

Alice Ashcroft (2022): I think "Hedging" could be a Feminist Issue in Software Engineering. In: Proceedings of the 20th European Conference on Computer-Supported Cooperative Work: The International Venue on Practice-centred Computing on the Design of Cooperation Technologies - Exploratory Papers, Reports of the European Society for Socially Embedded Technologies (ISSN 2510-2591), DOI: 10.48340/ecscw2022_ep01

I think "Hedging" could be a Feminist Issue in Software Engineering

Alice Ashcroft

School of Computing and Communication, Lancaster University

Contact Author: alice.ashcroft@lancaster.ac.uk

Abstract. When it comes to software engineering and the development life-cycle, there are a number of opportunities for under-represented groups, gender being the focus of this paper, for decisions to be affected by language. Considering existing linguistics research surrounding gendered language, specifically "Hedging", alongside various stages of the development life-cycle, this paper poses that "Hedging" should be seen as a feminist issue in software engineering, and presents five areas for further research to uncover the potential negative effects it is having, and what can be done to mitigate these. This paper focuses on the subtleties in conversation, and how conversation takes place, building on Feminist Conversation Analysis, Feminist Methodologies, and Software Engineering Methodologies.

1 Introduction

The effect that gender, language and interaction can have on Computer Science (CS) cannot be understated. From initial meetings where ideas are brought forward, to design processes, to the systems being built, to the impact this has on users; each of these stages, and the rest, rely heavily on communication. Sometimes, these interactions are textual, frequently face to face, and more and more since the Covid-19 pandemic began, through video calls.

Language, outside of CS has long been studied, particularly when it comes to the effects of gender¹. However, when looking at the direct impact this has on the software being built, the findings of this are much more recent and have only since been recognised as holding some importance. Whilst the effect of communication on Software Engineering has begun to be explored (Catolino et al., 2019), the subsection of gendered language which has yet to be examined in much detail, and is therefore reflected upon in this exploratory paper, is that of "**Hedging**". "Hedging" is short for "*hedging your bets*" and is often displayed through the use of phrases such as, "*I think*", "*you know*" and "*sort of*" (Holmes, 1986; Tannen, 1994; Murphy, 2010; Ashcroft, 2020b, 2021). There are, of course, many other gender differences in language (as outlined in Section 3.2), but the focus of this paper will be Hedging, due to the potential impact for miscommunication it may cause.

It is also important to consider these through the lens of Feminist Methodologies (discussed further in Section 2), and how these are already applied within CS, and how an application of these can contribute to the overall knowledge within the field, and in this particular instance the Software Development Life-cycle.

Therefore, this paper will focus on the Software Development Life-cycle, also known as the Software Process, as outlined by Sommerville (2016), and discusses how discourse, interaction and language have the potential to impact decisions and processes throughout. However, in order to do this, there first needs to be an understanding of Feminism in CS Methodologies (Section 2) and following this, the impact gender can have on language (Section 3).

2 Feminism in Computer Science Methodologies

It could be argued that Feminism in Computer Science and Feminist Methodologies are simply good research practices, however, Feminist Methodologies operate under the assumption that any research used as a basis for understanding may contain unconscious patriarchal bias (O'Leary, 2017). This assumption is supported by Davis (2019), who argues that diversity is "not only essential as a goal of intersectional feminism², but also as a means of continuing success for digital movements." Therefore, in order to overcome any existing bias in research, any research carried out going forward should not rely on past work, although many researchers would agree that this is rarely possible.

¹ Throughout this paper, the emphasis is on gender, and not sex, however much existing literature does not differentiate between the two. Where possible the terms "men", "women" and "gender" have been used, instead of "male", "female" and "sex", however, the cases where they are used are done so to correctly reflect previous work. This author also rejects the idea that gender is a binary, despite the fact that most existing literature refers to it as such. Again, literature will be referenced accurately, but this should not undermine the need for more extensive research into gender beyond the binary and these issues.

² Although linguistically, intersectionality may simply mean the cross over of multiple identities, when it comes to theory, this is a product of Black Feminist Theory, and therefore this should be recognised and credited, otherwise this risks the loss of credit from already underrepresented groups (De Hertogh et al., 2019).

Furthermore, Schlesinger et al. (2017) introduce intersectionality as a framework for research, recognising it as the overlap of attributes making up an individual's identity. They carried out further research into the intersectionality of HCI research by bringing together a keyword set and "included terms that describe gender, ethnicity, race, sexuality or class" and found that only 1% of all CHI publications included these keywords, showing that work has yet to be done in this area (Schlesinger et al., 2017). This is also supported by Bellini et al. (2018) who state that "an inclusive community must be built that connects research from across different areas within HCI, across academic generations, and across other tangible and intangible borders", an idea that was previously discussed at the EUSSET Colloquium on "CSCW Theories and Concepts" in January of 2022 (eus, 2022) in that research must be carried out regarding diversity, but it must also be part of other research where gender is not the focus.

In addition to this, when it comes to Feminist Epistemology, "there has been debate between feminists about whether there can be feminist epistemology" (Barbour, 2018), but if epistemology refers to the theory of knowledge and understanding, and the subjects of the research "have gender", then the argument from feminists that "gender and individual identity are significant in the process of becoming a subject and a knower" (Flax, 1993; Barbour, 2018), then surely they must be relevant in social research? However when it comes to applying feminist epistemologies in CS, where often logic and structure are seen as important values, this may clash with the more general understanding of how knowledge is formed. As stated by Hancox-Li and Kumar (2021), "feminist epistemology has long taken a critical stance towards fully formalized systems, instead emphasizing the interactive nature of knowledge creation and the importance of exploring multiple possible meanings". This could be argued to be even more important to consider where gendered language is the topic of research, as it is in this paper, as the difference in meaning and understanding of both the language used and the research being carried out, must be held to the same standards.

Therefore, considering feminist theories, and feminist research, and how this can support further research into language and its impact on design, particularly within a field where logic is valued so highly, a closer examination of literature regarding CA and DA has been carried out (Section 3), in order to provide a framework for uncovering how "Hedging" can have an impact on design in CS (as discussed in Section 4).

3 Gender, Language and Interaction

Conversation can happen in a multitude of ways, but there are classic traits of conversation that have been observed and researched through both Conversation Analysis (CA) and Discourse Analysis (DA), as discussed in Section 3.1. These traits, like so many things, are affected by gender (see Section 3.2), and one of these traits is known as "**Hedging**" (see Section 3.3). This Section will explore the classic traits of conversation from the field of linguistics, how gender affects these,

and how "Hedging" is an example of how this can have an impact on the decisions made throughout conversation. Following this examination of literature regarding gendered language, and hedging, this understanding will then be taken into Section 4 and analysed alongside the Software Development Cycle.

3.1 Traits of Conversation

Traits of conversation, and discourse, as stated above, are well established outside of the field of CS. Stokoe (2018) is just one of many to look at these interactions and traits, particularly when it comes to gender, with others including Holmes (1986) and Boden (1994). Opening conversation, turn takings, gaps and overlaps as well as how the conversation ends, are how conversations take place (Stokoe, 2018) and the impact that gender can have on these parts of conversation can be affected by sex differences in language, gender itself, how problems may be raised, and "**Hedging**".

How a **conversation opens** can have a significant impact on how the rest of the conversation is likely to go (Stokoe, 2018). There is seemingly little research on how gender can affect this well-known phenomenon discussed often in CA (Svennevig, 2012). **Turn-taking**, however, can largely be affected by gender due to the effect power dynamics can play on this trait; which could be caused by any number of attributes including; experience in a role, race, gender, etc. Ford (2008) found that in academic meetings the majority of turn-taking was decided through non-verbal cues "such as leaning forward, gazing at the chair and raising a hand" (Svennevig, 2012). Turn-taking, of course, may be impacted by a large number of meetings now taking place online. It could be argued that having meetings online may allow for more equal participation, due to these physical barriers being removed, or perhaps the more outspoken members of meetings are more likely to speak and therefore be heard, due to the lack of, or perhaps increased, opportunity for overlap in conversation. Paulus et al. (2016) analysed literature relating to online conversation analysis before the pandemic in 2016 and relate this back to the work of Sacks et al. (1974) and whilst finding some deviations, find a number of similarities in how turn taking takes place online compared to in person.

On the topic of **gaps and overlaps** in conversation, both can have a serious effect on the conclusions reached in a conversation (Liddicoat, 2011), especially when there are more than two people in the conversation. Overlaps in conversation could be seen to have rather negative connotations, anecdotally it seems that nobody enjoys being spoken over, but Stokoe (2018) argues that overlap can be "an example of collaboration", which could be argued as being vital to any software development process.

How **conversation closes** is often the last step of any conversation (Liddicoat, 2011; Jay, 2009), and when it comes to meetings, is often concluded by a summary of actions or leanings. Therefore it is vital to have a "good conversation" or "good meeting", where everybody leaves with the same understanding of what the

conversation has achieved; be it the exchanging of pleasantries or the recording of actions.

3.2 Gender and Language

Gender and language is an established field, with many researchers using CA and DA to uncover gender differences (Stokoe and Smithson, 2001; Benwell, 2006; Friedrich and Heise, 2019). Research has also started to be done examining the effect that gender and language can have on User Experience (UX) and Software Design (Ashcroft, 2020a). Therefore, it is vital to consider the effect gender has on the conversations that take place when gender is not the topic of conversation.

Speer and Stokoe (2011) state that one consideration of gender and language is "**sex differences in language**", which focuses on the way men and women are represented in the language itself. For example, referring to a generic user as "*he*", as opposed to using gender-neutral pronouns such as "*they*". **How gender, and varying identities, are constructed** is another area that affects discourse (Speer and Stokoe, 2011).

How problems are raised will have an impact on how groups make decisions (Park, 1996), and this has also been linked to gender (Ashcroft, 2020a). This could also have a major effect on software development processes, if women in the meeting feel unable to raise problems, or if they do they are not heard.

Following from the traits of conversation discussed above (Section 3.1), how people **overlap** in conversation and **take turns**, can also be impacted by gender (Kitzinger, 2008). For example, the order of speakers may seemingly be chosen at random (e.g. going around a table), but this may sometimes be led by the men who are present (Ashcroft, 2020a). Therefore, it should be considered who is leading the discussion, which may again tie into the roles of power, and existing dynamics caused by roles, gender, race etc. - especially given that "men often achieve leadership roles regardless of past performance" (Reuben et al., 2012).

This paper poses that each of these existing areas for potential inequality will only be worsened by the effects that "Hedging" may have on conversation³.

3.3 What is "Hedging"?

As outlined above, "Hedging" is short for "hedging your bets" and includes the use of phrases such as, "I think", "you know" and "sort of" (Holmes, 1986; Tannen, 1994; Murphy, 2010; Ashcroft, 2020b, 2021). Murphy (2010) states that "the use of hedges among females before a key word" is used "to avoid the appearance of playing the expert", also known as "Expressing Uncertainty" (Ashcroft, 2020b). Nearly all researchers agree that "Hedging" will be used with different motivations

³ It could be argued, that due to the existing inequality in the perception of talk, that "Hedging" could be used as a tool to encourage ideas and thoughts to be heard; for example being used to persuade, or to make the words of those who are under-represented more "palatable" for others. Therefore, "Hedging" should be seen as a tool for conversation, not always a hindrance, which is discussed further in Section 3.3

and that context plays a part in understanding the reason for its use (Holmes, 1990). For example, Holmes (1986) states that "Hedging" can have two purposes, either to express speaker confidence e.g. "you know" or "reflecting uncertainty" e.g. "I think". Regardless of the motivation, "Hedging" is recognised as a "significant communicative resource for academics" (Hyland, 1996), understanding "Hedging", the motivations for its use and the impact it can have on conversation and decisions, is vital to understanding how decisions are made in the real world, and in the case of this paper, in design.

"Hedging" can have an impact on discussions that take place, because it is mainly rooted in the aim of downplaying ideas, thoughts or suggestions. For example in past research, focusing on design meetings, a woman said "some lecturers don't use the interactive screen, so, um, so *I think* you want more people to use it..." (Ashcroft, 2020b). This example could be seen as the expression of genuine uncertainty, or they could have been attempting to reduce their contribution for fear of rejection or taking up space, which is a known feminist issue, one that affects Black Women even more (Blackwell, 2020).

The impact this may have cannot be understated, not only will this have an impact on design decisions that are made, but could potentially disrupt the entire process and negate or downplay the opinions portrayed by women. However, it is important that there is no assumption that "Hedging" is itself negative, as well as being used to express uncertainty, it can also be used to persuade, with phrases such as "you know". Therefore, the topic that is focused on in this paper (see Section 4) is the perception of the words being spoken, regardless of the they *type* of "Hedging" being used, particularly when it comes to gender differences.

Dixon and Foster (1997) found that "Hedging" was used the same number of times by both genders but the reason *why* it was used differed based on gender, which contradicts what was found by Holmes (1986), who found that "Hedging" was used more by women. It could be argued that the frequency of use is irrelevant without understanding the reasons why it was used. However, there seems to be little research in which participants have been interviewed after a recording has taken place, in which their motivations are questioned, and all analysis seems to be inferred by the researcher's digression (Holmes, 1986; Ashcroft, 2020b). One of the main issues with this research approach, however, could be that participants are not sure of their reasons for saying it, as it may be instinctive or done without thinking. Therefore, for intent to be uncovered, this will require further research.

All this being said, it is clear that there is a correlation in existing research between gender, "Hedging", and the effect this has on a conversation. It is these parallels that will be discussed alongside the software development cycle in Section 4, in order to uncover its potential impact. All of this, of course, should then be tested through the recording and analysis of real design meetings, to ensure its validity, as discussed in Section 6.

4 The Potential Effects of "Hedging" on the Software Development Life-cycle

The Software Development Life-cycle (SDL) is a well-documented process (Sommerville, 2016; Leau et al., 2012), and whilst variations on this exist such as through Innovation Practices (Kic, 2021), Agile Methodologies (Agile Alliance, 2020) and the more traditional Waterfall (Dima and Maassen, 2018); the fundamental principles supporting this remain the same. These are as follows;

1. **"Software Specification:** The functionality of the software and constraints on its operation must be defined." (Sommerville, 2016)
2. **"Software Development:** The software to meet the specification must be produced." (Sommerville, 2016)
3. **"Software Validation:** The software must be validated to ensure that it does what the customer wants." (Sommerville, 2016)
4. **"Software Evolution:** The software must evolve to meet changing customer needs." (Sommerville, 2016)

There, of course, exists the process before Specification even begins, where ideas or problems are brought forward to begin with (see Section 4.1). It could be argued that this is the true start of the "pipeline". Defined as Ideation by Adobe (Kic, 2021), problem raising through other methods, or project initiation through project management; there are a number of ways in which potential new software ideas can be brought forwards. The need for software and how this is raised, alongside the four areas outlined above, shall be examined for conversational spaces that may be affected by "Hedging", its uses, and gender.

4.1 The Need for Software

How the need for software or digital intervention is raised will vary between organisations. Some smaller companies may have a more informal model, while other companies may have a systematic process that is followed and applied to all suggestions. More realistically, it could be argued, it is more likely for the truth to lie somewhere between the extremes.

4.1.1 Project proposals

Project proposals may be a way in which ideas are brought forward (Heemstra and Kusters, 2004). A good project proposal should be evidence-based and include due diligence in talking to stakeholders and users, to prove the need for itself. How "Hedging" affects this way of input is seemingly as yet unresearched. Although "Hedging" may or may not be explicit in proposals themselves, the presentations of initial ideas, or presenting the proposals themselves, may be affected by "Hedging", and therefore their likelihood to be accepted may also be affected.

4.1.2 Problem raising

Raising issues with processes, practices or day-to-day working is one of the most common ways in which projects can be raised or changes can occur (Park, 1996; Ashcroft, 2020a). How people are heard when raising problems though, is seemingly one that can quite significantly be affected by "Hedging". For example, if a person says they "think" something can be improved, and another says they "know" it can be improved - it could be argued that it is obvious who will be taken more seriously. This is important to consider as a feminist issue, as we know that men and women use "Hedging" differently (Holmes, 1986), and therefore this is clearly a gendered issue.

4.1.3 Ideation through innovation

Sometimes, one of the main motivators for change can be to utilise existing or upcoming technologies for a new purpose through innovation practices such as through Adobe's Kickbox (Kic, 2021). Innovation, using Adobe's Kickbox (Kic, 2021) involves a six-step process; inception, ideation, improvement, investigation, iteration and infiltration. Each of these stages mainly happens through discussion and therefore, will be affected by language, which may, in turn, be affected by gender. Of course, other frameworks for innovation exist, which opens the opportunity to see how each of these may vary when it comes to the impact that gendered language may have on them.

4.1.4 The effect of gender on the need for software

Regardless of how ideas are brought forward, be it through project proposals, problems being raised, or innovation, many of them are brought forward using some form of language, either digitally or in person. Therefore these are open to being affected by traits of language such as "Hedging", which we know are affected by gender. Therefore, it should be considered, that before the specification process of software creation even begins, that some ideas or changes may already have been lost due to a system that does not allow all members of society to be heard, listened to, and understood. This should be researched further in addition to the above alternative ways of bringing forward the need for software.

4.2 Software Specification

Software specification is the first stage of Software Engineering as defined by Sommerville (2016). How software specifications, or requirements, are gathered varies based on the selected methodology, as well as the organisation or individual in question who is carrying it out. However, most agree that it should include the involvement of all stakeholders, including users of the system. How these are documented can also vary, from requirements listing in Waterfall methodologies (Dima and Maassen, 2018), to User Stories in Agile (Agile Alliance, 2020).

Through both the gathering and recording of requirements, language is instrumental in how this takes place. If the stakeholder engagement is done through *written formats* such as emails, instant messaging or surveys, there are many ways in which language will affect the interaction. If they are done through *spoken interaction*, e.g. video calls, audio calls, or in-person meetings, this also leaves room for misinterpretation, the personal preferences of individuals overriding, or what is recorded as the correct process i.e. human error. People are not without their bias, however unconscious.

However, it could be argued that it is not just the perception of these interactions that will affect the specification stage, but how current issues or ideas are raised. If "Hedging" is used, then perhaps those who use it, often women (Holmes, 1986), may be less likely to be heard and understood. What should be researched further is a deeper understanding of the effect of "Hedging" on this stage, and if any negative causation is found, what can be done to mitigate the effects of this.

4.3 Software Development

Software Development is the second stage of Software Engineering as defined by Sommerville (2016), and is often carried out by men due to the in-balance in the field, with just 11% of software developers being women (Criado Perez, 2019).

Beginning with the interpretation of the requirements, this may be affected by language in both how they have been written, and how they are understood, and as, again, this is done through language, this must also be researched further to find any potential areas of difference regarding gender. It is the subtleties here which could prove telling.

Regarding the programming itself, how men and women code and approach the problem has also previously been shown to be different (Terrell et al., 2017; Ashcroft, 2018), but how this may be affected by "Hedging" remains to be seen. Although language will have an impact on their interactions throughout the process (Ashcroft, 2018), for example in discussing how a problem or requirement should be approached, if and how the code itself differs due to a direct impact of "Hedging" remains to be seen. Previous work in this area does however indicate that approach in the code will differ based on gender and single sex groups (Ashcroft, 2018), the examination of differences when mixed groups are observed could also prove interesting.

Although existing literature shows that development is affected by gender, the specificity of the effect "Hedging" has on this remains to be seen, and definitely requires further investigation.

4.4 Software Validation

Described as the third stage of SE, Software Validation is vital to ensure the software that has been created meets the requirements outlined at the beginning of the process (Sommerville, 2016) (see Section 4.2), therefore, not only must this

step allow for equal representation and equity, but it relies on the same being said for the specification stage. Though testing takes many forms, some of which can be automated, there is also the need to involve as diverse a range of testers, as there are users. Examples of where this has not been the case are varied and ever prevalent in both business and the media, e.g. an Apple Watch battery not lasting for customers with darker skin tones (BBC, 2015).

When it comes to how language can affect the Software Validation stage, how users are asked, and listened to in their responses should be carefully planned, considered and analysed. It is still, of course, important that testers are listened to and comments noted if there is a problem with the tested software, but also that their comments are taken seriously regardless of whether "Hedging" is used. This may of course, not be the case, but again further research is needed to uncover any differences.

4.5 Software Evolution

The stage of Software Evolution recognises the need for continued development and support of systems once they are built and implemented (Sommerville, 2016). This software being correct and appropriate for use, is of course dependent on the correct specification, development and validation; but this stage may also leave room for a lack of input, or a misinterpreted output from all stakeholders. This could, for example, reflect similar practices to problems being raised, and the rest of the process being cycled through; and therefore all the above potential implications of "Hedging" once again apply.

4.6 The Potential Effect of "Hedging"

The overall process of ideation through to implementation involves discussion and the use of language throughout. Although the amount of discussion may vary from stage to stage, and this may also vary between organisations, what is consistent in practice and literature, is that this does occur. With existing research on language, gender, and "Hedging", it seems that there is a gap in the literature when it comes to how these areas affect the Software Engineering process in its entirety. Although literature has begun to emerge on its effect on the design process (Ashcroft, 2021), the rest of the process, from the need for software to its evolution, remains to be researched further with regards to gendered language. The effect of "Hedging", or other existing known gendered differences in language, should be examined alongside each process within Software Engineering to fully uncover "Hedging" as a feminist issue in Computer Science (as discussed further in Section 6).

5 Discussion

Software development, when done well, is a process in which designers, developers, stakeholders and users are involved throughout the entire process.

Therefore it stands to reason equal representation is crucial in the designing and creation of software, as the users of the software themselves are often a diverse range of people. However, it is clear that representation and being in the room is simply not enough. There is very little reason to have representation in the room, if they are not heard, respected, or even given the opportunity to speak. The role "Hedging" plays in this is one of vital importance, and is neither the fault of the individual men or women in the room, but a product of how they were all raised- for example men are significantly more likely to express opinions as certainty, in comparison to women (Holmes, 1986). The societal and systemic raising of children into adults in this way continues to perpetuate a cycle where men feel they are more confident in their opinions, and more likely to express them as facts; and women are much more likely to use "Hedging" as a linguistic tool to express uncertainty, perhaps when there is none, as a means of *taking up less space*.

Therefore, the impact that this has on software, the tools used by many in their day-to-day lives, as part of social interactions, work and for countless other reasons, cannot be understated. If women in design meetings are not listened to, due to the use of "Hedging" or otherwise, how can it be guaranteed that the software, or even hardware, that is created is appropriate for use? There are countless examples of technology not being suitable for wider groups making it into the market, with a release of the aforementioned Apple Watch's battery life not lasting for those with pigmented skin (BBC, 2015), or phones being designed for the average size of men's hands and not women's (Criado Perez, 2019). These are products designed by large companies with the budget and resources to ensure that these mistakes are not made, and yet consistently they are. Which leads this author to assume it is not a lack of ability, but a lack of willingness. With men being more likely to achieve positions of leadership, regardless of past experience (Reuben et al., 2012), how can it be ensured that their best interests are to support those who do not reflect their experience of life? It could, of course, be argued that empathy is a means of overcoming this, and by being able to place themselves in the shoes of others they would be able to see what is needed from the perspective of all potential users, which could also be argued to be the trait of simply a good designer. However, this only works if all information is openly and willingly shared and known, a designer does not know what they don't know e.g. if a designer does not know that on average women's hands are smaller than their own men's hands, how would they then practice empathy in order to change their design?

The above makes the case for representation in the room but does not discuss what happens once equal representation is present, the situation is arguably only slightly improved if when people of varying genders, races and backgrounds are in the room, and will only be significantly improved once they are listened to. As discussed in Section 4, the effect "Hedging" can have on each stage of the Software Engineering process is potentially quite large, but needs to be researched further to be understood on a more complex level. Furthermore, when considering Feminist Methodologies, as discussed in Section 2, these must also be reflected within the scope of Language and HCI more specifically. Whilst this paper has presented a

theoretical understanding of areas for further research, it is vital that this is carried out, as outlined below.

6 Further Research

Though this paper has examined the existing literature surrounding "Hedging", Feminist CA and Software Engineering, there is still a clear need for further research in a number of areas. Therefore, this author recommends five areas for future research.

1. "Hedging" and the need for software. To what extent, if at all, does "Hedging" affect the need for software being raised? As discussed in Section 4.1, could the impact of "Hedging" on discourse effect which problems or ideas for development are brought forwards and how they are perceived? A large scale study of organisations of varying sizes, over time, would be needed to uncover this and best practices if any problems are found.

2. "Hedging" and software specifications. The impact "Hedging" has on software specifications being created, including both the miscommunication in the recording of requirements and their interpretation (as discussed in Sections 4.2 and 4.3) should be examined in closer detail. If specifications are not accurately recorded due to the gender of those raising them, or if they are misinterpreted when the development begins, this is definitely a gendered and feminist issue, and should therefore be investigated more thoroughly.

3. "Hedging", design and coding. In the software development stage (see Section 4.3), there are a number of areas that could be affected by "Hedging". Coding differences, when an analysis of GitHub pull requests was carried out, have been found when it comes to gender (Terrell et al., 2017) and whether this has any link to "Hedging" has yet to be investigated. It could be that these areas are not at all linked, but this should be researched to know for certain and to allow for any subtleties to be examined in closer detail. Furthermore, when it comes to the UX design, which takes place over the specification and development stages, how is this impacted by "Hedging"? Although some research has started to develop in this area (Ashcroft, 2021), it has only been done on a small scale and not in active development or design teams, this should be looked at in practice.

4. "Hedging", validation and evolution. When it comes to software validation/testing (see Section 4.4) it is important to consider; does "Hedging" affect how seriously comments are taken? Even under the assumption that there is a wide range of testers or people carrying out the validation. Then, assuming representation across the board is present, are participants' issues or concerns addressed? Furthermore, on the topic of fair representation of users in the room, there seems to be a landslide agreement in the CSCW and HCI community that this is important, but there is seemingly no framework in place to support this that can be adopted by organisations. In addition to this, as discussed above, this should be done alongside the provision of training and support regarding compassion and empathy, so that validation is not the only means of ensuring

products are fit for use, but one of many. Therefore if one means of overcoming inequality should fail, marginalised groups (including underrepresented genders) are not further disadvantaged. When it comes to software evolution (as discussed in Section 4.5), an area for further research comprises of if "Hedging" increases the misinterpretation when evolution is discussed, and as raised above, in all other stages.

5. "Hedging" and methodologies. Finally, when it comes to all of these above areas of further investigation, how appropriate are the existing methodologies being practiced to uncover the effect of gender on the Software Engineering Life-cycle? An example of where Feminist Methodologies have been applied to an existing methodology is that of Conversation Analysis and the work done by Stokoe (2004). Taking an existing practice, or designing a new one, and ensuring it does not contain any existing bias from preexisting potentially patriarchal research, is vital to make sure that any further research done in the area of Computing, is not perpetuating this (as discussed in Section 2).

7 Contribution and Conclusion

What needs to be considered in more detail within HCI and CSCW is the subtleties of social interaction and the impact they will have on the entire development process. Similar to the work of Stokoe (2004) on Feminist Conversation Analysis, this needs to be applied within the realm of Computing, with enough attention given to the detail of not only *what* is being said, but *how* it is being said when a conversation takes place. This paper has dissected each component of the Software Development Life-cycle from a feminist standpoint and understanding of feminist linguistics, and poses that a more subtle version of this is required; i.e. it is not really what people say, but how they say it.

Theoretically, this paper contributes by taking CA theories, such as interruptions and transitions, and considers these through the lens of existing Feminist Methodologies, such as the consideration of power dynamics. Taking this example further, CA has been known to consider power but never has taken this further because power is not the focus (Button and Sharrock, 2016). The question that should be asked is, how can these power dynamics lead into gender in this way? The practical focus on whether women, and other under-represented genders, have something to say, and the means in which they say it and this is perceived, will have an impact on the products created. The subtlety discussed in this paper will have implications for theory, and therefore further research into the effect gender has on the need for software, specification, design and coding, validation and evolution, and methodologies must all be carried out with these nuances in mind.

Acknowledgments

With thanks to my PhD supervisors Dr Mark Rouncefield and Dr Lynne Blair for their support and guidance, and the reviewers for their comments on structure. I would also like to thank Dr Emily Winter for first bringing the concept of "Hedging" to my attention several years ago.

References

- (2021): 'Kickbox Foundation'.
- (2022): 'CSCW Theories and Concepts'. EUSSET Colloquium.
- Agile Alliance (2020): 'What Is Agile Software Development?'.
- Ashcroft, A. (2018): 'A nice brain teaser'. In: *ACM International Conference Proceeding Series*.
- Ashcroft, A. (2020a): 'Gender Differences in Innovation Design'. *OzCHI '20, December 02–04, 2020, Sydney, NSW, Australia*.
- Ashcroft, A. (2020b): "'Hedging' and Gender in Participatory Design", no. 1986, pp. 176–180.
- Ashcroft, A. (2021): 'Do I belong here? An exploration of meeting structure and language, alongside gender and a sense of belonging.'. *OzCHI '21, November 30-December 2, 2021, Melbourne, VIC, Australia*.
- Barbour, K. (2018): *Embodied Ways of Knowing: Revisiting Feminist Epistemology*, pp. 209–226. London: Palgrave Macmillan UK.
- BBC (2015): 'Tattooed Wrists Can Stop Wearables Like The Apple Watch Working'.
- Bellini, R., A. Strohmayer, E. Alabdulqader, A. A. Ahmed, K. Spiel, S. Bardzell, and M. Balaam (2018): 'Feminist HCI: taking stock, moving forward, and engaging community'. In: *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. pp. 1–4.
- Benwell, B. (2006): *Discourse and identity*. Edinburgh, [England]: Edinburgh University Press.
- Blackwell, E. C. (2020): 'Taking Up Space: A Phenomenological Study of the Shared Experiences of Black Women in STEM Graduate Programs at Predominantly White Institutions'.
- Boden, D. (1994): *The Business of Talk: Organizations in Action*. Cambridge: Polity Press.
- Button, G. and W. Sharrock (2016): 'In support of conversation analysis' radical agenda'. *Discourse Studies*, vol. 18, no. 5, pp. 610–620.
- Catolino, G., F. Palomba, D. A. Tamburri, A. Serebrenik, and F. Ferrucci (2019): 'Gender diversity and community smells: insights from the trenches'. *IEEE Software*, vol. 37, no. 1, pp. 10–16.
- Criado Perez, C. (2019): *Invisible Women: Exposing Data Bias in a World Designed for Men*. Vintage, 10 edition.
- Davis, C. (2019): 'Feminist Rhetorical Practices in Digital Spaces'. *Computers and Composition*, pp. 132–141.

- De Hertogh, L. B., L. Lane, and J. Ouellette (2019): “Feminist Leanings:” Tracing Technofeminist and Intersectional Practices and Values in Three Decades of Computers and Composition’. *Computers and Composition*, no. 51, pp. 4–13.
- Dima, A. and M. Maassen (2018): ‘From Waterfall to Agile software: Development models in the IT sector, 2006 to 2018. Impacts on company management.’. *Journal of international studies*, pp.315–325.
- Dixon, J. and D. Foster (1997): ‘No Title’. *Journal of Psycholinguistic Research*,, vol. 26, no. 1, pp. 89–107.
- Flax, J. (1993): ‘Disputed Subjects: Essays on Psychoanalysis, Politics, and Philosophy’.
- Ford, C. (2008): *Women Speaking Up: Getting and Using Turns in Workplace Meetings*. New York: Palgrave.
- Friedrich, M. C. G. and E. Heise (2019): ‘Does the Use of Gender-Fair Language Influence the Comprehensibility of Texts?’. *Swiss Journal of Psychology*.
- Hancox-Li, L. and I. E. Kumar (2021): ‘Epistemic Values in Feature Importance Methods: Lessons from Feminist Epistemology’. In: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. New York, NY, USA, p. 817–826, Association for Computing Machinery.
- Heemstra, F. J. and R. J. Kusters (2004): ‘Defining ICT proposals’. *Journal of Enterprise Information Management*.
- Holmes, J. (1986): ‘Functions of You Know in Women’s and Men’s Speech Language in Society’. *Source: Language in Society*.
- Holmes, J. (1990): ‘Hedges and boosters in women’s and men’s speech’. *Language and Communication*.
- Hyland, K. (1996): ‘Writing without conviction? Hedging in science research articles’. *Applied Linguistics*, vol. 17, no. 4, pp. 433–454.
- Jay, A. (2009): *How to Run a Meeting*, Harvard Business Review Classics. Harvard Business Review Press.
- Kitzinger, C. (2008): ‘Developing feminist conversation analysis: A response to Wowk’.
- Leau, Y. B., W. K. Loo, W. Y. Tham, and S. F. Tan (2012): ‘Software development life cycle AGILE vs traditional approaches’. In: *International Conference on Information and Network Technology*, Vol. 37. pp. 162–167.
- Liddicoat, A. J. (2011): *An Introduction to Conversation Analysis: Second Edition*. Continuum International Publishing Group.
- Murphy, B. (2010): *Corpus and sociolinguistics : investigating age and gender in female talk*. Amsterdam ; Philadelphia: John Benjamins Pub. Co.
- O’Leary, Z. (2017): *The essential guide to doing your research project*. SAGE, 3rd edition.
- Park, D. (1996): ‘Gender role, decision style and leadership style’. *Women in Management Review*, vol. 11, no. 8, pp. 13–17.

- Paulus, T., A. Warren, and J. N. Lester (2016): 'Applying conversation analysis methods to online talk: A literature review'. *Discourse, Context Media*, vol. 12, pp. 1–10.
- Reuben, E., P. Rey-Biel, P. Sapienza, and L. Zingales (2012): 'The emergence of male leadership in competitive environments'. *Journal of Economic Behavior and Organization*.
- Sacks, H., E. A. Schegloff, and G. Jefferson (1974): 'A Simplest Systematics for the Organization of Turn-Taking for Conversation'. *Language*, vol. 24, no. 2, pp. 111–134.
- Schlesinger, A., W. K. Edwards, and R. E. Grinter (2017): 'Intersectional HCI'.
- Sommerville, I. (2016): *Software engineering*. Boston, [Massachusetts]: Pearson, tenth edition, global edition. edition.
- Speer, S. A. and E. Stokoe (2011): *Conversation and gender*. Cambridge: Cambridge University Press.
- Stokoe, E. (2018): *Talk: The Science of Conversation*. Robinson.
- Stokoe, E. H. (2004): 'Gender and discourse, gender and categorization: Current developments in language and gender research'. *Qualitative Research in Psychology*, vol. 1, no. 2, pp. 107–129.
- Stokoe, E. H. and J. Smithson (2001): 'Making gender relevant: Conversation analysis and gender categories in interaction'. *Discourse and Society*, vol. 12, no. 2, pp. 217–244.
- Svennevig, J. (2012): 'Interaction in workplace meetings'. *Discourse studies*, vol. 14, no. 1, pp. 3–10.
- Tannen, D. (1994): *Talking from 9 to 5*. New York: Avon Books.
- Terrell, J., A. Kofink, J. Middleton, C. Rainear, E. Murphy-Hill, C. Parnin, and J. Stallings (2017): 'Gender differences and bias in open source: Pull request acceptance of women versus men'. *PeerJ Computer Science*.