Bayesian Synthesis for Knowledge Translation in Software Engineering: Method and Illustration

Deepika Badampudi
Department of Software Engineering
Blekinge Institute of Technology
Karlskrona, Sweden
Email: deepika.badampudi@bth.se

Claes Wohlin
Department of Software Engineering
Blekinge Institute of Technology
Karlskrona, Sweden
Email: claes.wohlin@bth.se

***Abstract*—Systematic literature reviews in software engineering are necessary to synthesize evidence from multiple studies to provide knowledge and decision support. However, synthesis methods are underutilized in software engineering research. Moreover, translation of synthesized data (outcomes of a systematic review) to provide recommendations for practitioners is seldom practiced. The objective of this paper is to introduce the use of Bayesian synthesis in software engineering research, in particular to translate research evidence into practice by providing the possibility to combine contextualized expert opinions with research evidence. We adopted the Bayesian synthesis method from health research and customized it to be used in software engineering research. The proposed method is described and illustrated using an example from the literature. Bayesian synthesis provides a systematic approach to incorporate subjective opinions in the synthesis process thereby making the synthesis results more suitable to the context in which they will be applied. Thereby, facilitating the interpretation and translation of knowledge to action/application. None of the synthesis methods used in software engineering allows for the integration of subjective opinions, hence using Bayesian synthesis can add a new dimension to the synthesis process in software engineering research.**

I. Introduction

The outcome (knowledge) of systematic literature reviews (SLR) should be useful for practitioners [1] and [2]. It should be translated into recommendations that can enable and support evidence-informed decision-making in software engineering practice [2]. The fourth step of the five-step process for adapting the practices of evidence-based software engineering (EBSE) is referred as knowledge translation [1] and [2].

Greenhalgh and Wieringa [3] argue that "objective, impersonal research findings" are unhelpful. Therefore knowledge translation should not be viewed as just supplying the outcomes of an SLR to professionals. Instead, it should be considered as a research activity involving researchers, subjective opinions of practitioners and policy-makers/decision-markers to make evidence-informed decisions [1].

Knowledge translation in software engineering is defined as "the exchange, synthesis and ethically sound application of knowledge - within a complex system of interactions between researchers and users - to accelerate the capture of the benefits of research through better quality software and software development processes" [1].

Budgen et al. [2] state that knowledge translation in software engineering is done in an ad-hoc manner and lacks adequate documentation. In addition, they highlight that "knowledge translation should itself be systematic and repeatable as possible, and it should also reflect the needs and mores of practitioners as well as of the different forms of organizational context within which they work" [1] and [2].

The need to develop guidelines for undertaking knowledge translation in software engineering has been identified [2]. The aim of this paper is to adopt/adapt Bayesian approaches to synthesis used in health research. Though Bayesian approaches are referred to as synthesis methods [4], they are essentially synthesis methods extended to support knowledge translation. They synthesize data and provide interpretation of the outcome in the application context by incorporating knowledge and experience of intended users (therefore supporting knowledge translation). As mentioned earlier, synthesis is not a separate activity as per the definition of knowledge translation in software engineering. The focus of this paper is on the synthesis process that facilitates knowledge translation rather that the knowledge translation activity itself.

In health research, Bayesian synthesis has been used to provide decision support by incorporating both subjective opinions of decision-makers and evidence/knowledge [5]. Bayesian synthesis is particularly useful when there is not enough evidence to confirm its suitability and the decisions need to be taken nevertheless in a reasonable and informed way [5]. Another advantage of Bayesian synthesis is that it takes the potential user of the analysis i.e. the practitioners' or decision-makers' and policy-makers' perspective into consideration [5]. It is more flexible and efficient in using evidence from all available sources [6] and can synthesize findings from methodologically diverse studies [7]. Hence, the Bayesian approach is an attractive method to synthesize evidence and support knowledge translation as it provides interpretations of what evidence means in a particular context by incorporating subjective opinions of the potential users of synthesized data.

Bayesian synthesis starts with a subjective opinion, and these opinions are updated based on the evidence available. Although conceptually it seems straightforward, it is not easy to implement, and some of the methodological issues remain unresolved [4]. Bayesian synthesis methods have been used in health research to synthesize primary studies [8]. However, it has not been implemented in software engineering research.

In this paper, we show how Bayesian approaches can be used to synthesize evidence and make use of the knowledge in practice. We also demonstrate the working of Bayesian approaches to synthesize and knowledge translation through examples in software engineering research.

The remainder of the paper is structured as follows. Section II describes related work and Bayesian approaches. We then provide the method description of Bayesian to synthesize and translate knowledge in the context of software engineering in Section III. The Bayesian synthesis is illustrated in Section IV. Finally in Section V we compare our method with the alternatives and conclude in Section VI.

II. Background and Related Work

We begin by providing a brief overview of the working of Bayesian principles in Section II-A (note that only the description of Bayesian principles based on which the Bayesian synthesis method is developed is provided here. The description of the Bayesian synthesis method for knowledge translation is provided in Section III). In Section II-B, we describe how Bayesian synthesis has been implemented in health research so far. In Section II-C, we discuss how synthesis is done in software engineering research. Although Bayesian methods are not used for synthesis in software engineering, it has been used in other ways which is discussed in Section II-D.

A. Overview of Bayesian synthesis

It is important to get an overview of how Bayesian theory applies to synthesis to understand the working of Bayesian synthesis. The basic idea of Bayesian synthesis is described in health research [5] and [6]. A brief summary of the method is as follows:

1) **Prior probability** - State a subjective opinion based on personal experience, excluding evidence.
2) **Likelihood** - Evaluate the evidence obtained from primary studies.
3) **Posterior probability** - Combine the prior probability and likelihood to produce a final opinion.

B. Bayesian synthesis in health research

Bayesian meta-analysis is a Bayesian approach used in health research that suits the EBSE requirement of incorporating prior knowledge and experience into the synthesis [4]. Roberts et al. use Bayesian meta-analysis to synthesize evidence from eleven qualitative and 32 quantitative primary studies [8]. The study also incorporated subjective opinions of five experts. The subjective opinions and data collected from qualitative studies were used to form the prior probability. The range of prior probabilities is known through qualitative data. Hence, it is called an informative prior probability where the range of probabilities (uncertainties) is narrow. However, we believe that the prior probability should be purely subjective based on personal experience and elicited before acquiring additional information. This allows tracking of how subjective opinions differ from collected data and how the subjective opinions are altered/refined based on the collected data. The prior probability is then updated through data collected from quantitative studies.

Two approaches called "quantitizing" [9] and "qualitizing" [7] have been proposed in health research in which the qualitative and quantitative data are considered together in the likelihood. In both the quantitizing and qualitizing approaches, the qualitative and quantitative data provide the weight of evidence in the posterior probability. In other words, the prior probability does not have any effect on the posterior probability. A uniform prior probability with same probability for every possible scenario is chosen. The two approach, namely quantitizing and qualitizing do not address the EBSE recommendation of incorporating subjective opinion into the synthesis.

In summary, the Bayesian meta-analysis method allows to incorporate subjective opinions. Although, it seems suitable to be used in software engineering it should be adapted so that the subjective opinion is unbiased.

C. Synthesis and knowledge translation in software engineering

A study was conducted to evaluate how evidence is synthesized in software engineering research [10]. The findings of the study show that limited attention is paid to the synthesis of evidence in SLRs. While 41% of studies did not report following any synthesis methods, among the studies that reported using synthesis methods thematic analysis (22.6%) and narrative synthesis (16.1%) were most used [10].

Knowledge translation is not widely practised in software engineering. According to a tertiary study [11], among the 143 SLRs reviewed, only few studies provided recommendations [2]. However, the recommendations were produced by experts and not by incorporating the subjective opinions of potential users [2]. In addition, the recommendations were generated in an ad-hoc manner without following a systematic method or guidelines. Although, the knowledge translation activity is discussed and defined in software engineering, there are no methods or guidelines proposed for undertaking knowledge translation in software engineering [1] and [2].

D. Bayesian in software engineering

Even though Bayesian theory is not yet used for synthesizing evidence in software engineering research, Bayesian networks have been applied to address various software engineering research problems. Approximately 72% of Bayesian networks applications are in the software quality (46.15%) and software engineering management (26.5%) areas [12]. The use of Bayesian networks for evidence-based decision-making in software engineering has been discussed in [12]. Three Bayesian network models are proposed to predict software reliability [12]. The study also claims that the use of Bayesian networks is not well recognized in software engineering research as compared to other disciplines such as health research

[12]. Even though Bayesian networks are used for decision-making in software engineering, it has been used to analyze data from single studies. For example, Bayesian networks are used to represent the software life cycle phases by incorporating expert judgment (collected from qualitative surveys) into quantitative data collected from software repositories [12]. The use of Bayesian synthesis to synthesize evidence from multiple studies has not been implemented yet.

III. Bayesian synthesis for knowledge translation - Method Description

In this section, we describe how Bayesian synthesis can be used for synthesizing evidence and knowledge translation. It consists of three main steps as depicted in Figure 1.

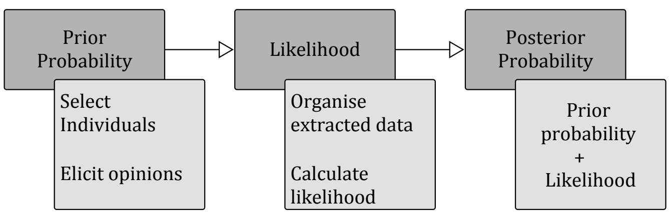

Fig. 1. Three steps of Bayesian synthesis

The Bayesian synthesis starts with prior probability which consists of subjective opinions based on personal experiences on a particular topic. Thereafter, the extracted data is summarized to understand what is known or what is already been studied about the topic. Likelihood is the probability that the extracted data is true. Now that the prior probability and likelihood are known, the posterior probability is formulated by refining the prior probability given the likelihood. The steps involved in Bayesian synthesis are described in following sections.

A. Step 1: The prior probability

This step includes the following two sub-steps:

1) Selecting individuals: Sampling of individuals.
2) Eliciting opinions: Capturing subjective opinions.

The advantage of Bayesian synthesis is that it allows incorporating subjective opinions and beliefs into the synthesis. In this way, the synthesis is not limited to the observed or collected data. Before using the data, the subjective opinions are captured.

Selecting individuals: In software engineering, the subjective beliefs of practitioners, decision-makers/policy-makers are relevant. The selection of individuals who will be the users' of the knowledge is important. For example, if a decision needs to be made in a software project then, all the practitioner roles that should be involved in making the decision should be selected to elicit their subjective opinions.

Elicit opinions: Opinions are elicited to collect prior probability. Prior probability can be captured in terms of percentages ranging from 0 to 100 % or in terms of absence/presence (0/1) of a parameter value. One of the advantages of Bayesian synthesis is that it is flexible. Hence, the prior probability can be formulated in a way that suits the research objective. Spiegelhalter et al. [6] state that there is no "correct" prior and that Bayesian synthesis should be seen as a means of transforming prior into posterior opinions, rather than producing the posterior probabilities. The subjective opinions form the prior probabilities and are represented as P(parameters).

B. Step 2: The likelihood

Likelihood is the representation of what is known. In other words, it is the summary of all the research studies within a specific research objective. For example likelihood is the outcome of an SLR.

This step includes the following two sub-steps:

1) Organise extracted data: Summarize relevant data from the primary studies.
2) Likelihood calculation: Calculate the likelihood of the data for the given parameters.

Organise extracted data: The extracted data can be summarized in a table format where, each extracted parameter is the column name and each row represents the results (parameter value) from a single primary study. The parameter values entered in the table depends on the research objective and the available data. For example, if the research objective is to find factors that affect the decision. Then, the factors from the primary studies should be entered in the table. The studies that report the factor can be represented as 1 and the studies that do not report the factor can be represented as 0. However, if the research objective is to identify the factors that affect the decision positively and the factors that affect negatively or have no effect. Then the positive effect can be represented as 1, negative as 0 and no effect as 0.5. The parameter values entered in the table also depends on the data that is available. If the primary studies include statistical analysis such as odds ratio of the factors then odds ratios are entered that provide the magnitude of the factors and not just the presence or absence of the effect. Similarly for quantitative (experimental) studies, the effect sizes must be entered.

The extracted data can be organized based on the evidence provided to support the results or based on the data type. The division of extracted data depends on what is extracted from the primary studies. If the extracted data is of different data types, then it is recommended to divide the likelihood based on the data type, i.e qualitative and quantitative. It will allow to aggregate similar data together. However, if the extracted data from all studies is of same data type then it is recommended to divide the likelihood based on the evidence supporting the results. This will allow to interpret the likelihood based on the evidence provided in the primary studies.

The difference between empirical and non-empirical studies might be more relevant to analyse in software engineering. Empirical studies might have more importance as they are based on empirical evidence. Thus, we suggest dividing studies into non-empirical and empirical studies in software engineering instead of qualitative and quantitative studies. However, as Bayesian synthesis is flexible, it allows to separate the likelihood based on qualitative and quantitative studies or empirical and non-empirical studies as illustrated in Figure 2.

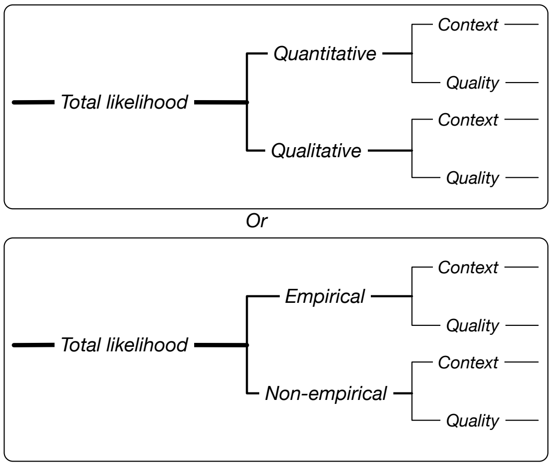

Fig. 2. Division of likelihood calculation

Likelihood calculation: The likelihood is written as the function of the observed data for the given parameters represented as P(data—parameters). In other words, the proportion of studies reporting the parameter to be true. For example, if 5 out of 10 studies report the parameter then, it is 50% ([5/10]*100) likely that the observed data is true to the population. Higher values of likelihood indicates that the observed data is more likely to occur.

The likelihood calculation can be divided for better interpretation and analysis of the observed data as shown in Figure 2. Based on how the extracted data is divided, the likelihood can be calculated for each division. There is further diversity in the results with respect to the context and the quality in the way the study has been conducted. The importance of reporting context details in software engineering research has been identified [13] and [14]. Hence, it is advisable to use the context information and separate the likelihood based on the context and quality assessment. Such representation of likelihood provides a detailed analysis of the collected data and provides a richer input to the next step which is discussed in Section III-C.

C. Step 3: The posterior probabilities - refining prior probabilities

Posterior probability is the refinement of the prior probability given the likelihood. The equation of posterior probability is stated in [7] as shown in Equation 1.

$$P(parameters|data) = \frac{P(data|parameters)P(parameters)}{P(data)} \quad (1)$$

Where, P(data|parameters) is the likelihood and P(parameters) is the prior probabilities.

However in our approach we do not follow a mathematical approach in the calculation of posterior probability as it may ignore some of the interpretations. For example, practitioners might want to refine the probabilities according to the context and the quality assessment of the primary studies and their individual opinions. Therefore, the practitioners who state the prior probability will refine their probabilities based on their interpretation of the observed data and likelihood calculation.

The posterior probability is the combination of prior probability and likelihood. The posterior probabilities are formulated stepwise by refining the prior probability given the likelihood. The stepwise formulation is based on how the likelihood calculation is separated. In the first step, the prior probability is refined when the likelihood of the first division of papers is available and in the next step, when the likelihood of the second division of papers is available. This final probability is regarded as the posterior probability. The prior probability and likelihood can be combined independently by each individual or collectively by discussing the differences in prior probability and a common interpretation of likelihood.

If the posterior probability is to be formulated with a common prior probability and common understanding of likelihood then, internal and external conflicts should be resolved. Internal conflicts refer to the conflicts between the individual opinions and external conflicts refer to the conflicts in interpretation of likelihood.

Discussing internal conflicts: Once the prior probability of each individual is known, the individuals discuss the differences in the probabilities and try to resolve conflicts. The differences in the probabilities could be due to the differences in the individual roles. In some cases it is important to resolve the differences rather than proceeding with the different probabilities. For example, a developer might not see the same factors to be important in the decision as the architect. The developer might not be aware or might not have encountered similar experience as the architect. The differences in the probabilities are discussed until they reach consensus. If the differences are due to lack of knowledge then the individual probabilities should be refined.

Spiegelhalter et al. have summarized the different strategies to refine probabilities from different individuals [6]:

1) Elicit a consensus: The diverse probabilities of all the individuals are brought into consensus using either informal or formal Delphi methods. During the process of eliciting consensus all individuals should be given equal importance. This way the dominant individuals' influence will not impact opinions of others. Once the prior probabilities are refined and are similar, an average of the probabilities is taken.
2) Calculate a pooled prior: A simple average of all individual prior probabilities is taken. In this case the individual prior probabilities are not similar before taking the average. If the prior probabilities represent the absence or presence of a factor then the proportions of the practitioners mentioning the presence or absence is calculated. For example, if 3/6 practitioners mention that the presence of factor X affects adherence then the combined probability is 50%.

Discussing external conflicts: The individuals who assigned the prior probabilities discuss how to interpret the extracted data and likelihood calculation. Particularly, how much importance should they give to the extracted data and likelihood calculation and how they should refine their probabilities. The context and quality are also taken into consideration. For example, the individuals might decide to only consider the high quality studies from a particular domain. In this way the external conflicts based on the interpretation of likelihood are resolved.

We propose four approaches to discuss internal and external conflicts as follows:

1) Discuss internal and external conflicts
2) Discuss only internal conflicts
3) Discuss only external conflicts
4) Discuss only at the end

The working of the four approaches is depicted in Figure 3.

Steps

Step 1: Prior probability
Select individuals
Elicit opinions
Step 2.1: Elicit opinions or calculate pooled prior probability
Step 2.2: Retain individual prior probabilities
Step 3: Likelihood
Organise extracted data
Calculate likelihood
Step 4: Discuss extracted data and likelihood calculation
Step 5: Formulate posterior probability
Step 6: Elicit opinions or calculate pooled posterior probability

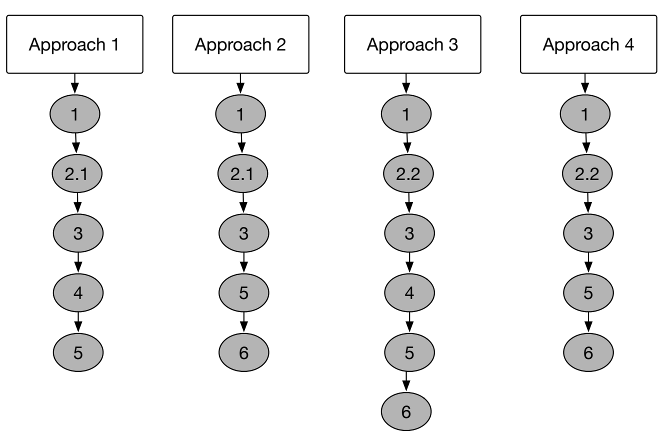

Fig. 3. Four different approaches to refine prior probabilities

Approach 1: Discuss internal and external conflicts: In this approach, both the prior probabilities and likelihood are discussed as part of progressing through the approach. The individual prior probabilities are combined by either eliciting consensus or calculating pooled prior probability. The common prior probability should be updated based on common understanding of the likelihood.

Approach 2: Discuss only internal conflicts: In this approach the prior probabilities are combined by either eliciting consensus or by calculating pooled prior probability. However, the likelihood is not discussed and the individuals do not have a common interpretation of the likelihood. The common prior probability should be updated based on individual understanding of the likelihood. Therefore, each individual formulates posterior probability by refining the common prior probability based on their individual understanding of the likelihood. This approach can be adopted when the differences in interpreting the evidence are important.

Approach 3: Discuss only external conflicts: In this approach the prior probabilities are neither discussed nor combined. In other words, the diversity of opinions is retained. Only the likelihood is discussed, i.e. all the individuals would have a common understanding on how to interpret the collected data and update their individual probabilities. Therefore, each individual formulates posterior probability by refining their individual prior probability based on the common understanding of the likelihood. One possible reason to not discuss or combine probabilities could be that all the practitioners have the same role and there are no confounding factors affecting the opinions. In this case the differences become relevant to capture.

Approach 4: Discuss only at the end: In this approach neither the prior probabilities nor the collected data and likelihood calculation are discussed. The individuals independently refine their probability without discussing their individual opinions and the likelihood resulting in individual posterior probabilities.

Based on the approach used to resolve conflicts, the posterior probability is formulated by refining either the pooled or individual prior probability based on the common or individual understanding of likelihood. In approach 1 a common posterior probability is formulated. However, in approaches 2, 3, and 4 individual posterior probabilities are formulated. The individual posterior probabilities are then combined by discussing the differences by eliciting consensus or by taking an average.

IV. Illustration

In this section we present an example to show the working of Bayesian synthesis for knowledge translation. In this example, the posterior probabilities are computed using approach 1 (described in Section III-C). The research problem is the decision to choose between in-house development and acquiring OSS components for building software systems. The objective is to identify the factors that impact the decision.

A. Step 1: Prior probability

Selecting individuals: As the research problem is related to decision-making in software engineering practice, the decision-makers' experiences and opinions in practice are relevant. However, in this example the prior probability values are hypothesized values provided by the authors of this paper for illustration purpose. Thus, the authors are acting as practitioners/decision-makers. In a real context, actual decision-makers' experiences and opinions should be considered.

Eliciting opinions: The decision-makers provide the factors that they think impact the decision to choose between in-house development and OSS. Table I represents the probabilities assigned by the practitioners (in this case, the authors). The probabilities assigned by the decision-makers are independent and are solely based on their personal experience and opinion without any information from the scientific research. If the decision-makers state that the factor is important in the decision, then, the value 1 is assigned. The value 0 indicates that the decision-maker has not mentioned the factor being important. Since the decision-makers only indicate if they think a factor is important of not, the values are in binary form i.e. either 1 or 0. Depending on the role (practitioner's perspective) some of the factors may or may not be considered as important factors.

TABLE I
PRIOR PROBABILITIES - DECISION-MAKERS' EXPERIENCE/OPINION

decision-maker Role	**Time**	**Cost**	**Effort**	**Quality**
Manager	1	1	0	0
Developer	0	0	1	0
Architect	1	1	0	1
Integrator	1	1	1	1
Tester	0	1	0	1

B. Step 2: Likelihood

Likelihood is the representation of what is known, i.e. the representation of the evidence from the literature. The primary studies mentioned is this example are a subset of the primary studies considered in an SLR conducted previously on a related topic [15]. Note that the outcome of the SLR reported in [15] is not translated into recommendations. However, it is regarded as future work to translate knowledge into recommendations for practitioners using Bayesian synthesis for knowledge translation.

Organise extracted data: The extracted data regarding the factors that impact the adoption decision are organised as shown in Table II. The primary studies consist of empirical and non-empirical studies. The non-empirical studies are mostly opinion, experience or philosophical papers (based on classification proposed be Wieringa et al. [16]). Personal opinions, views or experience are more focused on individual researchers or is project specific. Whereas empirical studies such as case studies and surveys are more generalized and are based on stronger evidence. The value "1" in Table II represents that the factor has been mentioned in a primary study, value "0" represents absence of the factor. Since the data extracted from all the studies is of same data type, we organized the extracted data based on empirical and non-empirical research types. In the empirical studies we found that some of the factors were mentioned but the conclusion was that the factor did not have any significant effect. We still record this evidence as the factor has been mentioned as they are validating a possible myth. Hence this is important to be considered in the synthesis. Such factors that are mentioned as not having any effect are assigned the value 0.5. Three new factors are identified by the papers which were not identified by the practitioners in step 1.

TABLE II
ORGANISATION OF EXTRACTED DATA FROM NON-EMPIRICAL AND EMPIRICAL STUDIES

Ref.	**T**	**C**	**E**	**Q**	**TS**	**L**	**R**
Non-empirical studies							
[17]	0	1	0	1	1	1	1
[18]	0	0	0	0	1	1	1
[19]	0	0	0	0	1	1	0
Empirical Papers							
[20]	0.5	1	1	0	1	0	1
[21]	0	1	0	1	0	1	0
[22]	0	1	0	0	0	0	0

T: Time, C: Cost, E: Effort, Q: Quality
TS: Technical support, L: License, R: Requirements

Likelihood calculation based on empirical and non-empirical studies: In this illustration the likelihood calculation is divided into empirical and non-empirical studies. The likelihood is the proportions of the primary studies reporting the factor being important for the decision. The context and quality of the primary studies are as mentioned in Table III. The likelihood for each factor is calculated by considering the percentage of primary studies that have mentioned the factor as an important factor in making adoption decisions. For example, four primary studies have indicated that the technical support (TS) factor is an important factor. Hence, the likelihood for technical factor (TS) equates to 67% ([4/6]*100). As seen in Table IV, the first row represents the total likelihood and the likelihood is separately calculated for empirical and non-empirical studies which is represented in the following two rows. The total likelihood of cost (C), technical support (TS) and license (L) is the same. However, when the likelihood is separated based on evidence, we see that all the empirical studies report cost being important and all the non-empirical studies report technical support and license being important. Based on this separate calculation the total likelihood of 67% may be interpreted differently based on the evidence supporting the factors.

It is recommended that the likelihood is further divided based on the context and quality, this can be done by the rigor and relevance assessment method proposed by Ivarsson and Gorschek [23] during data extraction while conducting SLRs. However as seen in Table III, at most one paper is supporting the same domain hence, the likelihood is not divided further based on context. In addition, since the quality of all empirical studies is high and non-empirical studies is low, the likelihood calculation would be as listed in Table IV. However, in a real case contextual factors may be viewed as very important when rating the outcome from different studies and hence, having a big impact on the likelihood.

TABLE III
CONTEXT AND QUALITY OF PRIMARY STUDIES

Ref.	Context	Quality
[17]	Mission-critical domain	Low
[18]	Not mentioned	Low
[19]	Not mentioned	Low
[20]	Telecommunication domain	High
[21]	Not mentioned	High
[22]	Multi domain	High

TABLE IV
LIKELIHOOD BASED ON EMPIRICAL AND NON-EMPIRICAL STUDIES

	T	C	E	Q	TS	L	R
Total likelihood	8%	67%	17%	33%	67%	67%	50%
Non-empirical likelihood	0%	33%	0%	33%	100%	100%	66%
Empirical likelihood	16%	100%	33%	33%	33%	33%	33%

T: Time, C: Cost, E: Effort, Q: Quality
TS: Technical support, L: License, R: Requirements

C. Step 3: Posterior probability - Refining prior probability

We have the probabilities of the decision-makers as shown in Table I. Following approach 1, the differences in the probabilities of the decision-makers should be resolved before continuing. Hence, a pooled prior is calculated by considering the percentage of decision-makers that have mentioned the factor as an important factor in making adoption decisions. For example, as seen in Table I, three decision-makers have indicated that time is an important factor. Hence, the prior probability for time (T) equates to 60% ([3/5]*100). The first row (Prior probability) in Table V represents the pooled prior probabilities computed from values in Table I. The second and forth rows (Non-empirical likelihood and Empirical likelihood) are the same as the second and third rows of Table IV. The third and fifth rows (Refined probability and Posterior probability) are new rows related to the formulation of posterior probabilities which are discussed below.

We have the probabilities of the decision-makers and likelihood from the non-empirical papers and empirical studies. The likelihood of empirical and non-empirical studies (Table IV) is provided to the decision-makers along with the context and quality of the papers (Table III). Once the decision-makers receive this information, they either decide to update (either increase or reduce) their prior probability or stay with their initial prior probability. The prior probabilities are updated step-wise, once after the data from non-empirical studies is available (third row in Table V - Refined probability) and later when data from the empirical studies is available (fifth row in Table V - Posterior probability). It depends on how the decision-makers react based on the additional information provided. Not all decision-makers will have the same interpretation of the likelihood (data from empirical and non-empirical studies). As we use approach 1 to formulate posterior probability, any conflicting interpretations should be resolved until consensus is achieved.

For example, one of the decision-maker might decide to lower the probability of cost (third column in Table V - C) based on the non-empirical likelihood (second row in Table V - Non-empirical likelihood) since only 1/3 (33%) non-empirical studies (paper [17] as seen in Table II) has reported this factor. However, other decision-makers might not change the probability as they decide to give more importance to paper [17] as it is in the same domain (Table III) as the decision-makers. Hence, the decision-makers discuss if they should consider the overall likelihood of the effort factor (C) or only focus on the studies that are within the same domain. The decision-makers collectively decide not to lower the initial prior probability. Hence, the refined probability (third row in Table V - Refined probability) is same as the first row in Table V - (Prior probability). In the next step when the decision-makers look at the likelihood from empirical studies (forth row in Table V - Empirical likelihood), all the empirical studies have reported cost as an important factor. Hence the decision-makers collectively decide to increase the probability as shown in the fifth row in Table V - Posterior probability. The knowledge and experience of decision-makers should facilitate in the interpretation of the data. If consensus is not achieved, the average of the probability should be considered. The refined probabilities are as shown in Table V. Note that in this illustration the authors refined the probabilities however, in a real context the decision-makers should refine the prior probability. The arrows indicate the change from prior probabilities, i.e. either the prior probability is increased or decreased.

TABLE V
REVISED PROBABILITIES

	T	C	E	Q	TS	L	R
Prior probability	60%	80%	40%	60%	NA	NA	NA
Non-empirical likelihood	0%	33%	0%	33%	100%	100%	66%
Refined probability	50% ↓	80%	40%	50% ↓	100%	100%	66%
Empirical likelihood	16%	100%	33%	33%	33%	33%	33%
Posterior probability	20% ↓	90% ↑	40%	30% ↓	75% ↑	75% ↑	60% ↑

T: Time, C: Cost, E: Effort, Q: Quality
TS: Technical support, L: License, R: Requirements

Table V highlights the importance of including non-empirical studies in the synthesis. If non-empirical studies were not included then the factors: technical support (TS), license (L) and requirements (R) would not have been considered as important in the decision-making process. If the decision-makers took the decision without considering the evidence then they would have only considered four out of seven factors. And only one among the four factors: i.e. cost (C) has high probability. The three new factors have high probability which means that if they were ignored it would have a significant impact on the decision. The non-empirical studies have the most impact on the high probabilities, hence, it indicates the importance of non-empirical studies. Non-empirical studies are criticized as they lack rigor and relevance. However, since decision-makers themselves are involved in the synthesis they can validate the information from the non-empirical studies. Involving the decisions-makers in the synthesis allows to interpret the synthesized data (SLR outcomes) in the application context and facilitates in providing recommendations for adapting the outcomes in practice. The examples of recommendations that can be provided to the decision-makers are as follows:

- Time is not as important as it is perceived by decision-makers. As likelihood suggests [20] that the time saved in developing in-house might not result in overall reduction in time as the selection and integration of OSS consumes the time saved by not developing in-house. The decision-makers accept and agree to the likelihood hence, we can say that time is not an important criteria for the decision.
- External factors such as technical support provided and license obligations are important decision criteria. Initially the decision-makers do not mention external factors as they might be overseen. However, once they receive the likelihood, they agree that external factors could potentially influence the decision.

Depending on the quality of the primary studies supporting the evidence and the experience of the involved practitioners, the recommendation can be regarded as strong or weak.

V. Discussion

Bayesian synthesis takes into account a wide range of evidence, including subjective opinions into the synthesis. None of the synthesis methods used in software engineering research allows to incorporate subjective opinions. Bayesian synthesis is flexible and works well with other synthesis methods such as thematic analysis. The novelty of Bayesian synthesis is in the extension of traditional synthesis methods to support knowledge translation.

Bayesian synthesis gives the probabilities of the data being true as compared to inferential statistics that provides the probability of the calculation of the result being true. In real life we are more close to Bayesian thinking. For example, we think about the probability of an event to be true instead of the probability of the computation being true.

Bayesian meta-analysis has been implemented in health research [8]. The Bayesian synthesis method proposed in this paper is inspired by the Bayesian meta-analysis approach. The Bayesian meta-analysis method considers the subjective opinions and qualitative evidence together in the prior probabilities. In order words, the individuals know the information from qualitative studies before assigning prior probabilities. However, as we want unbiased opinions, we separate the qualitative evidence from the prior probability. The previous methods implemented in health research do not differentiate between the types of evidence [7], [9] and [8]. It leaves it up to the researchers/practitioners to decide how they want to evaluate the evidence. However, we recommend providing a detailed description in terms of summaries, context and quality information. This guides the researchers/practitioners to make informed decisions. Unlike previous Bayesian methods, no mathematical equations are used to compute the posterior probability. Instead, the individuals (researchers/practitioners) themselves refine the probabilities based on their experience. The four approaches for formulating posterior probabilities of software engineering research is also another novel contribution of the Bayesian synthesis proposed in this paper.

VI. Conclusions

We conclude by discussing the contributions and limitations of Bayesian synthesis. The synthesis methods analyze the results from multiple studies and supports knowledge translation. Bayesian synthesis goes beyond synthesis of only research evidence by actually guiding how the results can be applied or used in a particular context.

SLR is one of the research methods where multiple studies are synthesized. Often SLRs follow a rigorous approach to search, select and extract information from the primary studies. However, if the primary studies are weak or lack necessary descriptions then, it is difficult to make conclusions on a phenomenon and provide recommendations based on the outcome of an SLR. However, Bayesian synthesis, due to its ability to incorporate subjective opinions, is particularly useful when the data from primary studies is insufficient to draw generally valid conclusions [5]. Bayesian synthesis can be regarded as a knowledge translation process to interpret and translate outcomes of SLRs, particularly when there are few primary studies or when there are differences in findings in the primary studies. In addition, Bayesian synthesis can be particularly useful for research problems related to decision-making where individual judgements and opinion can be combined with research evidence thereby making evidence-informed decisions. Developing detailed guidelines and evaluating Bayesian synthesis in practice (real context) is considered as future work.

Limitations: Capturing subjective opinions in the analysis allows for better synthesis. However, it is a known fact that people are not good probability accessors, validity threats in eliciting opinions have been summarized by Kandane and Wolfson [24]:

1) Availability: Recent or easily recalled events might be given higher probability, and vice versa.
2) Adjustment and anchoring: The opinions are anchored at some starting point and tend to exert an inertia. The subsequent opinions might not be adjusted sufficiently and might be too close to the first probability.
3) Conjunction fallacy: A higher probability might be assigned to a parameter that is a subset of a parameter that has lower probability.
4) Hindsight bias: The opinion might be biased if the data is available before elicitation of the opinions.

These validity threats can be mitigated by using elicitation techniques such as interactive feedback with a structured interview. In case of any issues such as conjunction fallacy, the person providing the probability can be asked to reflect more on the probability. Also by eliciting prior probability before providing evidence from qualitative and quantitative studies, unbiased probabilities can be obtained.

In this paper, we have described the use of Bayesian synthesis for knowledge translation in software engineering. The use of Bayesian synthesis is illustrated using an example to illustrate the working and flexible use of the Bayesian synthesis for knowledge translation.

REFERENCES

[1] B. A. Kitchenham, D. Budgen, and P. Brereton, *Evidence-Based Software Engineering and Systematic Reviews*. CRC Press, 2015, vol. 4.

[2] D. Budgen, B. Kitchenham, and P. Brereton, "The case for knowledge translation," in *Empirical Software Engineering and Measurement, 2013 ACM/IEEE International Symposium on*. IEEE, 2013, pp. 263–266.

[3] T. Greenhalgh and S. Wieringa, "Is it time to drop the knowledge translationmetaphor? a critical literature review," *Journal of the Royal Society of Medicine*, vol. 104, no. 12, pp. 501–509, 2011.

[4] M. Dixon-Woods, S. Agarwal, D. Jones, B. Young, and A. Sutton, "Synthesising qualitative and quantitative evidence: a review of possible methods," *Journal of health services research & policy*, vol. 10, no. 1, pp. 45–53B, 2005.

[5] C. Pope, N. Mays, and J. Popay, "Synthesising qualitative and quantitative health evidence: A guide to methods: A guide to methods," 2007.

[6] D. J. Spiegelhalter, K. R. Abrams, and J. P. Myles, *Bayesian approaches to clinical trials and health-care evaluation*. John Wiley & Sons, 2004, vol. 13.

[7] J. L. Crandell, C. I. Voils, and M. Sandelowski, "Bayesian approaches to the synthesis of qualitative and quantitative research findings," *Synthesizing Qualitative Research: Choosing the Right Approach*, pp. 137–159, 2012.

[8] K. A. Roberts, M. Dixon-Woods, R. Fitzpatrick, K. R. Abrams, and D. R. Jones, "Factors affecting uptake of childhood immunisation: a bayesian synthesis of qualitative and quantitative evidence," *The Lancet*, vol. 360, no. 9345, pp. 1596–1599, 2002.

[9] C. Voils, V. Hasselblad, J. Crandell, Y. Chang, E. Lee, and M. Sandelowski, "A Bayesian method for the synthesis of evidence from qualitative and quantitative reports: the example of antiretroviral medication adherence," *Journal of health services research & policy*, vol. 14, no. 4, pp. 226–233, 2009.

[10] D. S. Cruzes and T. Dybå, "Research synthesis in software engineering: A tertiary study," *Information and Software Technology*, vol. 53, no. 5, pp. 440–455, 2011.

[11] D. Budgen, S. Drummond, P. Brereton, and N. Holland, "What scope is there for adopting evidence-informed teaching in se?" in *Proceedings of the 34th International Conference on Software Engineering*. IEEE Press, 2012, pp. 1205–1214.

[12] A. T. Misirli and A. B. Bener, "Bayesian networks for evidence-based decision-making in software engineering," *Software Engineering, IEEE Transactions on*, vol. 40, no. 6, pp. 533–554, 2014.

[13] T. Dybå, B. Kitchenham, and M. Jørgensen, "Evidence-based software engineering for practitioners," *Software, IEEE*, vol. 22, no. 1, pp. 58–65, 2005.

[14] K. Petersen and C. Wohlin, "Context in industrial software engineering research," in *Proceedings of the 3rd International Symposium on Empirical Software Engineering and Measurement*. IEEE Computer Society, 2009, pp. 401–404.

[15] D. Badampudi and C. Wohlin, "Software component decision-making: In-house, open source, cots or outsourcing - a systematic literature review." *in revision after submission for Journal of Systems and Software*, 2015.

[16] R. Wieringa, N. A. M. Maiden, N. R. Mead, and C. Rolland, "Requirements engineering paper classification and evaluation criteria: a proposal and a discussion," *Requirement Engineering*, vol. 11, no. 1, pp. 102–107, 2006.

[17] J. S. Norris and P.-H. Kamp, "Mission-critical development with open source software: Lessons learned," *IEEE Software*, vol. 21, no. 1, pp. 42–49, Jan. 2004. [Online]. Available: http://dx.doi.org/10.1109/MS.2004.1259211

[18] R. Torres, "Developer-led adoption of open source software libraries: A conceptual model," 2012.

[19] J. Rudzki, K. Kiviluoma, T. Poikonen, and I. Hammouda, "Evaluating quality of open source components for reuse-intensive commercial solutions," in *Software Engineering and Advanced Applications, 2009. SEAA'09. 35th Euromicro Conference on*. IEEE, 2009, pp. 11–19.

[20] T. Helokunnas, "The dimensions of embedded cots and oss software component integration," in *Product Focused Software Process Improvement*. Springer, 2002, pp. 509–518.

[21] K. J. Stewart, A. P. Ammeter, and L. M. Maruping, "A preliminary analysis of the influences of licensing and organizational sponsorship on success in open source projects," in *System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on System Sciences*. IEEE, 2005, pp. 197c–197c.

[22] W. Chen, J. Li, J. Ma, R. Conradi, J. Ji, and C. Liu, "An empirical study on software development with open source components in the chinese software industry," *Software Process: Improvement and Practice*, vol. 13, no. 1, pp. 89–100, 2008.

[23] M. Ivarsson and T. Gorschek, "A method for evaluating rigor and industrial relevance of technology evaluations," *Empirical Software Engineering*, vol. 16, no. 3, pp. 365–395, 2011.

[24] J. B. Kadane and L. J. Wolfson, "Experiences in elicitation," *The Statistician*, pp. 3–19, 1998.