 IS)

Sub-class 2.4 Complex forced Su-Field Models (12 IS)

B.1.3 Class 3: System Transitions and Evolution-Transition to Super-system and Sub-system

Problems in this class are solved by developing solutions at different levels in the system (super-systems or sub-systems). The main recommendation in this class is how to improve the system by combining elements or combining with other systems. Class 3 contains 2 sub-classes containing 6 IS:

Sub-class 3.1 Simplicity-complexity-simplicity (mono-bi-poly) and increasing flexibility and dynamisation (Transition to super-system and to bi and poly systems; use no links, rigid links, flexible links, "field" links) (5 IS)

Sub-class 3.2 Transition to micro-level (examine the sub-system, use smart substances) (1 IS)

B.1.4 Class 4: Solutions for Detection and Measurement

This class is used for solving measuring or detection problems in engineering systems. These solutions contain many distinguished features, especially the use of indirect methods and the use of copies. The major recommendations of this class are:

- Try to change the system so that there is no need to measure/detect.
- Measure a copy.
- Introduce a substance that generates a field (introduce a mark internally or externally).

Class 4 contains 5 sub-classes and 17 IS:

Sub-class 4.1 Indirect Methods (3 IS)

Sub-class 4.2 Create or Build a Measurement System (4 IS)

Sub-class 4.3 Enhancing the Measurement System (3 IS)

Sub-class 4.4 Measure Ferromagnetic-field (5 IS)

Sub-class 4.5 Direction of Evolution of the Measuring Systems (2 IS)

B.1.5 Class 5: Standards on Application of Standards

With the help of previous four classes of the Standard Solutions, Class 5 is additionally helpful for the further general improvements and simplification of systems. These Standard Solutions provide recommendations of how to introduce new substances or fields or use scientific effects more effectively after applying the relevant Standard Solutions in the previous four classes.

Class 5 solutions are helpful when simplifying or pruning the system to remove components or reduce the intensity of the associated interactions. The first four classes of standard solutions mentioned above often lead to increased complexity of the solution because something is often added to the system in order to solve the problem. This fifth class of solutions illustrates how to achieve something extra by simplifying, but without introducing anything new. The useful recommendations from this class are:

- Instead of a substance, introduce a field.
- Instead of a substance, introduce a void.
- Introduce a substance for a limited time.
- Introduce a little bit of a substance, but in a very concentrated way.
- Use phase changes.
- Get the substance or environment to change themselves to solve the problem.
- Use segmentation.

Class 5 contains 5 sub-classes and 17 IS:

Sub-class 5.1 Indirect methods for introducing substances under restricted conditions (4 IS)

Sub-class 5.2 Introducing fields under restricted conditions (3 IS)

Sub-class 5.3 Phase transitions (5 IS)

Sub-class 5.4 Clever use of natural phenomena (2 IS)

Sub-class 5.5 Generating higher or lower forms of substances (3 IS)

Appendix C

11 Separation Methods

C.1 11 Separation Methods

11 separation methods (SM) are used to solve physical contradictions. This list was taken from the TRIZ Journal [1].

Separation in Space

- SM1. Separation of conflicting properties in space.

Separation in Time

- SM2. Separation of conflicting properties in time.

Separation by system transition

- SM3. Combination of homogeneous or heterogeneous systems into a super-system.
- SM4. Transition from a system to an anti-system, or combination of system with anti-system.
- SM5. The entire system has a property X while its parts have a property opposite to X (anti-X).
- SM6. System transition 2: transition to system that works on the micro-level. Separation by phase transition
- SM7. Substitution of the phase state of a system's part or external environment.
- SM8. Dual phase state of a system part (using substances capable of converting from one phase to another according to operating conditions).

[1] https://the-trizjournal.com/

- SM9. Using of phenomena associated with phase transitions.
- SM10. Substitution of a mono-phase substance with a dual-phase state.

Separation by physical-chemical transition

- SM11. Substance appearance-disappearance as a result of decomposition-combination, ionization-recombination.

Appendix D

39 Generic Engineering Parameters

1. Weight of moving object
2. Weight of stationary object
3. Length of moving object
4. Length of stationary object
5. Area of moving object
6. Area of stationary object
7. Volume of moving object
8. Volume of stationary object
9. Speed
10. Force (intensity)
11. Stress or pressure
12. Shape
13. Stability of object's composition
14. Strength
15. Duration of action of moving object
16. Duration of action of stationary object
17. Temperature
18. Illumination intensity
19. Use of energy by moving object
20. Use of energy by stationary object
21. Power
22. Loss of energy
23. Loss of substance
24. Loss of information
25. Loss of time
26. Quantity of substance or the matter
27. Reliability
28. Measurement accuracy
29. Manufacturing precision

30. Object-affected harmful factors
31. Object-generated harmful factors
32. Ease of manufacture
33. Ease of operation
34. Ease of repair
35. Adaptability or versatility
36. Device complexity
37. Difficulty of detecting and measuring
38. Extent of automation
39. Productivity

Appendix E

Contradiction Matrix

The "Y-axis" of the contradiction matrix stands for the parameters to be improved and the "X-axis" shows the "undesired results", that is, the parameters to be deteriorated.

#	Improving Feature / Worsening Feature	Weight of moving object	Weight of stationary object	Length of moving object	Length of stationary object	Area of moving object	Area of stationary object	Volume of moving object	Volume of stationary object	Speed	Force (Intensity)	Stress or pressure	Shape	Stability of the object's composition	Strength	Duration of action of moving object	Duration of action of stationary object	Temperature	Illumination intensity	Use of energy by moving object	Use of energy by stationary object
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	Weight of moving object	+		15, 8, 29, 34		29, 17, 38, 34		29, 2, 40, 28		2, 8, 15, 38	8, 10, 18, 37	10, 36, 37, 40	10, 14, 35, 40	1, 35, 19, 39	28, 27, 18, 40	5, 34, 31, 35		6, 29, 4, 38	19, 1, 32	35, 12, 34, 31	
2	Weight of stationary object		+		10, 1, 29, 35		35, 30, 13, 2		5, 35, 14, 2		8, 10, 19, 35	13, 29, 10, 18	13, 10, 29, 14	26, 39, 1, 40	28, 2, 10, 27		2, 27, 19, 6	28, 19, 32, 22	19, 32, 35		18, 19, 28, 1
3	Length of moving object	8, 15, 29, 34		+		15, 17, 4		7, 17, 4, 35		13, 4, 8	17, 10, 4	1, 8, 35	1, 8, 10, 29	1, 8, 15, 34	8, 35, 29, 34	19		10, 15, 19	32	8, 35, 24	
4	Length of stationary object		35, 28, 40, 29		+		17, 7, 10, 40		35, 8, 2, 14		28, 10	1, 14, 35	13, 14, 15, 7	39, 37, 35	15, 14, 28, 26		1, 10, 35	3, 35, 38, 18	3, 25		
5	Area of moving object	2, 17, 29, 4		14, 15, 18, 4		+		7, 14, 17, 4		29, 30, 4, 34	19, 30, 35, 2	10, 15, 36, 28	5, 34, 29, 4	11, 2, 13, 39	3, 15, 40, 14	6, 3		2, 15, 16	15, 32, 19, 13	19, 32	
6	Area of stationary object		30, 2, 14, 18		26, 7, 9, 39		+				1, 18, 35, 36	10, 15, 36, 37		2, 38	40		2, 10, 19, 30	35, 39, 38			
7	Volume of moving object	2, 26, 29, 40		1, 7, 4, 35		1, 7, 4, 17		+		29, 4, 38, 34	15, 35, 36, 37	6, 35, 36, 37	1, 15, 29, 4	28, 10, 1, 39	9, 14, 15, 7	6, 35, 4		34, 39, 10, 18	2, 13, 10	35	
8	Volume of stationary object		35, 10, 19, 14	19, 14	35, 8, 2, 14				+		2, 18, 37	24, 35	7, 2, 35	34, 28, 35, 40	9, 14, 17, 15		35, 34, 38	35, 6, 4			
9	Speed	2, 28, 13, 38		13, 14, 8		29, 30, 34		7, 29, 34		+	13, 28, 15, 19	6, 18, 38, 40	35, 15, 18, 34	28, 33, 1, 18	8, 3, 26, 14	3, 19, 35, 5		28, 30, 36, 2	10, 13, 19	8, 15, 35, 38	
10	Force (Intensity)	8, 1, 37, 18	18, 13, 1, 28	17, 19, 9, 36	28, 10	19, 10, 15	1, 18, 36, 37	15, 9, 12, 37	2, 36, 18, 37	13, 28, 15, 12	+	18, 21, 11	10, 35, 40, 34	35, 10, 21	35, 10, 14, 27	19, 2		35, 10, 21		19, 17, 10	1, 16, 36, 37
11	Stress or pressure	10, 36, 37, 40	13, 29, 10, 18	35, 10, 36	35, 1, 14, 16	10, 15, 36, 28	10, 15, 36, 37	6, 35, 10	35, 24	6, 35, 36	36, 35, 21	+	35, 4, 15, 10	35, 33, 2, 40	9, 18, 3, 40	19, 3, 27		35, 39, 19, 2		14, 24, 10, 37	
12	Shape	8, 10, 29, 40	15, 10, 26, 3	29, 34, 5, 4	13, 14, 10, 7	5, 34, 4, 10		14, 4, 15, 22	7, 2, 35	35, 15, 34, 18	35, 10, 37, 40	34, 15, 10, 14	+	33, 1, 18, 4	30, 14, 10, 40	14, 26, 9, 25		22, 14, 19, 32	13, 15, 32	2, 6, 34, 14	
13	Stability of the object's composition	21, 35, 2, 39	26, 39, 1, 40	13, 15, 1, 28	37	2, 11, 13	39	28, 10, 19, 39	34, 28, 35, 40	33, 15, 28, 18	10, 35, 21, 16	2, 35, 40	22, 1, 18, 4	+	17, 9, 15	13, 27, 10, 35	39, 3, 35, 23	35, 1, 32	32, 3, 27, 16	13, 19	27, 4, 29, 18
14	Strength	1, 8, 40, 15	40, 26, 27, 1	1, 15, 8, 35	15, 14, 28, 26	3, 34, 40, 29	9, 40, 28	10, 15, 14, 7	9, 14, 17, 15	8, 13, 26, 14	10, 18, 3, 14	10, 3, 18, 40	10, 30, 35, 40	13, 17, 35	+	27, 3, 26		30, 10, 40	35, 19	19, 35, 10	35
15	Duration of action of moving object	19, 5, 34, 31		2, 19, 9		3, 17, 19		10, 2, 19, 30		3, 35, 5	19, 2, 16	19, 3, 27	14, 26, 28, 25	13, 3, 35	27, 3, 10	+		19, 35, 39	2, 19, 4, 35	28, 6, 35, 18	
16	Duration of action by stationary object		6, 27, 19, 16		1, 40, 35				35, 34, 38					39, 3, 35, 23			+	19, 18, 36, 40			
17	Temperature	36,22, 6, 38	22, 35, 32	15, 19, 9	15, 19, 9	3, 35, 39, 18	35, 38	34, 39, 40, 18	35, 6, 4	2, 28, 36, 30	35, 10, 3, 21	35, 39, 19, 2	14, 22, 19, 32	1, 35, 32	10, 30, 22, 40	19, 13, 39	19, 18, 36, 40	+	32, 30, 21, 16	19, 15, 3, 17	
18	Illumination intensity	19, 1, 32	2, 35, 32	19, 32, 16		19, 32, 26		2, 13, 10		10, 13, 19	26, 19, 6		32, 30	32, 3, 27	35, 19	2, 19, 6		32, 35, 19	+	32, 1, 19	32, 35, 1, 15
19	Use of energy by moving object	12,18,28, 31		12, 28		15, 19, 25		35, 13, 18		8, 35, 35	16, 26, 21, 2	23, 14, 25	12, 2, 29	19, 13, 17, 24	5, 19, 9, 35	28, 35, 6, 18	-	19, 24, 3, 14	2, 15, 19	+	-
20	Use of energy by stationary object		19, 9, 6, 27								36, 37			27, 4, 29, 18	35				19, 2, 35, 32	-	+
21	Power	8, 36, 38, 31	19, 26, 17, 27	1, 10, 35, 37		19, 38	17, 32, 13, 38	35, 6, 38	30, 6, 25	15, 35, 2	26, 2, 36, 35	22, 10, 35	29, 14, 2, 40	35, 32, 15, 31	26, 10, 28	19, 35, 10, 38	16	2, 14, 17, 25	16, 6, 19	16, 6, 19, 37	
22	Loss of Energy	15, 6, 19, 28	19, 6, 18, 9	7, 2, 6, 13	6, 38, 7	15, 26, 17, 30	17, 7, 30, 18	7, 18, 23	7	16, 35, 38	36, 38			14, 2, 39, 6	26			19, 38, 7	1, 13, 32, 15		
23	Loss of substance	35, 6, 23, 40	35, 6, 22, 32	14, 29, 10, 39	10, 28,24	35, 2, 10, 31	10, 18, 39, 31	1, 29, 30, 36	3, 39, 18, 31	10, 13, 28, 38	14, 15, 18, 40	3, 36, 37, 10	29, 35, 3, 5	2, 14, 30, 40	35, 28, 31, 40	28, 27, 3, 18	27, 16, 18, 38	21, 36, 39, 31	1, 6, 13	35, 18, 24, 5	28, 27, 12, 31
24	Loss of Information	10, 24, 35	10, 35, 5	1, 26	26	30, 26	30, 16		2, 22	26, 32						10	10		19		
25	Loss of Time	10, 20, 37, 35	10, 20, 26, 5	15, 2, 29	30, 24, 14, 5	26, 4, 5, 16	10, 35, 17, 4	2, 5, 34, 10	35, 16, 32, 18		10, 37, 36,5	37, 36,4	4, 10, 34, 17	35, 3, 22, 5	29, 3, 28, 18	20, 10, 28, 18	28, 20, 10, 16	35, 29, 21, 18	1, 19, 26, 17	35, 38, 19, 18	1
26	Quantity of substance/the matter	35, 6, 18, 31	27, 26, 18, 35	29, 14, 35, 18		15, 14, 29	2, 18, 40, 4	15, 20, 29		35, 29, 34, 28	35, 14, 3	10, 36, 14, 3	35, 14	15, 2, 17, 40	14, 35, 34, 10	3, 35, 10, 40	3, 35, 31	3, 17, 39		34, 29, 16, 18	3, 35, 31
27	Reliability	3, 8, 10, 40	3, 10, 8, 28	15, 9, 14, 4	15, 29, 28, 11	17, 10, 14, 16	32, 35, 40, 4	3, 10, 14, 24	2, 35, 24	21, 35, 11, 28	8, 28, 10, 3	10, 24, 35, 19	35, 1, 16, 11		11, 28	2, 35, 3, 25	34, 27, 6, 40	3, 35, 10	11, 32, 13	21, 11, 27, 19	36, 23
28	Measurement accuracy	32, 35, 26, 28	28, 35, 25, 26	28, 26, 5, 16	32, 28, 3, 16	26, 28, 32, 3	26, 28, 32, 3	32, 13, 6		28, 13, 32, 24	32, 2	6, 28, 32	6, 28, 32	32, 35, 13	28, 6, 32	28, 6, 32	10, 26, 24	6, 19, 28, 24	6, 1, 32	3, 6, 32	
29	Manufacturing precision	28, 32, 13, 18	28, 35, 27, 9	10, 28, 29, 37	2, 32, 10	28, 33, 29, 32	2, 29, 18, 36	32, 23, 2	25, 10, 35	10, 28, 32	28, 19, 34, 36	3, 35	32, 30, 40	30, 18	3, 27	3, 27, 40		19, 26	3, 32	32, 2	
30	Object-affected harmful factors	22, 21, 27, 39	2, 22, 13, 24	17, 1, 39, 4	1, 18	22, 1, 33, 28	27, 2, 39, 35	22, 23, 37, 35	34, 39, 19, 27	21, 22, 35, 28	13, 35, 39, 18	22, 2, 37	22, 1, 3, 35	35, 24, 30, 18	18, 35, 37, 1	22, 15, 33, 28	17, 1, 40, 33	22, 33, 35, 2	1, 19, 32, 13	1, 24, 6, 27	10, 2, 22, 37
31	Object-generated harmful factors	19, 22, 15, 39	35, 22, 1, 39	17, 15, 16, 22		17, 2, 18, 39	22, 1, 40	17, 2, 40	30, 18, 35, 4	35, 28, 3, 23	35, 28, 1, 40	2, 33, 27, 18	35, 1	35, 40, 27, 39	15, 35, 22, 2	15, 22, 33, 31	21, 39, 16, 22	22, 35, 2, 24	19, 24, 39, 32	2, 35, 6	19, 22, 18
32	Ease of manufacture	28, 29, 15, 16	1, 27, 36, 13	1, 29, 13, 17	15, 17, 27	13, 1, 26, 12	16, 40	13, 29, 1, 40	35	35, 13, 8, 1	35, 12	35, 19, 1, 37	1, 28, 13, 27	11, 13, 1	1, 3, 10, 32	27, 1, 4	35, 16	27, 26, 18	28, 24, 27, 1	28, 26, 27, 1	1, 4
33	Ease of operation	25, 2, 13, 15	6, 13, 1, 25	1, 17, 13, 12		1, 17, 13, 16	18, 16, 15, 39	1, 16, 35, 15	4, 18, 39, 31	18, 13, 34	28, 13 35	2, 32, 12	15, 34, 29, 28	32, 35, 30	32, 40, 3, 28	29, 3, 8, 25	1, 16, 25	26, 27, 13	13, 17, 1, 24	1, 13, 24	
34	Ease of repair	2, 27 35, 11	2, 27, 35, 11	1, 28, 10, 25	3, 18, 31	15, 13, 32	16, 25	25, 2, 35, 11	1	34, 9	1, 11, 10	13	1, 13, 2, 4	2, 35	11, 1, 2, 9	11, 29, 28, 27	1	4, 10	15, 1, 13	15, 1, 28, 16	
35	Adaptability or versatility	1, 6, 15, 8	19, 15, 29, 16	35, 1, 29, 2	1, 35, 16	35, 30, 29, 7	15, 16	15, 35, 29		35, 10, 14	15, 17, 20	35, 16	15, 37, 1, 8	35, 30, 14	35, 3, 32, 6	13, 1, 35	2, 16	27, 2, 3, 35	6, 22, 26, 1	19, 35, 29, 13	
36	Device complexity	26, 30, 34, 36	2, 26, 35, 39	1, 19, 26, 24	26	14, 1, 13, 16	6, 36	34, 26, 6	1, 16	34, 10, 28	26, 16	19, 1, 35	29, 13, 28, 15	2, 22, 17, 19	2, 13, 28	10, 4, 28, 15		2, 17, 13	24, 17, 13	27, 2, 29, 28	
37	Difficulty of detecting and measuring	27, 26, 28, 13	6, 13, 28, 1	16, 17, 26, 24	26	2, 13, 18, 17	2, 39, 30, 16	29, 1, 4, 16	2, 18, 26, 31	3, 4, 16, 35	30, 28, 40, 19	35, 36, 37, 32	27, 13, 1, 39	11, 22, 39, 30	27, 3, 15, 28	19, 29, 39, 25	25, 34, 6, 35	3, 27, 35, 16	2, 24, 26	35, 38	19, 35, 16
38	Extent of automation	28, 26, 18, 35	28, 26, 35, 10	14, 13, 17, 28	23	17, 14, 13		35, 13, 16		28, 10	2, 35	13, 35	15, 32, 1, 13	18, 1	25, 13	6, 9		26, 2, 19	8, 32, 19	2, 32, 13	
39	Productivity	35, 26, 24, 37	28, 27, 15, 3	18, 4, 28, 38	30, 7, 14, 26	10, 26, 34, 31	10, 35, 17, 7	2, 6, 34, 10	35, 37, 10, 2		28, 15, 10, 36	10, 37, 14	14, 10, 34, 40	35, 3, 22, 39	29, 28, 10, 18	35, 10, 2, 18	20, 10, 16, 38	35, 21, 28, 10	26, 17, 19, 1	35, 10, 38, 19	1

#	Improving Feature / Worsening Feature	Power	Loss of Energy	Loss of Substance	Loss of Information	Loss of Time	Quantity of substance	Reliability	Measurement accuracy	Manufacturing precision	Object-affected harmful factors	Object-generated harmful factors	Ease of manufacture	Ease of operation	Ease of repair	Adaptability or versatility	Device complexity	Difficulty of detecting and measuring	Extent of automation	Productivity
		21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
1	Weight of moving object	12, 36, 18, 31	6, 2, 34, 19	5, 35, 3, 31	10, 24, 35	10, 35, 20, 28	3, 26, 18, 31	1, 3, 11, 27	28, 27, 35, 26	28, 35, 26, 18	22, 21, 18, 27	22, 35, 31, 39	27, 28, 1, 36	35, 3, 2, 24	2, 27, 28, 11	29, 5, 15, 8	26, 30, 36, 34	28, 29, 26, 32	26, 35 18, 19	35, 3, 24, 37
2	Weight of stationary object	15, 19, 18, 22	18, 19, 28, 15	5, 8, 13, 30	10, 15, 35	10, 20, 35, 26	19, 6, 18, 26	10, 28, 8, 3	18, 26, 28	10, 1, 35, 17	2, 19, 22, 37	35, 22, 1, 39	28, 1, 9	6, 13, 1, 32	2, 27, 28, 11	19, 15, 29	1, 10, 26, 39	25, 28, 17, 15	2, 26, 35	1, 28, 15, 35
3	Length of moving object	1, 35	7, 2, 35, 39	4, 29, 23, 10	1, 24	15, 2, 29	29, 35	10, 14, 29, 40	28, 32, 4	10, 28, 29, 37	1, 15, 17, 24	17, 15	1, 29, 17	15, 29, 35, 4	1, 28, 10	14, 15, 1, 16	1, 19, 26, 24	35, 1, 26, 24	17, 24, 26, 16	14, 4, 28, 29
4	Length of stationary object	12, 8	6, 28	10, 28, 24, 35	24, 26,	30, 29, 14		15, 29, 28	32, 28, 3	2, 32, 10	1, 18		15, 17, 27	2, 25	3	1, 35	1, 26	26		30, 14, 7, 26
5	Area of moving object	19, 10, 32, 18	15, 17, 30, 26	10, 35, 2, 39	30, 26	26, 4	29, 30, 6, 13	29, 9	26, 28, 32, 3	2, 32	22, 33, 28, 1	17, 2, 18, 39	13, 1, 26, 24	15, 17, 13, 16	15, 13, 10, 1	15, 30	14, 1, 13	2, 36, 26, 18	14, 30, 28, 23	10, 26, 34, 2
6	Area of stationary object	17, 32	17, 7, 30	10, 14, 18, 39	30, 16	10, 35, 4, 18	2, 18, 40, 4	32, 35, 40, 4	26, 28, 32, 3	2, 29, 18, 36	27, 2, 39, 35	22, 1, 40	40, 16	16, 4	16	15, 16	1, 18, 36	2, 35, 30, 18	23	10, 15, 17, 7
7	Volume of moving object	35, 6, 13, 18	7, 15, 13, 16	36, 39, 34, 10	2, 22	2, 6, 34, 10	29, 30, 7	14, 1, 40, 11	25, 26, 28	25, 28, 2, 16	22, 21, 27, 35	17, 2, 40, 1	29, 1, 40	15, 13, 30, 12	10	15, 29	26, 1	29, 26, 4	35, 34, 16, 24	10, 6, 2, 34
8	Volume of stationary object	30, 6		10, 39, 35, 34		35, 16, 32 18	35, 3	2, 35, 16		35, 10, 25	34, 39, 19, 27	30, 18, 35, 4	35		1		1, 31	2, 17, 26		35, 37, 10, 2
9	Speed	19, 35, 38, 2	14, 20, 19, 35	10, 13, 28, 38	13, 26		10, 19, 29, 38	11, 35, 27, 28	28, 32, 1, 24	10, 28, 32, 25	1, 28, 35, 23	2, 24, 35, 21	35, 13, 8, 1	32, 28, 13, 12	34, 2, 28, 27	15, 10, 26	10, 28, 4, 34	3, 34, 27, 16	10, 18	
10	Force (Intensity)	19, 35, 18, 37	14, 15	8, 35, 40, 5		10, 37, 36	14, 29, 18, 36	3, 35, 13, 21	35, 10, 23, 24	28, 29, 37, 36	1, 35, 40, 18	13, 3, 36, 24	15, 37, 18, 1	1, 28, 3, 25	15, 1, 11	15, 17, 18, 20	26, 35, 10, 18	36, 37, 10, 19	2, 35	3, 28, 35, 37
11	Stress or pressure	10, 35, 14	2, 36, 25	10, 36, 3, 37		37, 36, 4	10, 14, 36	10, 13, 19, 35	6, 28, 25	3, 35	22, 2, 37	2, 33, 27, 18	1, 35, 16	11	2	35	19, 1, 35	2, 36, 37	35, 24	10, 14, 35, 37
12	Shape	4, 6, 2	14	35, 29, 3, 5		14, 10, 34, 17	36, 22	10, 40, 16	28, 32, 1	32, 30, 40	22, 1, 2, 35	35, 1	1, 32, 17, 28	32, 15, 26	2, 13, 1	1, 15, 29	16, 29, 1, 28	15, 13, 39	15, 1, 32	17, 26, 34, 10
13	Stability of the object's composition	32, 35, 27, 31	14, 2, 39, 6	2, 14, 30, 40		35, 27	15, 32, 35		13	18	35, 24, 30, 18	35, 40, 27, 39	35, 19	32, 35, 30	2, 35, 10, 16	35, 30, 34, 2	2, 35, 22, 26	35, 22, 39, 23	1, 8, 35	23, 35, 40, 3
14	Strength	10, 26, 35, 28	35	35, 28, 31, 40		29, 3, 28, 10	29, 10, 27	11, 3	3, 27, 16	3, 27	18, 35, 37, 1	15, 35, 22, 2	11, 3, 10, 32	32, 40, 25, 2	27, 11, 3	15, 3, 32	2, 13, 25, 28	27, 3, 15, 40	15	29, 35, 10, 14
15	Duration of action of moving object	19, 10, 35, 38		28, 27, 3, 18	10	20, 10, 28, 18	3, 35, 10, 40	11, 2, 13	3	3, 27, 16, 40	22, 15, 33, 28	21, 39, 16, 22	27, 1, 4	12, 27	29, 10, 27	1, 35, 13	10, 4, 29, 15	19, 29, 39, 35	6, 10	35, 17, 14, 19
16	Duration of action by stationary object	16		27, 16, 18, 38	10	28, 20, 10, 16	3, 35, 31	34, 27, 6, 40	10, 26, 24		17, 1, 40, 33	22	35, 10	1	1	2		25, 34, 6, 35	1	20, 10, 16, 38
17	Temperature	2, 14, 17, 25	21, 17, 35, 38	21, 36, 29, 31		35, 28, 21, 18	3, 17, 30, 39	19, 35, 3, 10	32, 19, 24	24	22, 33, 35, 2	22, 35, 2, 24	26, 27	26, 27	4, 10, 16	2, 18, 27	2, 17, 16	3, 27, 35, 31	26, 2, 19, 16	15, 28, 35
18	Illumination intensity	32	13, 16, 1, 6	13, 1	1, 6	19, 1, 26, 17	1, 19		11, 15, 32	3, 32	15, 19	35, 19, 32, 39	19, 35, 28, 26	28, 26, 19	15, 17, 13, 16	15, 1, 19	6, 32, 13	32, 15	2, 26, 10	2, 25, 16
19	Use of energy by moving object	6, 19, 37, 18	12, 22, 15, 24	35, 24, 18, 5		35, 38, 19, 18	34, 23, 16, 18	19, 21, 11, 27	3, 1, 32		1, 35, 6, 27	2, 35, 6	28, 26, 30	19, 35	1, 15, 17, 28	15, 17, 13, 16	2, 29, 27, 28	35, 38	32, 2	12, 28, 35
20	Use of energy by stationary object			28, 27, 18, 31			3, 35, 31	10, 36, 23			10, 2, 22, 37	19, 22, 18	1, 4					19, 35, 16, 25		1, 6
21	Power	+	10, 35, 38	28, 27, 18, 38	10, 19	35, 20, 10, 6	4, 34, 19	19, 24, 26, 31	32, 15, 2	32, 2	19, 22, 31, 2	2, 35, 18	26, 10, 34	26, 35, 10	35, 2, 10, 34	19, 17, 34	20, 19, 30, 34	19, 35, 16	28, 2, 17	28, 35, 34
22	Loss of Energy	3, 38	+	35, 27, 2, 37	19, 10	10, 18, 32, 7	7, 18, 25	11, 10, 35	32		21, 22, 35, 2	21, 35, 2, 22		35, 32, 1	2, 19		7, 23	35, 3, 15, 23	2	28, 10, 29, 35
23	Loss of substance	28, 27, 18, 38	35, 27, 2, 31	+		15, 18, 35, 10	6, 3, 10, 24	10, 29, 39, 35	16, 34, 31, 28	35, 10, 24, 31	33, 22, 30, 40	10, 1, 34, 29	15, 34, 33	32, 28, 2, 24	2, 35, 34, 27	15, 10, 2	35, 10, 28, 24	35, 18, 10, 13	35, 10, 18	28, 35, 10, 23
24	Loss of Information	10, 19	19, 10		+	24, 26, 28, 32	24, 28, 35	10, 28, 23			22, 10, 1	10, 21, 22	32	27, 22				35, 33	35	13, 23, 15
25	Loss of Time	35, 20, 10, 6	10, 5, 18, 32	35, 18, 10, 39	24, 26, 28, 32	+	35, 38, 18, 16	10, 30, 4	24, 34, 28, 32	24, 26, 28, 18	35, 18, 34	35, 22, 18, 39	35, 28, 34, 4	4, 28, 10, 34	32, 1, 10	35, 28	6, 29	18, 28, 32, 10	24, 28, 35, 30	
26	Quantity of substance/the matter	35	7, 18, 25	6, 3, 10, 24	24, 28, 35	35, 38, 18, 16	+	18, 3, 28, 40	13, 2, 28	33, 30	35, 33, 29, 31	3, 35, 40, 39	29, 1, 35, 27	35, 29, 25, 10	2, 32, 10, 25	15, 3, 29	3, 13, 27, 10	3, 27, 29, 18	8, 35	13, 29, 3, 27
27	Reliability	21, 11, 26, 31	10, 11, 35	10, 35, 29, 39	10, 28	10, 30, 4	21, 28, 40, 3	+	32, 3, 11, 23	11, 32, 1	27, 35, 2, 40	35, 2, 40, 26		27, 17, 40	1, 11	13, 35, 8, 24	13, 35, 1	27, 40, 28	11, 13, 27	1, 35, 29, 38
28	Measurement accuracy	3, 6, 32	26, 32, 27	10, 16, 31, 28		24, 34, 28, 32	2, 6, 32	5, 11, 1, 23	+		28, 24, 22, 26	3, 33, 39, 10	6, 35, 25, 18	1, 13, 17, 34	1, 32, 13, 11	13, 35, 2	27, 35, 10, 34	26, 24, 32, 28	28, 2, 10, 34	10, 34, 28, 32
29	Manufacturing precision	32, 2	13, 32, 2	35, 31, 10, 24		32, 26, 28, 18	32, 30	11, 32, 1		+	26, 28, 10, 36	4, 17, 34, 26		1, 32, 35, 23	25, 10		26, 2, 18		26, 28, 18, 23	10, 18, 32, 39
30	Object-affected harmful factors	19, 22, 31, 2	21, 22, 35, 2	33, 22, 19, 40	22, 10, 2	35, 18, 34	35, 33, 29, 31	27, 24, 2, 40	28, 33, 23, 26	26, 28, 10, 18	+		24, 35, 2	2, 25, 28, 39	35, 10, 2	35, 11, 22, 31	22, 19, 29, 40	22, 19, 29, 40	33, 3, 34	22, 35, 13, 24
31	Object-generated harmful factors	2, 35, 18	21, 35, 2, 22	10, 1, 34	10, 21, 29	1, 22	3, 24, 39, 1	24, 2, 40, 39	3, 33, 26	4, 17, 34, 26		+					19, 1, 31	2, 21, 27, 1	2	22, 35, 18, 39
32	Ease of manufacture	27, 1, 12, 24	19, 35	15, 34, 33	32, 24, 18, 16	35, 28, 34, 4	35, 23, 1, 24		1, 35, 12, 18		24, 2		+	2, 5, 13, 16	35, 1, 11, 9	2, 13, 15	27, 26, 1	6, 28, 11, 1	8, 28, 1	35, 1, 10, 28
33	Ease of operation	35, 34, 2, 10	2, 19, 13	28, 32, 2, 24	4, 10, 27, 22	4, 28, 10, 34	12, 35	17, 27, 8, 40	25, 13, 2, 34	1, 32, 35, 23	2, 25, 28, 39		2, 5, 12	+	12, 26, 1, 32	15, 34, 1, 16	32, 26, 12, 17		1, 34, 12, 3	15, 1, 28
34	Ease of repair	15, 10, 32, 2	15, 1, 32, 19	2, 35, 34, 27		32, 1, 10, 25	2, 28, 10, 25	11, 10, 1, 16	10, 2, 13	25, 10	35, 10, 2, 16		1, 35, 11, 10	1, 12, 26, 15	+	7, 1, 4, 16	35, 1, 13, 11		34, 35, 7, 13	1, 32, 10
35	Adaptability or versatility	19, 1, 29	18, 15, 1	15, 10, 2, 13		35, 28	3, 35, 15	35, 13, 8, 24	35, 5, 1, 10		35, 11, 32, 31		1, 13, 31	15, 34, 1, 16	1, 16, 7, 4	+	15, 29, 37, 28	1	27, 34, 35	35, 28, 6, 37
36	Device complexity	20, 19, 30, 34	10, 35, 13, 2	35, 10, 28, 29		6, 29	13, 3, 27, 10	13, 35, 1	2, 26, 10, 34	26, 24, 32	22, 19, 29, 40	19, 1	27, 26, 1, 13	27, 9, 26, 24	1, 13	29, 15, 28, 37	+	15, 10, 37, 28	15, 1, 24	12, 17, 28
37	Difficulty of detecting and measuring	18, 1, 16, 10	35, 3, 15, 19	1, 18, 10, 24	35, 33, 27, 22	18, 28, 32, 9	3, 27, 29, 18	27, 40, 28, 8	26, 24, 32, 28		22, 19, 29, 28	2, 21	5, 28, 11, 29	2, 5	12, 26	1, 15	15, 10, 37, 28	+	34, 21	35, 18
38	Extent of automation	28, 2, 27	23, 28	35, 10, 18, 5	35, 33	24, 28, 35, 30	35, 13	11, 27, 32	28, 26, 10, 34	28, 26, 18, 23	2, 33	2	1, 26, 13	1, 12, 34, 3	1, 35, 13	27, 4, 1, 35	15, 24, 10	34, 27, 25	+	5, 12, 35, 26
39	Productivity	35, 20, 10	28, 10, 29, 35	28, 10, 35, 23	13, 15, 23		35, 38	1, 35, 10, 38	1, 10, 34, 28	18, 10, 32, 1	22, 35, 13, 24	35, 22, 18, 39	35, 28, 2, 24	1, 28, 7, 10	1, 32, 10, 25	1, 35, 28, 37	12, 17, 28, 24	35, 18, 27, 2	5, 12, 35, 26	+

Appendix F

Case Studies

We introduce additional real solved use cases found in the literature by SAM-IDM. We highlight the output of SAM-IDM on these problems. SAM-IDM mines all similar problems and their patents using a fixed similarity threshold. These similar problems and corresponding solutions are investigated by experts to check their potential in creating an inventive solution.

F.1 Case Studies

Target Problem 1:"*It is thus possible to prevent a decrease in the conductivity of the conductive film 421 due to oxidation.*" (**US9536627, Physics**)

- **Similar Problem (Similarity Value: 0.83)**: "*materials function as a heat sink due to the latent heat of vaporization.*" (**US9537344, Electricity**)
- **Corresponding Solution**: "*materials transitioning from a liquid to a gas or vapor, or from a solid to a gas or vapor, could also be used as a heat sink, so long as the temperature associated with the phase change was in the desired range.*"
- **Latent Inventive Solution Reference**: From the corresponding solution according to the similar problem, we notice that (Eranna et al., 2004) mentioned oxide materials for development of integrated gas sensors are related to the conductivity.

- **Similar Problem (Similarity Value: 0.83)**: "*It may lead to a decrease in the yield in subsequent production steps.*" (**US9538663, Electricity**)
- **Corresponding Solution**: "*connecting a sidewall of the wiring board to a sidewall in an opening of the metal frame by subjecting the metal frame to plastic deformation.*"

- **Latent Inventive Solution Reference**: With the hints of the corresponding solution, we notice that raising the mix ratio of the powdered metal is, in the conductive filler kneading method, able to obtain the desired surface resistance (Kawaguchi et al., 2006). This might be useful to solve the target problem.

- **Similar Problem (Similarity Value: 0.82)**: "*it is difficult to form a roughened surface.*" (**US9538642, Electricity**)
- **Corresponding Solution**: "*Therefore, in the present embodiment, the above-described thin resin layer 1012 is formed through a method that uses resin contraction caused by heating.*"
- **Latent Inventive Solution Reference**: Based on the corresponding solution, we notice that a solid polymer electrolyte film from hydrogen bonding layer (DeLongchamp and Hammond, 2004) and a percolating-assisted resin film infusion method (Wang et al., 2018) have been proposed to solve the related problem.

Target Problem 2: "*Segmentation and analysis can add computational complexity and overhead.*" (**US9534958, Physics**)

- **Similar Problem (Similarity Value: 0.86)**: "*which can incur high computational complexity.*" (**US9538483, Electricity**)
- **Corresponding Solution**: "*Some embodiments described herein provide a novel approach to the weighted sum-rate maximization in the MIMO interference network, and apply a novel and efficient algorithm with guaranteed monotonic convergence as well as an elegant way to establish rate duality between an interference network and its reciprocal.*"
- **Latent Inventive Solution Reference**: A Low-Complexity Intrusion Detection Algorithm is proposed by (Sajana, 2011) to solve the similar target issue.

Target Problem 3: "*many of the devices also require the use of a pressure washer.*" (**US9533320, Performing Operations**)

- **Similar Problem (Similarity Value: 0.87)**: "*a hole of a microphone is blocked*" (**US9537985, Electricity**)

- **Corresponding Solution**: "*The hall structure 100 according to the present invention can better reduce blocking of a hole of a microphone by combining the hole of the microphone and a hole of a speakerphone or receiver.*"
- **Latent Inventive Solution Reference**: The solution from US9537985 that combining two different holes is close to the target problem's solution that the device consists of a connector and conduits in US9533320.

Target Problem 4: "*Patient-specific implants are expensive to engineer and manufacture. Moreover, the plate can cause bone necrosis if the fit is too snug.*" (**US9532825, Human Necessities**)

- **Similar Problem (Similarity Value: 0.81)**: "*reservoirs are expensive and difficult to manufacture.* " (**US9534198, Chemistry**)
- **Corresponding Solution**: "*One aspect of the present invention is to provide an EC fluid cycling unit that enables fluid level control without the use of expensive ultrasonics or load cells.*"
- **Latent Inventive Solution Reference**: The bioprinted constructs for treatment of degenerated intervertebral disc (Costa et al., 2019) that is similar with the fluid cycling unit has been proposed to solve the similar issue with the target problem.

Appendix G

Multiple Criteria Decision Analysis

Multiple Criteria Decision Analysis (MCDA) is a sub-discipline of the operation research that explicitly evaluates multiple criteria in the decision making to aid in understanding the inherent trade-off (Greene et al., 2011; Masud and Ravindran, 2008; Busemeyer and Diederich, 2002). It provides decision makers with a set of tools that can help stakeholders make consistent and transparent decisions.The MCDA approach draws on theories that consider both the qualitative and quantitative aspects of decision making. This is achieved through an integrated approach that includes problem structuring and model building (Regier and Peacock, 2017).

Furthermore, decision aiding can be defined as follows: decision aiding is the activity of people using models (not necessarily completely formalized ones) to obtain elements of responses to the questions asked by a stakeholder in a decision process. These elements work towards clarifying the decision and usually towards recommending, or simply favoring, a behavior that will increase the consistency between the evolution of the process and this stakeholder's objectives and value system (Greco, Figueira, and Ehrgott, 2016). In this thesis, we use it to build an inventive solutions ranking model. We attempt to address an intractable problem of ranking latent solutions from different domains patent documents according to their inventiveness via MCDA approaches.

A comprehensive MCDA draws knowledge from several different fields, including mathematics, economics, information technology, software engineering, and other information systems. Typical executing steps of MCDA are as follows:

- **Define Context** The prior item for performing multiple criteria decision analysis is to define context. For instance, current situation, involved elements, and stakeholders in the decision-making process suppose to be presented.

- **Identify Available Options** MCDA compares several different options against each other. Whether pre-determined or not yet developed, all options are able to be changed.

- **Choose Suitable Criteria** Consequences play a significant role in MCDA since each option may contribute to different consequences, such as a lower quality of the product following a production line modification. Several criteria should be therefore established. Criteria represent clearly defined standards by which different options can be measured and compared, and also express the different levels of value created by each option. When reading a patent, engineers prefer to achieve a patent with fewer words but more inventiveness. However, inventiveness is difficult to be measured. In such the case, inventiveness needs to be subdivided into quantifiable criteria such as number of citations, number of inventors, etc.

- **Measure Each of the Criteria** Simply choosing the right criteria is not sufficient to combine and analyze different scales of choice. One unit of preference is not necessarily the same as another unit of preference. For instance, buyers prefer the comfortable feature of the car compared to speed even if they are all significant. The weights of the different criteria thus illustrate not only the differences between criteria but also the relevance of such differences.

- **Calculate Different Values by Averaging Weights** In order to let different attributes to be within the same range, we normalize each value. The general preference score is the weighted average of all criteria.

Bibliography

Abrams, David S, Ufuk Akcigit, and Jillian Grennan (2013). *Patent value and citations: Creative destruction or strategic disruption?* Tech. rep. National Bureau of Economic Research.

Altshuller, Genrich, Lev Shulyak, and Steven Rodman (40). "principles: TRIZ keys to technical innovation. Technical Innovation Center". In: *Inc., Worcester, MA*.

Altshuller, Genrikh Saulovich (1984). *Creativity as an exact science: the theory of the solution of inventive problems*. Gordon and Breach.

— (1999). *The innovation algorithm: TRIZ, systematic innovation and technical creativity*. Technical innovation center, Inc.

Arora, Ashish, Andrea Fosfuri, and Alfonso Gambardella (2004). *Markets for technology: The economics of innovation and corporate strategy*. MIT press.

Bae, J-HJ (2005). "Creative problem solving with TRIZ: an example". In: *Proceedings. The 9th Russian-Korean International Symposium on Science and Technology, 2005. KORUS 2005.* IEEE, pp. 1082–1083.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio (2014). "Neural machine translation by jointly learning to align and translate". In: *arXiv preprint arXiv:1409.0473*.

Benedetti, Fabio et al. (2019). "Computing inter-document similarity with context semantic analysis". In: *Information Systems* 80, pp. 136–147.

Bengio, Yoshua, Aaron Courville, and Pascal Vincent (2013). "Representation learning: A review and new perspectives". In: *IEEE transactions on pattern analysis and machine intelligence* 35.8, pp. 1798–1828.

Bengio, Yoshua et al. (2003). "A neural probabilistic language model". In: *The journal of machine learning research* 3, pp. 1137–1155.

Berdyugina, Daria and Denis Cavallucci (2021). "Automatic Extraction of Potentially Contradictory Parameters from Specific Field Patent Texts". In: *International TRIZ Future Conference*. Springer, pp. 150–161.

Block, Joern et al. (2013). "Economic and technological importance of innovations in large family and founder firms: An analysis of patent data". In: *Family Business Review* 26.2, pp. 180–199.

Bojanowski, Piotr et al. (2017). “Enriching word vectors with subword information”. In: *Transactions of the Association for Computational Linguistics* 5, pp. 135–146.

Brown, Peter F et al. (1992). “Class-based n-gram models of natural language”. In: *Computational linguistics* 18.4, pp. 467–480.

Brown, Tom B et al. (2020). “Language models are few-shot learners”. In: *arXiv preprint arXiv:2005.14165*.

Bultey, Alexis, Francois De Bertrand De Beuvron, and Francois Rousselot (2007). “A substance-field ontology to support the TRIZ thinking approach”. In: *International Journal of Computer Applications in Technology* 30.1-2, pp. 113–124.

Busemeyer, Jerome R and Adele Diederich (2002). “Survey of decision field theory”. In: *Mathematical social sciences* 43.3, pp. 345–370.

Cavallucci (2009). “From TRIZ to inventive design method (IDM): towards a formalization of Inventive Practices in R&D Departments”. In: *Innovation* 18.2.

Cavallucci, Denis and Nikolai Khomenko (2007). “From TRIZ to OTSM-TRIZ: addressing complexity challenges in inventive design”. In: *International Journal of Product Development* 4.1-2, pp. 4–21.

Cavallucci, Denis, François Rousselot, and Cecilia Zanni (2009). “Linking contradictions and laws of engineering system evolution within the TRIZ framework”. In: *Creativity and Innovation Management* 18.2, pp. 71–80.

— (2010). “Initial situation analysis through problem graph”. In: *CIRP Journal of Manufacturing Science and Technology* 2.4, pp. 310–317.

Cavallucci, Denis, Francois Rousselot, and Cecilia Zanni (2011). “Using patents to populate an inventive design ontology”. In: *Procedia Engineering* 9, pp. 52–62.

Chen, Huimin et al. (2021). “From Symbols to Embeddings: A Tale of Two Representations in Computational Social Science”. In: *Journal of Social Computing* 2.2, pp. 103–156.

Chowdhury, Gobinda G (2010). *Introduction to modern information retrieval*. Facet publishing.

Collobert, Ronan and Jason Weston (2008). “A unified architecture for natural language processing: Deep neural networks with multitask learning”. In: *Proceedings of the 25th international conference on Machine learning*, pp. 160–167.

Collobert, Ronan et al. (2011). “Natural language processing (almost) from scratch”. In: *Journal of Machine Learning Research* 12.Aug, pp. 2493–2537.

Costa, João Bebiano et al. (2019). "Engineering patient-specific bioprinted constructs for treatment of degenerated intervertebral disc". In: *Materials Today Communications* 19, pp. 506–512.

Cronier, P et al. (2010). "The concept of locking plates". In: *Orthopaedics & Traumatology: Surgery & Research* 96.4, S17–S36.

Dai, Zihang et al. (2019). "Transformer-xl: Attentive language models beyond a fixed-length context". In: *arXiv preprint arXiv:1901.02860*.

Deerwester, Scott et al. (1990). "Indexing by latent semantic analysis". In: *Journal of the American Society for Information Science* 41.6, pp. 391–407.

DeLongchamp, Dean M and Paula T Hammond (2004). "Highly ion conductive poly (ethylene oxide)-based solid polymer electrolytes from hydrogen bonding layer-by-layer assembly". In: *Langmuir* 20.13, pp. 5403–5411.

Deng, Li and Dong Yu (2013). "Deep learning for signal and information processing". In: *Microsoft research monograph*.

Dernis, Hélène and Mosahid Khan (2004). "Triadic patent families methodology". In.

Devlin, Jacob et al. (2018). "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *arXiv preprint arXiv:1810.04805*.

Dimitrakis, Eleftherios, Konstantinos Sgontzos, and Yannis Tzitzikas (2020). "A survey on question answering systems over linked data and documents". In: *Journal of Intelligent Information Systems* 55.2, pp. 233–259.

Ding, Zhikun et al. (2017). "A new TRIZ-based patent knowledge management system for construction technology innovation". In: *Journal of Engineering, Design and Technology*.

Domb, Ellen (1997). "Contradictions: air bag applications". In: *TRIZ Journal* 2.7.

— (1998). "QFD and Tips/triz". In: *The TRIZ Journal*.

Dubois, Sébastien et al. (2005). "A formal model for the representation of problems based on TRIZ". In: *International Conference on Engineering Design, ICED 05*, NA.

Edunov, Sergey, Alexei Baevski, and Michael Auli (2019). "Pre-trained language model representations for language generation". In: *arXiv preprint arXiv:1903.09722*.

Elman, Jeffrey L (1990). "Finding structure in time". In: *Cognitive science* 14.2, pp. 179–211.

Ensan, Faezeh and Weichang Du (2019). "Ad hoc retrieval via entity linking and semantic similarity". In: *Knowledge and Information Systems* 58.3, pp. 551–583.

Eranna, G et al. (2004). "Oxide materials for development of integrated gas sensors—a comprehensive review". In: *Critical Reviews in Solid State and Materials Sciences* 29.3-4, pp. 111–188.

Fernando, Tharindu et al. (2018). "Tree memory networks for modelling long-term temporal dependencies". In: *Neurocomputing* 304, pp. 64–81.

Gasmi, Houssem, Jannik Laval, and Abdelaziz Bouras (2019). "Cold-start cybersecurity ontology population using information extraction with LSTM". In: *2019 International Conference on Cyber Security for Emerging Technologies (CSET)*. IEEE, pp. 1–6.

Gehring, Jonas et al. (2017). "Convolutional sequence to sequence learning". In: *International Conference on Machine Learning*. PMLR, pp. 1243–1252.

Girodon, Julien et al. (2015). "An organizational approach to designing an intelligent knowledge-based system: Application to the decision-making process in design projects". In: *Advanced Engineering Informatics* 29.3, pp. 696–713.

Goldberg, Yoav (2017). "Neural network methods for natural language processing". In: *Synthesis Lectures on Human Language Technologies* 10.1, pp. 1–309.

Greco, Salvatore, Jose Figueira, and Matthias Ehrgott (2016). *Multiple criteria decision analysis*. Vol. 37. Springer.

Greene, Randal et al. (2011). "GIS-based multiple-criteria decision analysis". In: *Geography compass* 5.6, pp. 412–432.

Grefenstette, Edward (2013). "Towards a formal distributional semantics: Simulating logical calculi with tensors". In: *arXiv preprint arXiv:1304.5823*.

Guan, Jian Cheng and Xia Gao (2009). "Exploring the h-index at patent level". In: *Journal of the American Society for Information Science and Technology* 60.1, pp. 35–40.

Guarino, Guillaume, Ahmed Samet, and Denis Cavallucci (2020). "Summarization as a Denoising Extraction Tool". In: *International TRIZ Future Conference*. Springer, pp. 77–87.

Guarino, Guillaume et al. (2020). "SummaTRIZ: Summarization Networks for Mining Patent Contradiction". In: *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, pp. 979–986.

Guehika, Adehi (2019). "TRIZ, a Systematic Approach to Create Quantum Activation Function for Deep Learning's Hidden Layers, in Order to Make AI Explainable with Quantum Computer". In: *International TRIZ Future Conference*. Springer, pp. 371–387.

Guellec, Dominique and Bruno Van Pottelsberghe de La Potterie (2007). *The economics of the European patent system: IP policy for innovation and competition*. Oxford University Press on Demand.

Hall, Bronwyn H, Adam Jaffe, and Manuel Trajtenberg (2005). "Market value and patent citations". In: *RAND Journal of economics*, pp. 16–38.

Hao, Jia et al. (2019). "An evolutionary computation based method for creative design inspiration generation". In: *Journal of Intelligent Manufacturing* 30.4, pp. 1673–1691.

Harhoff, Dietmar, Frederic M Scherer, and Katrin Vopel (2003). "Citations, family size, opposition and the value of patent rights". In: *Research policy* 32.8, pp. 1343–1363.

Harris, Zellig S (1954). "Distributional structure". In: *Word* 10.2-3, pp. 146–162.

Hinton, Geoffrey E et al. (1986). "Learning distributed representations of concepts". In: *Proceedings of the eighth annual conference of the cognitive science society*. Vol. 1. Amherst, MA, p. 12.

Hoch, Nino and Stelian Brad (2020). "Managing AI Technologies in Earthwork Construction: A TRIZ-Based Innovation Approach". In: *International TRIZ Future Conference*. Springer, pp. 3–14.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). "Long short-term memory". In: *Neural Computation* 9.8, pp. 1735–1780.

Houssin, Remy et al. (2015). "TRIZ theory and case based reasoning: synergies and oppositions". In: *International Journal on Interactive Design and Manufacturing (IJIDeM)* 9.3, pp. 177–183.

Huang, Eric H et al. (2012). "Improving word representations via global context and multiple word prototypes". In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 873–882.

Huenteler, Joern et al. (2016). "Technology life-cycles in the energy sector—Technological characteristics and the role of deployment for innovation". In: *Technological Forecasting and Social Change* 104, pp. 102–121.

Jardim-Goncalves, Ricardo et al. (2011). "Knowledge framework for intelligent manufacturing systems". In: *Journal of Intelligent Manufacturing* 22.5, pp. 725–735.

Jiao, Xiaoqi et al. (2019). "Tinybert: Distilling bert for natural language understanding". In: *arXiv preprint arXiv:1909.10351*.

Jones, Karen Sparck (1972). "A statistical interpretation of term specificity and its application in retrieval". In: *Journal of documentation*.

Joulin, Armand et al. (2016). "Fasttext. zip: Compressing text classification models". In: *arXiv preprint arXiv:1612.03651*.

Kalchbrenner, Nal, Edward Grefenstette, and Phil Blunsom (June 2014). "A Convolutional Neural Network for Modelling Sentences". In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Baltimore, Maryland: Association for Computational Linguistics, pp. 655–665. DOI: 10.3115/v1/P14-1062. URL: https://www.aclweb.org/anthology/P14-1062.

Kawaguchi, Toshiyuki et al. (Apr. 2006). *Shield box and shield method*. US Patent App. 10/536,870.

Kenter, Tom and Maarten De Rijke (2015). "Short text similarity with word embeddings". In: *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. ACM, pp. 1411–1420.

Khanna, Rajat, Isin Guler, and Atul Nerkar (2016). "Fail often, fail big, and fail fast? Learning from small failures and R&D performance in the pharmaceutical industry". In: *Academy of Management Journal* 59.2, pp. 436–459.

Kim, Yoon (Oct. 2014). "Convolutional Neural Networks for Sentence Classification". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, pp. 1746–1751. DOI: 10.3115/v1/D14-1181. URL: https://www.aclweb.org/anthology/D14-1181.

Kusiak, Andrew (2016). "Put innovation science at the heart of discovery". In: *Nature* 530.7590, pp. 255–255.

Kusner, Matt et al. (2015). "From word embeddings to document distances". In: *International Conference on Machine Learning*, pp. 957–966.

Lanjouw, Jean O and Mark Schankerman (2004). "Patent quality and research productivity: Measuring innovation with multiple indicators". In: *The Economic Journal* 114.495, pp. 441–465.

Lee, Joon Ho, Myoung Ho Kim, and Yoon Joon Lee (1993). "Information retrieval based on conceptual distance in IS-A hierarchies". In: *Journal of documentation*.

Lerner, Joshua (1994). "The importance of patent scope: an empirical analysis". In: *The RAND Journal of Economics*, pp. 319–333.

Li, Zhen et al. (2012). "A framework for automatic TRIZ level of invention estimation of patents using natural language processing, knowledge-transfer and patent citation metrics". In: *Computer-Aided Design* 44.10, pp. 987–1010.

Litvin, Simon et al. (2007). "TRIZ body of knowledge". In: *TRIZ developers summit, Russia. Accessed December* 18, p. 2012.

Liu, Pengfei, Xipeng Qiu, and Xuanjing Huang (2016). "Recurrent neural network for text classification with multi-task learning". In: *arXiv preprint arXiv:1605.05101*.

Liu, Zhiyuan, Yankai Lin, and Maosong Sun (2020). *Representation learning for natural language processing*. Springer Nature.

MacAvaney, Sean et al. (2019). "CEDR: Contextualized embeddings for document ranking". In: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1101–1104.

Majumder, P, M Mitra, and BB Chaudhuri (2002). "N-gram: a language independent approach to IR and NLP". In: *International conference on universal knowledge and language*.

Mann, Darrell (2002). "Axiomatic design and TRIZ: compatibilities and contradictions". In: *2nd International Conference on Axiomatic Design*.

Marcheggiani, Diego, Joost Bastings, and Ivan Titov (2018). "Exploiting semantics in neural machine translation with graph convolutional networks". In: *arXiv preprint arXiv:1804.08313*.

Martínez, Catalina (2011). "Patent families: When do different definitions really matter?" In: *Scientometrics* 86.1, pp. 39–63.

Masud, Abu SM and A Ravi Ravindran (2008). "Multiple criteria decision making". In: CRC Press, An imprint of the Taylor and Francis Group.

Mikolov, Tomas, Quoc V Le, and Ilya Sutskever (2013). "Exploiting similarities among languages for machine translation". In: *arXiv preprint arXiv:1309.4168*.

Mikolov, Tomáš et al. (2010). "Recurrent neural network based language model". In: *Eleventh Annual Conference of the International Speech Communication Association*.

Mikolov, Tomáš et al. (2012). "Statistical language models based on neural networks". In: *Presentation at Google, Mountain View, 2nd April* 80, p. 26.

Mikolov, Tomas et al. (2013a). "Distributed representations of words and phrases and their compositionality". In: *arXiv preprint arXiv:1310.4546*.

Mikolov, Tomas et al. (2013b). "Efficient estimation of word representations in vector space". In: *arXiv preprint arXiv:1301.3781*.

Miller, Douglas J, Michael J Fern, and Laura B Cardinal (2007). "The use of knowledge for technological innovation within diversified firms". In: *Academy of Management journal* 50.2, pp. 307–325.

Mitchell, Jeff and Mirella Lapata (2010). “Composition in distributional models of semantics”. In: *Cognitive science* 34.8, pp. 1388–1429.

Mnih, Andriy and Geoffrey E Hinton (2009). “A scalable hierarchical distributed language model”. In: *Advances in Neural Information Processing Systems*, pp. 1081–1088.

Mueller, Jonas and Aditya Thyagarajan (2016). “Siamese recurrent architectures for learning sentence similarity”. In: *Thirtieth AAAI Conference on Artificial Intelligence*.

Nagaoka, Sadao, Kazuyuki Motohashi, and Akira Goto (2010). “Patent statistics as an innovation indicator”. In: *Handbook of the Economics of Innovation*. Vol. 2. Elsevier, pp. 1083–1127.

Negny, Stéphane et al. (2012). “Toward an eco-innovative method based on a better use of resources: application to chemical process preliminary design”. In: *Journal of Cleaner Production* 32, pp. 101–113.

Ni, Xin, Ahmed Samet, and Denis Cavallucci (2019). “An Approach Merging the IDM-Related Knowledge”. In: *International TRIZ Future Conference*. Springer, pp. 147–158.

— (2021). “Similarity-based approach for inventive design solutions assistance”. In: *Journal of Intelligent Manufacturing*, pp. 1–18.

Noailly, Joëlle and Victoria Shestalova (2017). “Knowledge spillovers from renewable energy technologies: Lessons from patent citations”. In: *Environmental Innovation and Societal Transitions* 22, pp. 1–14.

Otter, Daniel W, Julian R Medina, and Jugal K Kalita (2020). “A survey of the usages of deep learning for natural language processing”. In: *IEEE Transactions on Neural Networks and Learning Systems*.

Park, Hyunseok, Janghyeok Yoon, and Kwangsoo Kim (2013). “Identification and evaluation of corporations for merger and acquisition strategies using patent information and text mining”. In: *Scientometrics* 97.3, pp. 883–909.

Pawar, Atish and Vijay Mago (2018). “Calculating the similarity between words and sentences using a lexical database and corpus statistics”. In: *arXiv preprint arXiv:1802.05667*.

Pennington, Jeffrey, Richard Socher, and Christopher Manning (2014). “Glove: Global vectors for word representation”. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543.

Perren, Stephan M (2002). “Evolution of the internal fixation of long bone fractures: the scientific basis of biological internal fixation: choosing a new balance between stability and biology". In: *The Journal of Bone and Joint Surgery. British volume* 84.8, pp. 1093–1110.

Peters, Matthew E et al. (2018). "Deep contextualized word representations". In: *arXiv preprint arXiv:1802.05365*.

Petruzzelli, Antonio Messeni, Daniele Rotolo, and Vito Albino (2015). "Determinants of patent citations in biotechnology: An analysis of patent influence across the industrial and organizational boundaries". In: *Technological Forecasting and Social Change* 91, pp. 208–221.

Qiu, Xipeng et al. (2020). "Pre-trained models for natural language processing: A survey". In: *Science China Technological Sciences*, pp. 1–26.

Radford, Alec et al. (2019). "Language models are unsupervised multitask learners". In: *OpenAI blog* 1.8, p. 9.

Rahim, Zulhasni Abdul et al. (2018). "The application of computational thinking and TRIZ methodology in patent innovation analytics". In: *International Conference of Reliable Information and Communication Technology*. Springer, pp. 793–802.

Rajaraman, Anand and Jeffrey David Ullman (2011). *Mining of massive datasets*. Cambridge University Press.

Rajpurkar, Pranav et al. (2016). "Squad: 100,000+ questions for machine comprehension of text". In: *arXiv preprint arXiv:1606.05250*.

Ravichandran, Deepak and Eduard Hovy (2002). "Learning surface text patterns for a question answering system". In: *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, pp. 41–47.

Regier, Dean A and Stuart Peacock (2017). "Theoretical foundations of MCDA". In: *Multi-criteria decision analysis to support healthcare decisions*. Springer, pp. 9–28.

Reimers, Nils and Iryna Gurevych (2019). "Sentence-bert: Sentence embeddings using siamese bert-networks". In: *arXiv preprint arXiv:1908.10084*.

Renjith, Sarath C, Kijung Park, and Gül E Okudan Kremer (2020). "A design framework for additive manufacturing: Integration of additive manufacturing capabilities in the early design process". In: *International Journal of Precision Engineering and Manufacturing* 21.2, pp. 329–345.

Rousselot, François, Cecilia Zanni-Merk, and Denis Cavallucci (2012). "Towards a formal definition of contradiction in inventive design". In: *Computers in Industry* 63.3, pp. 231–242.

Sajana, Abu R (2011). "A Low-Complexity Intrusion Detection Algorithm For Surveillance Using PIR Sensors In A Wireless Sensor Network". PhD thesis.

Savransky, Semyon D (2000). *Engineering of creativity: Introduction to TRIZ methodology of inventive problem solving*. CRC press.

Sheu, D Daniel, Chia-Hung Chen, and Pang-Yen Yu (2012). "Invention principles and contradiction matrix for semiconductor manufacturing industry: chemical mechanical polishing". In: *Journal of Intelligent Manufacturing* 23.5, pp. 1637–1648.

Shirwaiker, Rohan A and Gül E Okudan (2008). "Triz and axiomatic design: a review of case-studies and a proposed synergistic use". In: *Journal of Intelligent Manufacturing* 19.1, pp. 33–47.

Shulyak, Lev (1998). "Introduction to TRIZ". In: *Altshuller, G.(Ed.)* 40, pp. 15–22.

Silverberg, Gerald and Bart Verspagen (2007). "The size distribution of innovations revisited: an application of extreme value statistics to citation and value measures of patent significance". In: *Journal of Econometrics* 139.2, pp. 318–339.

Slocum, Michael S and Amir HM Kermani (2006). "Case study: integrating TRIZ into six sigma". In: *The TRIZ Journal*.

Smirnov, Alexander et al. (2013). "Knowledge management for complex product development". In: *IFIP International Conference on Product Lifecycle Management*. Springer, pp. 110–119.

Socher, Richard et al. (2011). "Semi-supervised recursive autoencoders for predicting sentiment distributions". In: *Proceedings of the 2011 conference on empirical methods in natural language processing*, pp. 151–161.

Socher, Richard et al. (2013). "Recursive deep models for semantic compositionality over a sentiment treebank". In: *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642.

Souili, Achille and Denis Cavallucci (2017). "Automated Extraction of Knowledge Useful to Populate Inventive Design Ontology from Patents". In: *TRIZ–The Theory of Inventive Problem Solving*. Springer, pp. 43–62.

Souili, Achille, Denis Cavallucci, and François Rousselot (2015a). "A lexico-syntactic pattern matching method to extract IDM-TRIZ knowledge from on-line patent databases". In: *Procedia Engineering* 131, pp. 418–425.

— (2015b). "Natural language processing (NLP)–a solution for knowledge extraction from patent unstructured data". In: *Procedia engineering* 131, pp. 635–643.

Soumya George, K and Shibily Joseph (2014). "Text classification by augmenting bag of words (BOW) representation with co-occurrence feature". In: *IOSR J. Comput. Eng* 16.1, pp. 34–38.

Squicciarini, Mariagrazia, Hélène Dernis, and Chiara Criscuolo (2013). "Measuring patent quality". In.

Sutskever, Ilya, Oriol Vinyals, and Quoc V Le (2014). "Sequence to sequence learning with neural networks". In: *Advances in neural information processing systems*, pp. 3104–3112.

Tai, Kai Sheng, Richard Socher, and Christopher D Manning (2015). "Improved semantic representations from tree-structured long short-term memory networks". In: *arXiv preprint arXiv:1503.00075*.

Toutanova, Kristina et al. (2015). "Representing text for joint embedding of text and knowledge bases". In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1499–1509.

Turian, Joseph, Lev Ratinov, and Yoshua Bengio (2010). "Word representations: a simple and general method for semi-supervised learning". In: *Proceedings of the 48th annual meeting of the association for computational linguistics*, pp. 384–394.

Turney, Peter D and Patrick Pantel (2010). "From frequency to meaning: Vector space models of semantics". In: *Journal of artificial intelligence research* 37, pp. 141–188.

Ungvari, Steven (1999). "TRIZ within the Context of the Kano Model or Adding the Third Dimension to Quality". In: *Triz Journal Retrieved* 4, p. 2010.

Van Zeebroeck, Nicolas (2011). "The puzzle of patent value indicators". In: *Economics of innovation and new technology* 20.1, pp. 33–62.

Vaswani, Ashish et al. (2017). "Attention is all you need". In: *arXiv preprint arXiv:1706.03762*.

Wang, Ben et al. (2018). "Fabrication of an enriched graphene surface protection of carbon fiber/epoxy composites for lightning strike via a percolating-assisted resin film infusion method". In: *Composites Science and Technology* 158, pp. 51–60.

Wang, Wenhui et al. (2017). "Gated self-matching networks for reading comprehension and question answering". In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 189–198.

Wang, Zhiguo, Haitao Mi, and Abraham Ittycheriah (2016). "Sentence similarity learning by lexical decomposition and composition". In: *arXiv preprint arXiv:1602.07019*.

Whiteside, A et al. (2009). "Developing a current capability design for manufacture framework in the aerospace industry". In: *Proceedings of the 19th CIRP Design Conference–Competitive Design*. Cranfield University Press.

Widianta, MMD et al. (2018). "Comparison of multi-criteria decision support methods (AHP, TOPSIS, SAW & PROMENTHEE) for employee placement". In: *Journal of Physics: Conference Series*. Vol. 953. 1, p. 12116.

Yan, Wei (2014). "(Meta) Knowledge modeling for inventive design". PhD thesis. Université de Strasbourg.

Yan, Wei, Cecilia Zanni-Merk, and François Rousselot (2011). "An Application of Semantic Distance between Short Texts to Inventive Design." In: *KEOD*, pp. 261–266.

Yan, Wei et al. (2013). "Ontology matching for facilitating inventive design based on semantic similarity and case-based reasoning". In: *International Journal of Knowledge-Based and Intelligent Engineering Systems* 17.3, pp. 243–256.

Yang, Zhilin et al. (2019). "Xlnet: Generalized autoregressive pretraining for language understanding". In: *Advances in neural information processing systems*, pp. 5754–5764.

Yeap, Tralvex, Gim Hwa Loo, and Serena Pang (2003). "Computational patent mapping: intelligent agents for nanotechnology". In: *Proceedings International Conference on MEMS, NANO and Smart Systems*. IEEE, pp. 274–278.

Yessenalina, Ainur and Claire Cardie (2011). "Compositional matrix-space models for sentiment analysis". In: *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pp. 172–182.

Yoon, Janghyeok and Kwangsoo Kim (2011). "An automated method for identifying TRIZ evolution trends from patents". In: *Expert Systems with Applications* 38.12, pp. 15540–15548.

Zhai, Dongsheng, Mengyang Li, and Wenhao Cai (2020). "TRIZ Technical Contradiction Extraction Method Based on Patent Semantic Space Mapping". In: *2020 The 11th International Conference on E-business, Management and Economics*, pp. 125–130.

Zhang, Daniel et al. (2021). "The ai index 2021 annual report". In: *arXiv preprint arXiv:2103.06312*.

Zou, Will Y et al. (2013). "Bilingual word embeddings for phrase-based machine translation". In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1393–1398.

Publications

1. ***Ni Xin***, et al. "PatRIS: Patent Ranking Inventive Solutions." International Conference on Database and Expert Systems Applications. Springer, Cham, 2021. (***Published***)

2. ***Ni Xin***, Ahmed Samet, and Denis Cavallucci. "Replicating TRIZ Reasoning Through Deep Learning." International TRIZ Future Conference. Springer, Cham, 2021. (***Published***)

3. ***Ni Xin***, Ahmed Samet, and Denis Cavallucci. "Similarity-based approach for inventive design solutions assistance." Journal of Intelligent Manufacturing (2021): 1-18. (***Published***)

4. ***Ni Xin***, Ahmed Samet, and Denis Cavallucci. "Build Links Between Problems and Solutions in the Patent." International TRIZ Future Conference. Springer, Cham, 2020. (***Published***)

5. ***Ni Xin***, Ahmed Samet, and Denis Cavallucci. "Similarity Computation Supporting Creative Activities." Proceedings of the Sixth International Conference on Design Creativity (ICDC 2020). 2020. (***Published***)

6. ***Ni Xin***, Ahmed Samet, and Denis Cavallucci. "An approach merging the IDM-related knowledge." International TRIZ future conference. Springer, Cham, 2019. (***Published***)

Recherche de Solutions Inventives dans les Documents de Brevets par Traitement du Langage Naturel

Doctorante: Xin Ni (ICUBE)
Directeur de thèse: Denis Cavallucci (ICUBE)
Encadrante: Ahmed Samet (ICUBE)
Laboratoire: ICUBE (UMR CNRS 7357)

1 Contexte de la Recherche

L'innovation est un facteur clé pour les entreprises qui développent des produits et s'engagent dans une logique de progrès continu au sein d'un secteur fortement concurrencé. Les solutions inventives, qui sont une sorte de résultats significatifs de l'innovation, peuvent être utilisées pour résoudre de manière inventive des problèmes difficiles afin de faciliter les activités de recherche et développement. En particulier, un nombre croissant de chercheurs ont remarqué que la connaissance de différents domaines industriels pourrait être utile pour élaborer des solutions inventives afin de résoudre des problèmes cibles complexes.

Aujourd'hui, dans le contexte de cette préoccupation croissante pour l'innovation en matière d'ingénierie, la demande de solutions d'ingénierie inventive a également augmenté rapidement dans les entreprises (Shirwaiker and Okudan, 2008; Jardim-Goncalves et al., 2011; Smirnov et al., 2013). En outre, l'exploration de champs de connaissances plus larges pour trouver des inspirations inventives est devenue une alternative importante pour relever les défis complexes de la fabrication (Ni, Samet, and Cavallucci, 2021). Cependant, de nombreuses entreprises s'appuient encore sur l'expérience des ingénieurs, sur des séances de remue-méninges entre différents experts ou sur la recherche classique de solutions sur Internet pour promouvoir les activités de recherche et développement. Ces méthodes obsolètes ne dorénavant plus s'adapter à la croissance actuelle du renouvellement permanent de l'information et des données dans tous les domaines.

En outre, en tant que part importante de leur stratégie, la conception inventive est devenue un facteur important pour les entreprises afin de survivre dans un contexte de concurrence exacerbée (Hao et al., 2019; Renjith, Park, and Kremer, 2020). De plus, le niveau d'innovation des produits se prête d'avantage à la fabrication ouverte (Kusiak, 2016). Bien que la plupart des ingénieurs ont compris l'importance du mélange des connaissances entre différents domaines pour la création et le développement de produits (Whiteside et al., 2009), les connaissances inventives sont toujours intrinsèquement liées aux personnes qui les utilisent (Girodon et al., 2015) et la circulation des connaissances dans différents domaines reste difficile.

Depuis le milieu du 20e siècle, TRIZ, la théorie de la résolution de problèmes inventifs initialement proposée par Altshuller (Altshuller, Shulyak, and Rodman, 0040) après avoir analysé des millions de brevets dans tous les domaines de l'industrie, a commencé à être utilisée pour améliorer et faciliter la résolution de problèmes technologiques (Altshuller, 1999). Les modèles de TRIZ et notamment les 40 principes inventifs sont souvent utilisés pour générer des solutions inventives relatives à un problème cible. Les contradictions sont généralement contenues dans les problèmes d'ingénierie. Elles sont classées en tant que contradictions techniques ou physiques. La matrice des contradictions est un des outils les pluys répandu de TRIZ. Elle est utilisée pour résoudre les contradictions en fonction des 40 principes inventifs et des 39 paramètres du système. Comme le montre la Fig. 1, un processus classique de résolution de problèmes TRIZ est présenté.

- Le problème cible spécifique est d'abord préparé par l'utilisateur.
- À l'aide de 39 paramètres du système, TRIZ transforme le problème spécifique en un problème général.
- Trouver des solutions générales par des modèles TRIZ pour le problème général.
- Transformer les solutions générales en solutions spécifiques en appliquant les modèles TRIZ.

Au cours des dernières décennies, un nombre croissant de travaux de recherche approfondis ont été proposés pour faciliter d'avantage la résolution des problèmes. De nouvelles techniques et de nouveles techniques sont utilisés dans ce domaine, comme les approches d'apprentissage

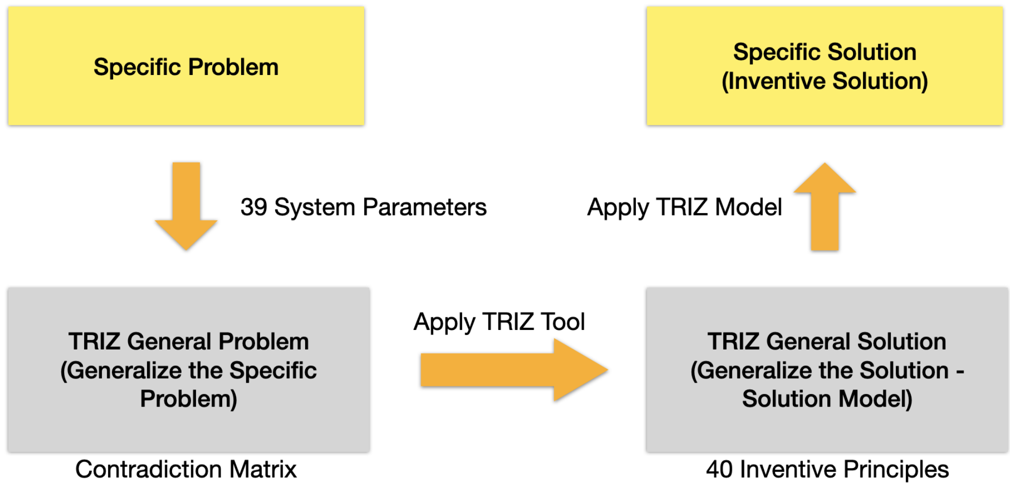


Fig. 1. Le processus de résolution de problèmes de TRIZ

automatique, l'exploration de brevets, les méthodes d'apprentissage profond, la conception assistée par ordinateur, etc.

En particulier, par rapport à ces techniques, les documents de brevet jouent un rôle important pour présenter les dernières connaissances inventives dans chaque domaine. En outre, les connaissances relatives à la méthode de conception inventive (IDM) qui dérive de TRIZ (Sheu, Chen, and Yu, 2012) sont principalement contenues dans les documents de brevet (Ni, Samet, and Cavallucci, 2019). Plus de 80 % des connaissances techniques de l'humanité sont décrites dans des documents de brevet (Souili, Cavallucci, and Rousselot, 2015) et l'Organisation mondiale de la propriété intellectuelle a révélé que 90% à 95 % de toutes les inventions du monde se trouvent dans des documents de brevet (Yeap, Loo, and Pang, 2003). En outre, les documents de brevet constituent des ressources intellectuelles importantes pour la protection des intérêts des individus, des organisations et des entreprises (Ni, Samet, and Cavallucci, 2019). Ils fournissent également des informations précieuses pour résoudre les problèmes d'ingénierie et renforcer l'inventivité. Les connaissances innovantes contenues dans les brevets tendent toujours à présenter les solutions les plus récentes pour résoudre les problèmes.

Par conséquent, l'utilisation efficiente et efficace des connaissances inventives existantes contenues dans les documents de brevet de différents domaines mérite d'être explorée plus avant. Néanmoins, il a toujours été difficile pour les ingénieurs qui n'ont pas une connaissance approfondie des différents domaines d'utiliser pleinement les connaissances inventives contenues dans les documents de brevet. En particulier, l'exploration de plusieurs brevets par un expert s'avère être une tâche compliquée. Il est donc important et utile d'automatiser l'ensemble du processus de recherche de solutions inventives dans les documents de brevet de différents domaines pour résoudre de manière inventive des problèmes cibles.

Aujourd'hui, divers outils et techniques d'apprentissage automatique ont été mis au point pour aider les experts à analyser les brevets et à automatiser le processus d'extraction des connaissances. En effet, les techniques de traitement du langage naturel (NLP) ont fait des progrès importants. Nous utilisons donc particulièrement les techniques NLP dans ce travail. Comme illustré dans la Fig. 2, la thèse est réalisée dans le cadre de la résolution de problèmes à partir de documents de brevets avec des techniques NLP. Elle a pour but d'aider les ingénieurs à résoudre des problèmes complexes et multidisciplinaires par le biais du processus de recherche de problèmes similaires, de mise en correspondance des problèmes et des solutions et de classement des solutions inventives. Dans ce cadre, un grand nombre de documents de brevets provenant de différents domaines sont utilisés pour extraire des solutions inventives latentes pour des problèmes réels donnés. Les connaissances inventives existantes et publiées contenues dans ces documents de brevet peuvent être un outil disponible et utile pour les ingénieurs qui n'ont pas une grande compréhension des connaissances de différents domaines afin de faciliter leurs activités de R&D.

Dans cette thèse, nous nous concentrons sur le travail du processus de recherche automatique de solutions inventives complètes, qui concerne l'utilisation des connaissances inventives

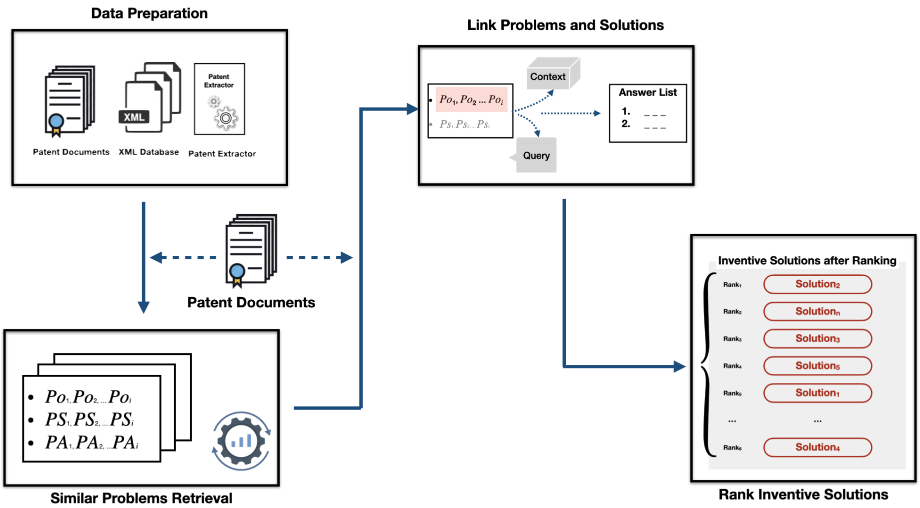


Fig. 2. L'architecture globale de la thèse

contenues dans différents domaines de connaissances pour résoudre de manière inventive les problèmes cibles des milieux industriels.

2 Motivation de la Recherche

La motivation de ce travail est de faciliter et d'automatiser le processus de résolution inventive de problèmes en se basant sur les technologies NLP.

Comme indiqué dans la Chapitre 1, les documents de brevet de différents domaines contiennent la richesse des connaissances inventives publiées et les plus récentes. Ils fournissent également des informations précieuses pour résoudre les problèmes d'ingénierie et renforcer l'inventivité. Ces connaissances inventives contenues dans les documents de brevet tendent toujours à présenter les solutions les plus récentes pour résoudre les problèmes. En outre, les connaissances inventives contenues dans les brevets pourraient être définies des **problèmes** (Souili, Cavallucci, and Rousselot, 2015). Le problème décrit les caractéristiques insatisfaisantes des méthodes ou des situations existantes.

Par exemple, pour le cas d'utilisation du stylo tactile[1].

– *non-conductive material like plastic could hampers users to operate the pen with wearing gloves, having very dry skin, or some situations in which the user does not make good conductive contact with the device to the touch screen.*

Ce problème pourrait se manifester surtout lorsque l'environnement est froid. Le brevet propose donc des *solutions partielles* qui apportent des améliorations ou des changements aux problèmes définis. Une solution partielle pourrait être :

– *replacing the inner molding built by non-conductive material of touch pen with a ideally metal material device so that the stylus tip operates even when held by an extremely good insulator.*

Par conséquent, l'exploitation de ces connaissances inventives en matière de résolution de problèmes contenues dans les documents de brevet peut être utilisée pour résoudre les problèmes cibles de différents domaines lorsque ces problèmes sont suffisamment similaires.

Cependant, il existe déjà une grande quantité de documents de brevet dans le monde et de nombreux nouveaux documents de brevet sont publiés chaque année. Il est impossible de vérifier manuellement ces très nombreux documents de brevet. En outre, les documents de brevet sont répartis sur un large éventail de domaines différents. Prenons l'exemple du système de brevets classique, l'Office des brevets et des marques des États-Unis (USPTO).

Les brevets américains se répartissent en trois catégories : les brevets d'utilité, les brevets de design et les brevets de plante. Les brevets d'utilité sont accordés à toute personne qui

[1] le lecteur peut se référer à ce lien pour le brevet complet `https://patents.google.com/patent/US8847930B2/`

invente ou découvre un procédé, une machine, un article manufacturé ou une composition de matière nouveaux et utiles, ou toute amélioration nouvelle et utile de ceux-ci ; les brevets de design sont accordés à toute personne qui invente un design nouveau, original et ornemental pour un article manufacturé ; les brevets de plante sont accordés à toute personne qui invente un design nouveau, original et ornemental pour un article manufacturé. Parmi eux, 90% des brevets américains sont des brevets d'utilité, qui protègent l'utilité ou les aspects fonctionnels d'une invention (Ni, Samet, and Cavallucci, 2019). En outre, le lien de l'invention des brevets d'utilité entre différents domaines est un peu plus fort que celui des brevets de design et de plantes. Ainsi, dans ce travail, nous prenons uniquement les brevets d'utilité et ils peuvent être classés en huit domaines différents : les nécessités humaines (HN), les opérations d'exécution (PO), la chimie (C), les textiles (T), les constructions fixes (FC), le génie mécanique (ME), la physique (P) et l'électricité (E). De toute évidence, pour permettre aux ingénieurs d'avoir une large compréhension de ces différents domaines, la connaissance est un obstacle crucial. Ainsi, ce travail vise à supprimer cet obstacle pour les ingénieurs afin qu'ils soient en mesure d'obtenir facilement les dernières connaissances en matière de résolution de problèmes inventifs à partir de documents de brevets dans différents domaines.

En outre, nous remarquons en particulier que les documents de brevets de différents domaines peuvent contenir des problèmes sémantiques similaires. Lorsque ces problèmes sont suffisamment similaires dans la partie sémantique, leurs solutions correspondantes proposées par des documents de brevets de domaines différents pourraient être un type de solutions inventives pour le problème cible. La distance plus grande entre leurs domaines pourrait fournir des solutions latentes plus inventives ou des inspirations pour résoudre le problème cible. En particulier, la plupart des ingénieurs ne parviennent pas à maitriser les connaissances des domaines éloignés. Pour mieux utiliser ces connaissances éloignées, nous proposons un nouveau processus de recherche de connaissances inventives.

Dans le contexte de l'automatisation d'un processus complexe d'extraction de données, cela nous incite à automatiser l'ensemble du processus de recherche de solutions inventives afin d'éviter les travaux de coopération complexes entre les différentes phases de préparation. De la préparation des données à la recherche de problèmes similaires, en passant par la mise en correspondance des problèmes et des solutions et le classement des solutions inventives, ces différentes étapes coopèrent ensemble pour permettre l'automatisation de l'ensemble du processus de recherche de solutions inventives, en particulier lorsqu'un grand nombre de documents de brevets sont saisis.

Les approches d'apprentissage profond ont connu un développement rapide ces dernières années. De nombreuses approches de pointe en matière de réseaux de neurones ont obtenu des résultats prometteurs dans différents domaines et tâches de recherche, notamment dans le domaine du langage naturel. En raison de la nature de leur conception et de leur structure, les différentes approches de réseaux neuronaux présentent des performances différentes pour différentes tâches. Par exemple, pour prédire le mot cible sur la base d'informations contextuelles longues, les réseaux de mémoire à long terme (LSTM) sont plus performants que les réseaux neuronaux Word2vec. Il peut mieux apprendre les informations contextuelles plus longues autour du mot cible grâce à sa conception originale d'oubli et de portes de mémoire. En particulier, les documents de brevet sont différents des autres types de textes. Ils contiennent beaucoup de longues phrases par rapport aux autres documents. En outre, de nombreuses expressions complexes et des mots uniques sont toujours contenus dans ces longues phrases. Les informations contextuelles sont généralement plus longues et difficiles à comprendre. Tous ces éléments font que la tâche de recherche de solutions inventives sur les documents de brevets est plus unique que d'autres travaux. Cela nous incite donc à explorer différentes approches d'apprentissage profond pour répondre à notre tâche.

Dans l'ensemble, les défis susmentionnés nous incitent à proposer un processus de recherche automatique de solutions inventives. Les approches proposées visent à combiner des méthodes d'exploration de données, des technologies de calcul de similarité sémantique et des approches d'apprentissage profond pour automatiser la recherche de solutions inventives à partir d'un grand nombre de documents de brevet. Elles peuvent fournir aux ingénieurs qui n'ont pas une connaissance approfondie de différents domaines une nouvelle façon de trouver des inspirations inventives à partir de documents de brevets pour des problèmes donnés. D'autre part, à notre connaissance, notre travail est le premier à utiliser pleinement la connaissance des documents de brevet de différents domaines pour fournir automatiquement des solutions inventives. Il facilite davantage le travail de recherche sur la résolution de problèmes dans le domaine de TRIZ.

3 Contributions de la Thèse

Dans cette thèse, nous résumons les contributions proposées dans plusieurs directions de recherche de la façon suivante:

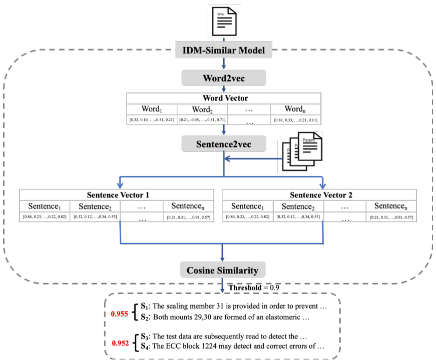


Fig. 3. Le cadre d'IDM-Similar

– **Modèle IDM-Similar basé sur les réseaux neuronaux Word2vec** : Un modèle de recherche de problèmes similaires appelé IDM-similaire est proposé par la thèse, comme le montre la Fig. 3. Selon les réseaux neuronaux Word2vec de Google (Mikolov et al., 2013), Le modèle IDM-Similar est capable d'extraire des problèmes similaires de documents de brevets de différents domaines. Il obtient le vecteur de phrase pour chaque problème cible dans les documents de brevet via Word2vec. De plus, nous formons d'abord le modèle Word2vec sur la base d'un ensemble de données Wikipedia en anglais. Il peut ainsi apprendre la similarité sémantique entre différents mots. Les représentations des phrases peuvent ensuite être réalisées à partir des représentations des mots. La métrique de similarité Cosine est également combinée pour prédire la valeur de similarité entre les paires de phrases. Le modèle IDM-Similar est finalement capable d'extraire les phrases problématiques similaires des documents de brevet par le biais du calcul de la similarité.

– **Modèle SAM-IDM basé sur des réseaux de neurones à mémoire à long terme (LSTM)**: Afin de permettre au modèle de mieux apprendre les informations contextuelles longues en fonction des caractéristiques du document de brevet, un nouveau modèle d'extraction de problèmes similaires appelé SAM-IDM reposant sur des LSTM est proposé, comme l'illustre la Fig. 4. Il combine un modèle LSTM de Manhattan pour effectuer la tâche de comparaison de similarité sémantique entre différentes phrases. En outre, la mise en œuvre d'un processus d'élagage est utilisée pour garantir un niveau plus élevé d'inventivité et d'efficacité temporelle. Comparé au modèle IDM-Similar, SAM-IDM affiche de meilleures performances sur des documents de brevets américains des milieux industriels.

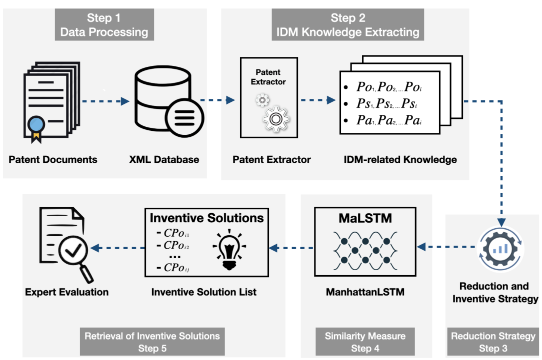


Fig. 4. Le cadre de SAM-IDM

– **IDM-Modèle de mise en correspondance basé sur les réseaux neuronaux XL-Net** : Faire correspondre les problèmes et les solutions correspondantes dans les documents de brevets, un modèle appelé IDM-Matching est proposé dans la thèse, comme le montre la Fig. 5. Il combine un réseau de neurones de pointe appelé XLNet dans le domaine du langage naturel. En particulier, nous traitons cette tâche comme un système de réponse aux questions (Ravichandran and Hovy, 2002). Nous convertissons spécialement chaque problème en une requête afin d'utiliser pleinement les réseaux neuronaux XLNet et d'éviter les inconvénients des méthodes traditionnelles de correspondance lexico-syntaxique. Ce modèle vise à établir le lien entre les problèmes et les solutions dans les documents de brevet afin de faire correspondre les problèmes similaires du modèle SAM-IDM avec les solutions inventives des documents de brevet de différents domaines.

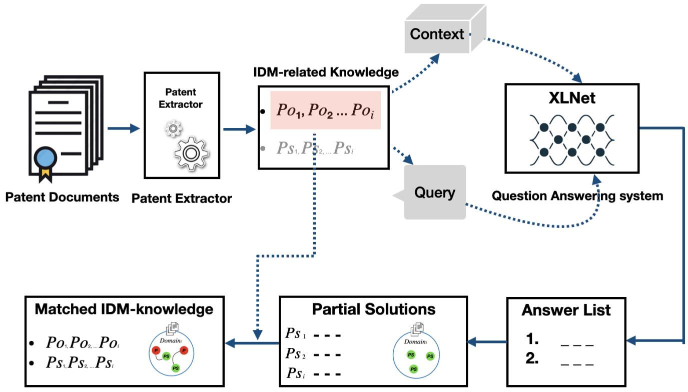


Fig. 5. Le cadre de l'IDM-Matching

– **Modèle PatRIS basé sur l'analyse de décision à critères multiples** : Un modèle appelé PatRIS basé sur l'analyse décisionnelle à critères multiples (MCDA) est proposé pour classer les solutions inventives latentes. Lorsqu'un grand nombre de documents de brevet est utilisé comme entrée du modèle SAM-IDM, plusieurs problèmes similaires peuvent être générés à partir de documents de brevet de différents domaines. Plusieurs solutions inventives latentes correspondantes peuvent donc être obtenues via le modèle IDM-Matching. Afin de mieux classer ces solutions inventives latentes, le modèle PatRIS combine une approche MCDA appelée TOP-SIS. En outre, plusieurs indicateurs d'inventivité des brevets et la valeur de similarité sont combinés avec différents poids pour construire un système de classement de l'inventivité des solutions cibles. Ce travail vise à fournir aux ingénieurs un moyen d'obtenir le plus de solutions inventives possibles lorsque le nombre de solutions inventives latentes est élevé.

– **Un démonstrateur nommé PatentSolver**: Suite aux recherches susmentionnées, un démonstrateur nommé PatentSolver est proposé dans cette thèse. Il contient les modèles décrits précédemment et vise l'automatisation de l'ensemble du processus. Plusieurs fonctions telles que la présentation des détails du brevet, la recherche du numéro de brevet, la présentation de la liste des problèmes similaires ainsi que la liste des solutions correspondantes, et le classement des solutions inventives, sont toutes développées et assemblées dans PatentSolver. Il s'agit d'un prototype du logiciel réel à fournir aux ingénieurs. Le futur logiciel peut être basé sur PatentSovler pour traiter les données de brevet en temps réel afin de permettre au secteur industriel de bénéficier de nos travaux de recherche et de faciliter davantage les activités de R&D.