Intelligent decision support system for CV evaluation based on natural language processing

Hejab Alfawareh [1,*], Shaidah Jusoh [1,2]

[1]Faculty of Computing and IT, Northern Border University, Arar, Saudi Arabia
[2]King Hussein School of Computing Sciences, Princess Sumaya University for Technology, Amman, Jordan

ARTICLE INFO

ABSTRACT

A curriculum vitae (CV) has become one of the most important documents for applying or hiring any job positions. The CV document is normally assessed by a hiring committee based on some predefined criteria. However, the assessment process is lengthy and fraught with human-engendered bias. To the best of our knowledge, none tool that is able to read and filter CV documents which are presented in texts form had been introduced. The purpose of this research is to create a tool that is able to screen and filter hundreds of CVs automatically. This paper proposed an approach based on natural language techniques to develop the tool. The tool can be considered as a decision support system (DSS) in recruiting new employees. One hundred seventy-eight CV documents were used to test the proposed approach. Obtained results suggest that the proposed approach is successful.

1. Introduction

Human resource management (HRM) in any organization is responsible for attracting, assessing, hiring, training, and rewarding employees. In other words, human resource processes include recruitment, development, and management of competent employees. Recruitment activities include screening, selecting, and hiring qualified candidates. Selecting and hiring the "right" candidate for any position can be difficult and costly to many human resource departments.

The first process in hiring a new employee is advertising a position on websites, newspapers or social media. In organizations such as universities or colleges, the process of hiring a new academic staff normally starts with an advertisement from a department in need of new academic staff. This advertisement is normally displayed on university or college websites, in a newspaper, and also on relevant social media sites. In the advertisement, any interested individuals are asked to provide an application letter, a curriculum vitae (CV) and contact information of references. A CV is a document which is normally contain personal information, such as name, address, phone numbers, and other information relevant to a dedicated position such academic qualifications, experiences and so on. Almost all universities conduct the same practice in which an individual is asked to submit an application along with a CV document to the dean of the college or department via email. In normal circumstances, a departmental hiring committee is formed under the leadership of a dean. Committee members are selected based on certain criteria such as seniority or expertise needed to select a skilled candidate. The second process of hiring is reviewing each submitted CV. For one faculty position, normally the college/faculty may receive hundreds of emails. Hiring the right candidate is not an easy task. CV review is a process of scrutinizing each document page by page. The hiring committee is responsible for making decisions, assigning a score to each evaluated CV according to its content based on certain criterion such as advanced degrees, publication, research and teaching experiences. In this case, the hiring committee members will use his/her experience, personal judgment, and intuition, to select the most suitable candidate for an offered position. The process is lengthy. However, reliance solely on human knowledge, judgment, and preference may also cause inconsistency in decision making. In some cases, it also may cause bias against certain applicants. Decision support systems can be used to help decision makers in improving and accelerating decision processes and making a fair and consistent decision (Pérez et al., 2010). A

decision support system is considered intelligent if it can mimic human decision makers (Prakash and Sarkar, 2015). To achieve such a system, a modeling tool must be integrated with human knowledge (Sperandio et al., 2014; Indurkhya and Damerau, 2010).

Intelligent Decision Support Systems (IDSSs) success mainly depends on available data, information, and knowledge. Nearly, 80% of present digital information exists in the form of non-structures, and the majority of the non-structural information exists in the form of texts that are made up of natural language description (Indurkhya and Damerau, 2010).

The deployment of an intelligent decision support system in the hiring process can greatly improve the search for a qualified and competent candidate. Consequently, natural language processing technique is required to process this digital information. To the best our knowledge, a methodology for developing an intelligent CV evaluation tool has not been proposed so far. The aim of this paper is to introduce a DSS for hiring academic staffs. The system should be intelligent enough to evaluate and filter a pile of CV documents. The DSS system is developed by utilizing natural language processing techniques. This paper is organized as follows. Section 2 presents related work, section 3 presents our intelligent CV evaluation tool. A brief discussion on implementation and experiments is presented in section 4. Section 5 presents a conclusion.

2. Related works

2.1. An overview of DSS and IDSS

An IDSS is designed and developed by adding Artificial Intelligence (AI) functions to a traditional DSS, for the purpose of giving new capabilities to the system such as the ability to mimic human thinking and reasoning in decision-making phases. A traditional DSS was defined as a "model-based set of procedures for processing data and judgments to assist a manager in his decision making" (Little, 2004). An IDSS is intelligent if it has the capability of capturing, organizing and interpreting data in helping human decision makers during decision-making processes. Practically, any IDSS is intended to support the decision-making process, not to replace the decision maker's task. IDSSs help decision makers during different phases of decision making by integrating modeling tools and human knowledge.

IDSSs' programs are support systems that contain some degree of human knowledge and intelligence in one or more components, such as in interfaces, databases, and model management components (Turban et al., 2010). Along with knowledge-based decision analysis models and methods, IDSSs incorporate databases, model bases and intellectual resources of individuals or groups. Wan and Lei (2009) aimed at providing decision makers with

some pre-coded domain knowledge (Burstein and Carlsson, 2008)

As stated by Phillips-Wren et al. (2009), human knowledge and intelligence can be applied to a system in various ways. Intelligent DSS, along with knowledge-based decision analysis models and methods, incorporate well databases, model bases and intellectual resources of individuals or groups to improve the quality of complex decisions. IDSSs work under an assumption that decision makers know what is needed for the solution and how to achieve it. In that case, IDSSs give full control to the decision makers regarding information acquisition, evaluation and making the final decision. An IDSS is flexible and adaptable, specifically and is developed to support a solution of a non-structured management problem (Quintero et al., 2005). Moreover, the system is an interactive computer-based system which incorporates artificial intelligence techniques, uses data, and expert knowledge to solve semi structured problems (Turban and Aronson, 2001).

According to Bohanec and Rajkovič (1990), a typical IDSS consists of five main components, database system, model base system, knowledge-based system, and user interface and kernel/inference engine. An IDSS is suitable for generic problems which require repetitive decisions. It provides services to users and attempts to satisfy the user's requirements through interaction, cooperation, and negotiation. The system also offers support for well-defined tasks such as data conversion, information filtering, and data mining, as well as ill-structured tasks in dynamic cooperation (Matsatsinis and Siskos, 1999). Without doubt, the system can reduce decision making costs. It is worth to mention that there are a number of techniques which belongs to artificial intelligence domains which can be integrated into to a decision support system to make IDSS (Gao et al., 2007). These include techniques in the area of expert system, fuzzy logic and natural language processing (NLP).

2.2. Expert system for DSS

An expert system (ES) is a method in which human knowledge is encoded as rules and stored in a knowledge base. The main components of an expert system include user interface, inference engine, and knowledge base. The integration of an ES and a DSS results in an IDSS; a system which utilizes human knowledge in enhancing the capabilities of decision makers in understanding a decision problem and selecting an alternative solution. The IDSS based on expert system incorporates the stored knowledge in the knowledge-based to assist the decision-making process through a set of recommended solutions reflecting domain expertise (Quintero et al., 2005). The IDSS based on ES are most suitable for applications which require interpretation, prediction, diagnosis, planning, monitoring, and control. Other options which an expert system offers are explanation of the decisions,

analysis of options and handling qualitative knowledge (Turban et al., 2010).

2.3. Fuzzy logic for DSS

Fuzzy logic is based on a fuzzy set theory, which is a generalization of the classical set theory as proposed by Zadeh (1965), the father of Fuzzy. The classical set is also called a clear set, which is also known as Boolean logic or binary. A fuzzy system is a system in which the knowledge is represented in a linguistic variable such as warm, cold, hot, and so on, where there is no absolute truth and false value. Fuzzy logic along with an expert system has been used to develop an intelligent DSS for solving problems involving uncertainty. For example, many human experts describe life problems and solutions through vague words, such as *good*, *bad*, and *excellent*. Using Boolean logic, good will take a value of 1, and not good will have a value of 0. However, in real life there is a degree of being *good*. Different people may have different degree of *good*. Thus by using fuzzy logic it allows us to have a range of values for good, from 0.1 to 0.9. These values are represented as fuzzy sets. Thus, a variable linguistic good is described with the membership of fuzzy function. Fuzzy is suitable to handle incomplete and uncertain information (Jusoh and Alfawareh, 2013). Furthermore, it is useful for many applications that deal with human and language variables, such as patients who have to describe degree of his/her illness. Consequently, applying a fuzzy set theory on an expert system for decision making process offers instruments for modeling and dealing with expert rules (Froelich and Ananyan, 2008). It is possible to transform expert rules into mathematical terms by modeling linguistic variables in form of fuzzy sets. Consequently, when fuzzy is applied to the expert system, the decision reasoning is carried out using fuzzy logic.

2.4. NLP for DSS

In DSS, NLP is used for two main tasks; understanding natural language or generating natural language. Natural language understanding can be used to process texts to support decision making phases. Examples of NLP techniques which can be used for IDSS are tokenization, morphological analysis, quality improvement using string similarity, part of speech tagging, collocation analysis, named entity recognition, word association, keyword extraction, summarization and concept analysis, classification, clustering and semantic analysis (Zolnoori et al., 2012). NLP techniques can offer an intelligent feature of a DSS, because it has the ability to identify and extract required data for a decision making process from free texts. However, much effort in utilizing NLP techniques for a decision support system has been only shown in the medical field. For example, Aronsky et al. (2001) reported the usefulness of NLP techniques for a clinical decision support system (CDSS). They claimed that

performance of CDSS was significantly better with the NLP output. According to Demner-Fushman et al. (2009), NLP CDSS have been not only targeted to clinicians, but also other users such as researchers, patients, administrators, students, and coders.

2.5. Examples of IDSS

Researchers of intelligent systems have proposed IDSS for various application domains. Dasgupta and Gonzalez (2001) proposed IDSS for providing active detection and automated responses during intrusions of networks. The system was designed to be a sense and response system which can monitor various activities on the network such as malfunctions, faults, abnormalities, misuse, deviations, intrusions, etc.

Abel et al. (2004) described the PetroGrapher system, an intelligent database application to support petrographic analysis, interpretation of oil reservoir rocks, and management of relevant data using resources from both knowledge-based system technology and database technology. In this project, the visual tacit knowledge was applied in petrographic analysis.

Ahmad and Simonovic (2006) designed and developed IDSS for flood management, for the Red River Basin in Manitoba Canada in 2006. The system was developed as a virtual planning tool which can address both engineering and non-engineering issues related to flood management. The system was by integrating an expert system and artificial neural network to assist in selecting suitable flood damage reduction options and forecasting floods, modeling the operation of flood control structures and describing the impacts of floods in time and space.

Doukas et al. (2007) presented an intelligent decision support model using rule sets based on a typical building energy management system in 2007. The decision support model's infrastructure is based on the characteristics of typical building energy management system logic. Gajzler (2010) presented possibilities of using text mining techniques in building IDSS for the construction industry. In this work, the text mining approach is suggested to be useful for building up knowledge-based.

Shen et al. (2013) presented an Experience Mining System (ExMS). The system was built based on the theories of experience representation, storage and mining. Major components of the system include a Sustainable Urbanization Practices Database (SUPD), a Refinery process, and a Mine-sweeper. ExMS can facilitate decision-makers in the selection of strategies and solutions when addressing urbanization practice challenges.

Lee et al. (2014) presented an intelligent data management induced resource allocation system which aims at providing effective and timely decision making for resource allocation. The system was comprised of product materials, people, information, control and supporting function for the effectiveness in production. The proposed system incorporates a database management system and fuzzy logic to

analyze data for intelligent decision making, and radio frequency identification for result verification.

Salah et al. (2014) presented a work on proposing IDSS for evaluating water pollution in Tigris Basin. In their work, the frame of the water pollution DSS was based on a mathematical model. The system detects types and causes of pollution and suggests a decision for cleaning the water so that it can be fit for human consumption. They stated that the results support the primary decision in emergency cases and offer a suggestion to apply a suitable method for treating polluted water. In spite of many approaches and algorithms have been deployed for designing and implementing IDSS, very few have directly used natural language processing techniques. In this paper, we present an IDSS for hiring academic staffs by introducing an intelligent CV evaluation tool.

3. CV evaluation

The purpose of this study is to propose an approach for creating a tool that is able to screen CV documents which are in the form of unstructured texts. The tool is considered as a DSS.

In a traditional procedure, applicants submit their CVs to the dean of the faculty/college, and then the dean will set up a hiring committee to review each submitted CV. If applicants fulfill all requirements set up by the college, the applicants will be selected as shortlist applicants. Other submitted CVs will be archived in the college repository for future use, if needed. A shortlist applicant will be called for a face to face interview. If an applicant is successful in this interview, then an offer letter will be sent to the applicant.

In this study, a CV screening tool (Fig. 1) is proposed to handle the process of screening and evaluating CV for each applicant using criteria established prior to the interview process. As shown in Fig. 1, if a CV's score is above threshold score, then the applicant is listed into a shortlist, otherwise the CV is archived. The final decision is made after the oral interview.

The decision maker will use the tool to automatically decide whether to accept or reject the CV. If the CV is accepted, then the decision maker decides the status of applicant to be placed on a shortlist and contacts the applicant for an interview session. If the applicant is successful in the interview, he/she will receive an offer letter. The proposed approach consists of 3 main components; syntactic processing, named entity recognition, and decision.
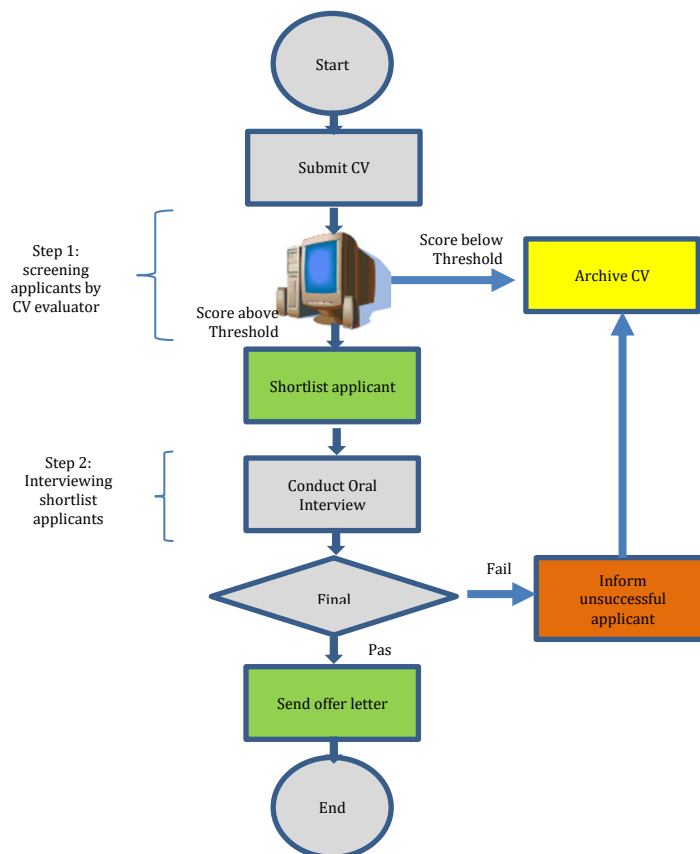

Fig. 1: CV screening tool flowchart as a part of hiring

3.1. Syntactic processing

Syntactic processing is composed of 3 major steps; sentence segmentation, tokenization, and part of speech tagging. Sentence segmentation is frequently referred to as sentence boundary detection, sentence boundary disambiguation, or sentence boundary recognition. Technically, sentence segmentation is a process of parsing a paragraph of texts into sentences by recognizing

periods (.), exclamation mark (!) and a question mark (?) at their end. However, while this is a reasonable list of punctuation characters that can end sentences, this technique does not recognize the punctuation characters that appear in the middle of sentences. For example, a sentence ''Ph.D. in Engineering System and Computing'' has `.' in the two places in a sentence where it does not mean the end of a sentence. In this case, we propose to use a split method. Using the split method on this input text, each of the characters '.', will be treated as a potential an end of a sentence, rather than a definite end-of sentence marker. With the split method, the input text is scanned, and each time it comes to this character, it needs a way of deciding whether or not it marks the end of a sentence. To solve this problem, we built a binary classifier (decision tree approach) using hand-written rules. The rationale of this is texts in a CV document are normally precise and focus. Tokenization is a process of breaking down a sentence into a list of tokens; text needs to be tokenized at least into linguistic units such as a word, punctuation, number, alphanumeric, etc. For example, an email address such as "myself@gmail.com" is considered as one token. In general, many factors can affect the difficulty of tokenizing a particular natural language. Issues of tokenization are language-specific.

The use of punctuation marks such as commas, periods, quotation marks, apostrophes, and hyphens, causes tokenization ambiguity because, the same punctuation mark can serve many different functions in a single text or sentence. For instance, the hypen used in *Hewlet-Packard* may cause an ambiguity problem. However, texts in CV use specified linguistic units to represent types of degrees or a list of publications. The main concern of CV evaluation is only the texts that describe types of study degrees, scientific publications, and other qualifications like language ability. In dealing with ambiguity of tokenization, we ignore characters such as commas, quotation marks, and apostrophes, and let the white space be a word boundary; our tokenizer would consider a separate token for any sequence of characters preceded and followed by space. This successfully tokenizes words that are a sequence of alphabetic characters, but does not take into account punctuation characters. Part-of-speech tagging is the process of assigning a part-of speech tag to each word in an input text. The important process in tagging is to tag the proper noun correctly. This step is necessary for conducting named entity extraction later.

3.2. Named entity recognition

On the most basic level, an entity in text is simply a proper noun. Name entity recognition is a process of finding spans of text that constitute proper names and then classifying the entities referred to according to predefined types. In this approach, a proper noun which start with a word journal, Ph.D., master, bachelor, proceedings, conference, book, book chapter, reviewer, language are predefined entities according to their types. These entities are stored as a keyword lexicon. If a decision maker is looking for a specific information, then the decision maker can add a specific entity as a keyword into a lexicon. For instance, it might be in a certain case that a decision maker is looking for an individual who has published in a journal entitled "ACTA NEOPHILOLOGICA", then the term ACTA NEOPHILOLOGICA will be added into the keyword lexicon. This technique reflects how decision makers' knowledge is coded and represented in a system. A list of proper noun (an array) is then compared to the entities which are stored in the lexicon. If the proper noun in the array is same as the one in the keyword lexicon, then the word is extracted and stored in an array of extracted_ entities, keyword_*kx*.

It is based on a pattern matching method. If *tx* is found to be identical to *kx*, *tx* is an identified token, then *tx* is extracted and stored into a created two-dimensional array of keyword_*kx*. An array of keyword_*kx* is arrays of identical *tx*. Consequently, the size of keyword_*kx* represents the frequency of keyword *kx* appears in the CV texts document. Assume *kx*, represents a word 'journal', if the size of keyword_*kx* is 5, then this indicates that the applicants have published 5 research articles in scientific journals, or if the size of 0, this indicates none similar token has been found.

3.3. Decision

A decision for accepting a CV to be further reviewed is made based on a score. The score calculation is a process of calculating a total score for each applicant. The decision maker can set various score merits based on a priority. For example, a Ph.D. entity will have more score merit than a M.Sc. entity, while journal entity will have more score merit than proceeding, and so on. Assume the score for PhD is 5 points, and then the score for MSc can be assigned to 3 points, and BSc to 2 points. The score calculation is formalized as in Eq. 1,

$$ts = (şk1 \times S1) + (şk2 \times S2) + (şk3 \times S3) + \cdots, + (şkn \times Sn) \tag{1}$$

where *ts* is a total score, $şk_1$ is the size of array for the first keyword, $şk_2$ is the size of array of the second keyword and $şk_n$ is the size of array for the n keyword, S_1 represents the score merit for the first keyword S_n while represents the score merit of *n* keyword. The final decision is made based on the total points score. A score threshold (T_s) is setup by the decision maker. The decision maker can set up a threshold for each assessment based on the department requirements. The decision is formalized as in Eq. 2,

$$D = ts > Ts \tag{2}$$

where D is a decision. If-then-rule can be simply used to make a decision. If the *ts* is higher than the

setup threshold, then the applicant is decided to be in the shortlist applicants.

4. Implementation

Our proposed methodology for the development of an intelligent CV evaluation tool is generic. Any high-programming languages including C, C++, C#, Java, or even Visual Basic, can be used to implement the tool. The details of development and deployment of the tool to build IDSS for hiring academic staffs, however, depends on the nature of the chosen programming language. In this work, the proposed approach had been implemented in C# platform. A prototype system consists of 4 main interfaces. The first interface as shown in Fig. 2, allows users to select and display CV documents which had been stored in a local drive of a computer, by clicking the *select and display CV* button. This prototype is able to display the documents which were saved as Microsoft Word (Word) and Portable Document Format (pdf). The structure of the CV representation is varied. The CV documents in pdf and Word formats were then converted into .txt files by our program before applying NLP techniques. Fig. 3 and Fig. 4 show examples of CV structure which were originally stored in doc and pdf file formats. The main interface of the developed prototype has 4 main buttons for a navigation purpose; *calculate CV score, details report, edit score merit, and shortlisted applicant.* The score merit can be edited by the decision maker (user). This gives flexibility to a human decision maker to alter the merit of a selected criteria based on the organization needs during processing applications.
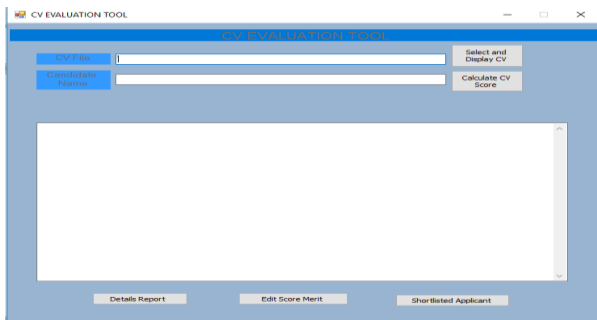

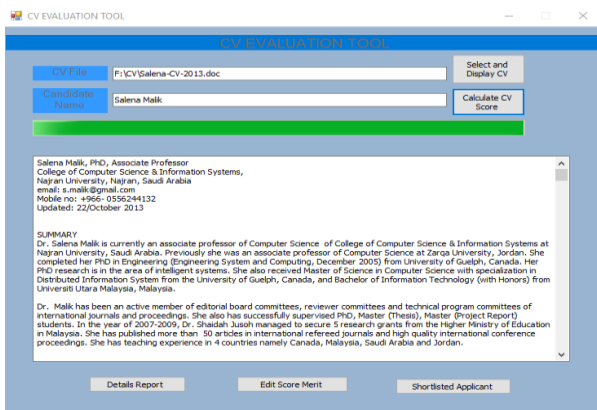**Fig. 2:** The main interface page of the prototype

Fig. 3: An example CV document which was originally saved in *doc* file format

Fig. 5 shows scores used in our experiments. We assigned 5, 3 and 2 points for the score of having a Ph.D., master's and bachelor degree. Each publication in a scientific journal, proceeding and book chapter was graded as 2, 1, and 1 point, and any published book is graded as 2 points.

Other qualification such as being a reviewer for any scientific research article is graded as 1 point. Any applicant who has an ability in more than 1 language will get another 1 point. Threshold score was set up to be 75 points.

By clicking the *calculate CV score* button, the system will extract and calculate the score for a dedicated CV and produce a detail report. The system will display to the user the following information *name*, *phone number*, *email address*, *total score*, *decision,* and *CV path at the local drive* as shown in Fig. 6. Phone numbers and email addresses have been extracted automatically, while score values are calculated based on the predefined points as stated in Fig. 5.
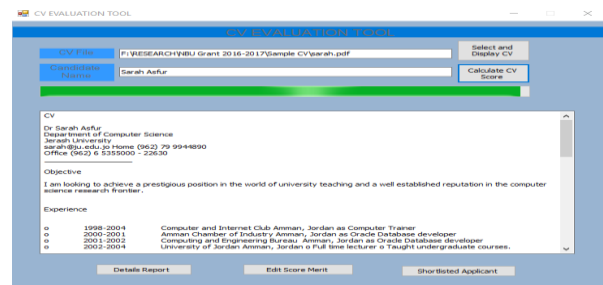

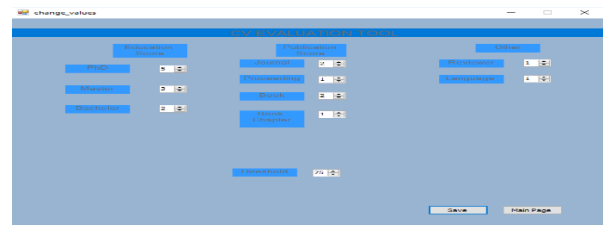**Fig. 4:** An example CV document which was originally saved as in *pdf* format

Fig. 5: A score merit interface allows a decision maker to set score point for each criterion as well as the threshold score
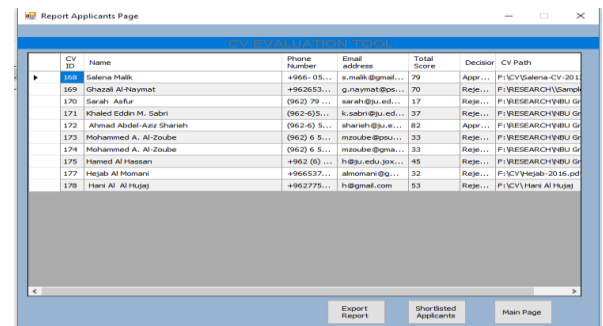

Fig. 6: A detail report of processed CVs

As a result, CV ID 172 scored 82 points (Fig. 7), which is above the threshold and it is approved to be called for an interview. With the threshold score of 75, Fig. 7 demonstrates two applicants have been approved for oral interviews. The decision maker

can save this report in a pdf format by clicking the *export shortlist* button.

Fig. 7: A shortlisted page presents applicants who has been approved for oral interviews

In this experiment, we have used 178 CV documents as a dataset. We have tested the tool with 9 criteria of qualifications including study degrees, publications and other personal achievement such the ability to speak more than one language. The obtained results have been reviewed and evaluated by a human. The comparison has been made in terms of the total score with respect to the accuracy of extracted required data on the 9 criteria of qualifications. Despite total scores which are obtained by a human are slightly higher than the ones obtained through the system, the result of approved CVs are same as in the Fig. 7.

5. Conclusion

This paper has introduced a new approach for developing a CV screening tool which can be utilized as a DSS in recruiting new employees. The new approach deploys NLP techniques including sentence segmentation, tokenization, and part of speech tagging and named entity recognition. The approach had been implemented as a prototype tool, which is able to read, extract, and suggest a preliminary decision for a given CV document. Results obtained from the preliminary tests with respect to a human evaluation, suggested the proposed approach is successful. However, we do aware that criteria for hiring academic staffs are not limited to the scholarly degrees obtained, and the number of publications only. Despite there are other issues should be taken into account such as the quality of the published journals, teaching experience, achievement and certificates, professional activities, and so on, this research work had started an initiative step of using NLP techniques to process CV documents. In the future we will integrate NLP techniques with machine learning approaches to enable the tool to produce a decision without a human intervention.

Acknowledgement

Compliance with ethical standards

Conflict of interest

The authors declare that they have no conflict of interest.

References

Abel M, Silva LA, De Ros LF, Mastella LS, Campbell JA, and Novello T (2004). PetroGrapher: Managing petrographic data and knowledge using an intelligent database application. Expert Systems with Applications, 26(1): 9-18. https://doi.org/10.1016/S0957-4174(03)00104-0

Ahmad S and Simonovic SP (2006). An intelligent decision support system for management of floods. Water Resources Management, 20(3): 391-410. https://doi.org/10.1007/s11269-006-0326-3

Aronsky D, Fiszman M, Chapman WW, and Haug PJ (2001). Combining decision support methodologies to diagnose pneumonia. In the AMIA Symposium, American Medical Informatics Association, Bethesda, Maryland, USA: 12-16.

Bohanec M and Rajkovič V (1990). DEX: An expert system shell for decision support. Sistemica, 1(1): 145-157.

Burstein F and Carlsson SA (2008). Decision support through knowledge management. In: Burstein F and Holsapple CW (Eds.), Handbook on decision support systems 1: 103-120. Springer, Berlin, Heidelberg, Germany. https://doi.org/10.1007/978-3-540-48713-5_6

Dasgupta D and Gonzalez FA (2001). An intelligent decision support system for intrusion detection and response. In the International Workshop on Mathematical Methods, Models, and Architectures for Network Security, Springer, Berlin, Heidelberg, Germany: 1-14. https://doi.org/10.1007/3-540-45116-1_1

Demner-Fushman D, Chapman WW, and McDonald CJ (2009). What can natural language processing do for clinical decision support?. Journal of Biomedical Informatics, 42(5): 760-772. https://doi.org/10.1016/j.jbi.2009.08.007 **PMid:19683066 PMCid:PMC2757540**

Doukas H, Patlitzianas KD, Iatropoulos K, and Psarras J (2007). Intelligent building energy management system using rule sets. Building and Environment, 42(10): 3562-3569. https://doi.org/10.1016/j.buildenv.2006.10.024

Froelich J and Ananyan S (2008). Decision support via text mining. In: Burstein F and Holsapple CW (Eds.), Handbook on decision support systems 1: 609-635. Springer, Berlin, Heidelberg, Germany. https://doi.org/10.1007/978-3-540-48713-5_28

Gajzler M (2010). Text and data mining techniques in aspect of knowledge acquisition for decision support system in construction industry. Technological and Economic Development of Economy, 16(2): 219-232. https://doi.org/10.3846/tede.2010.14

Gao S, Wang H, Xu D, and Wang Y (2007). An intelligent agent-assisted decision support system for family financial planning. Decision Support Systems, 44(1): 60-78. https://doi.org/10.1016/j.dss.2007.03.001

Indurkhya N and Damerau FJ (2010). Handbook of natural language processing. Vol. 2, CRC Press, Florida, USA.

Jusoh S and Alfawareh HM (2013). Applying fuzzy sets for opinion mining. In the International Conference on Computer Applications Technology, IEEE, Sousse, Tunisia: 1-5. https://doi.org/10.1109/ICCAT.2013.6521965

Lee CKH, Choy KL, Law KMY, and Ho GTS (2014). Application of intelligent data management in resource allocation for effective operation of manufacturing systems. Journal of Manufacturing Systems, 33(3): 412-422. https://doi.org/10.1016/j.jmsy.2014.02.002

Little JD (2004). Models and managers: The concept of a decision calculus. Management Science, 50(12): 1841-1853. https://doi.org/10.1287/mnsc.1040.0267

Matsatsinis NF and Siskos Y (1999). MARKEX: An intelligent decision support system for product development decisions. European Journal of Operational Research, 113(2): 336-354. https://doi.org/10.1016/S0377-2217(98)00220-3

Pérez IJ, Cabrerizo FJ, and Herrera-Viedma E (2010). A mobile decision support system for dynamic group decision-making problems. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 40(6): 1244-1256. https://doi.org/10.1109/TSMCA.2010.2046732

Phillips-Wren G, Mora M, Forgionne GA, and Gupta JN (2009). An integrative evaluation framework for intelligent decision support systems. European Journal of Operational Research, 195(3): 642-652. https://doi.org/10.1016/j.ejor.2007.11.001

Prakash N and Sarkar A (2015). Development of an intelligent decision support system for a hierarchical business organization. In the International Conference and Workshop on Computing and Communication, IEEE, Vancouver, Canada: 1-8. https://doi.org/10.1109/IEMCON.2015.7344440

Quintero A, Konaré D, and Pierre S (2005). Prototyping an intelligent decision support system for improving urban infrastructures management. European Journal of Operational Research, 162(3): 654-672. https://doi.org/10.1016/j.ejor.2003.10.019

Salah HA, Mocanu M, and Florea A (2014). Towards an integrated decision support system for the evaluation of water pollution in Tigris Basin (DSSWAPIT). In the 2014 IEEE 10th International Conference on Intelligent Computer Communication and Processing (ICCP), IEEE, Cluj Napoca, Romania: 391-398. https://doi.org/10.1109/ICCP.2014.6937026

Shen LY, Ochoa JJ, Zhang X, and Yi P (2013). Experience mining for decision making on implementing sustainable urbanization—An innovative approach. Automation in Construction, 29: 40-49. https://doi.org/10.1016/j.autcon.2012.07.001

Sperandio F, Gomes C, Borges J, Brito AC, and Almada-Lobo B (2014). An intelligent decision support system for the operating theater: A case study. IEEE Transactions on Automation Science and Engineering, 11(1): 265-273. https://doi.org/10.1109/TASE.2012.2225047

Turban E and Aronson JE (2001). Expert systems and intelligent systems. Prentice Hall, New Jersey, USA.

Turban E, Sharda R, and Delen D (2010). Decision support and business intelligence systems. Prentice Hall, New Jersey, USA. **PMCid:PMC2939501**

Wan S and Lei TC (2009). A knowledge-based decision support system to analyze the debris-flow problems at Chen-Yu-Lan River, Taiwan. Knowledge-Based Systems, 22(8): 580-588. https://doi.org/10.1016/j.knosys.2009.07.008

Zadeh LA (1965). Fuzzy sets. Information and Control, 8(3): 338-353. https://doi.org/10.1016/S0019-9958(65)90241-X

Zolnoori M, Zarandi MHF, Moin M, and Teimorian S (2012). Fuzzy rule-based expert system for assessment severity of asthma. Journal of Medical Systems, 36(3): 1707-1717. https://doi.org/10.1007/s10916-010-9631-8 **PMid:21128097**