Optimizing Yone’s Item Build: A Weighted Multi-Criteria Decision Analysis Approach to Maximizing Character Strength

Author: P.S

December 2024

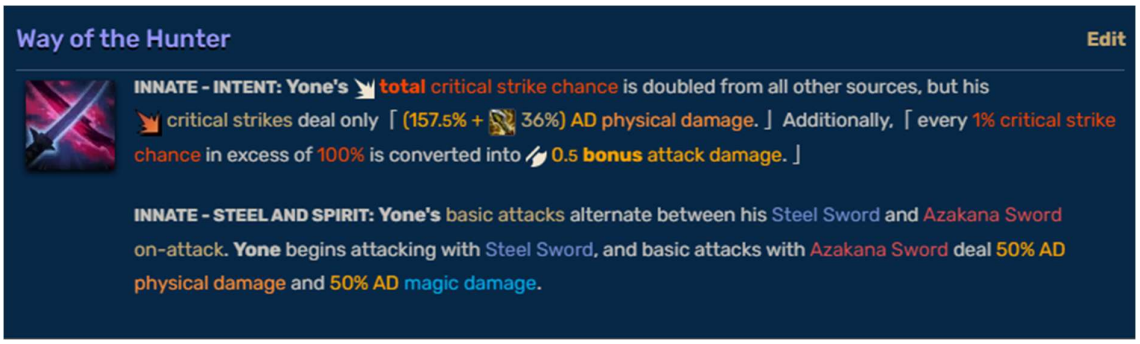
Optimizing Yone's Item Build: A Weighted Multi-Criteria Decision Analysis Approach to Maximizing Character Strength

Abstract:

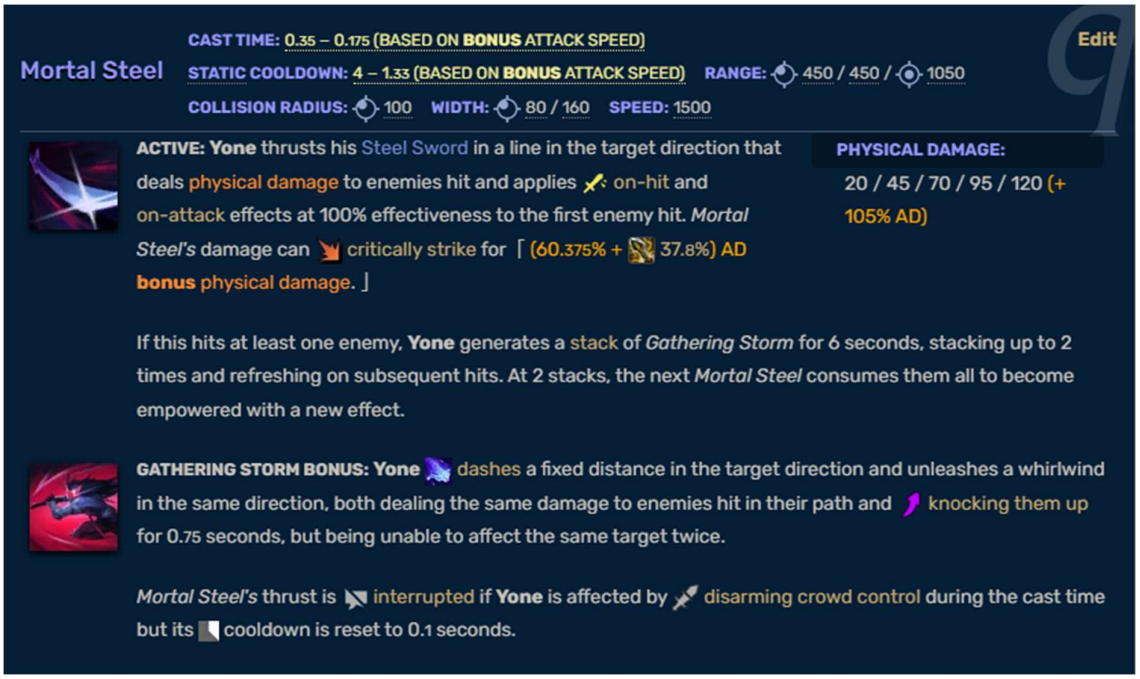
This paper is an attempt at arbitrarily determining the best possible build for Yone at full items. One note is that each stat's innate values will differ depending on game state and team composition. The scenario in which these calculations are made is within the assumption that Yone is max level, and the stats magic resist, armor, and health are valued similarly. The most optimal build is derived from a weighted stat distribution formed from an arbitrary criterion.

Character analysis

When figuring out the most optimal set of items to buy for a character, it's crucial to understand how each item augments the character's overall strength. As a start to determine the most valuable stats, we can investigate Yone's kit. Starting with his passive,

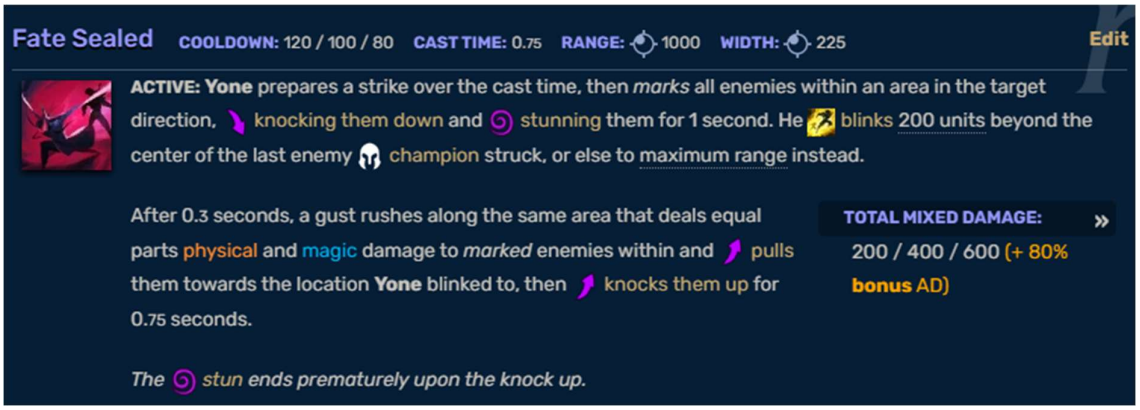


Yone's core passive doubles his critical strike chance while converting excess crit chance into 0.5 ad for every excess percent. His secondary passive converts every other attack into half ad and half ap damage. From the passive alone, we can see that Yone is incentivized to build crit and AD since he receives double value from buying critical strike chance although actual effective value may be lower due to a decreased critical strike damage.



Yone's Q is a core component of his kit due to its lower cooldown and its third cast acting as a mobility boost for Yone. The ability applies on-hit effects while also being able to crit. This leaves his build path open to an on-hit option, but typically on-hit builds tend to be deprived of crit. An important characteristic to note is that the Q cooldown scales with attack speed which makes the stat itself more valuable to Yone.

Yone's E is highly regarded as one of the most bullshit abilities in league of legends due to the true damage amp, mobility, and cleanse that is provided to Yone while also acting as a fail-safe. The only stats which affect this ability is ability haste which changes the cooldown of the ability. One important thing to note is that the true damage amps are still gated by resistances as it is based on cumulative damage in which Yone is applying ad and ap damage.



Yone's ultimate is an effective tool in breaking up enemy formations or securing kills due to its role as a near instant long range gap closing crowd control. This ability is affected by ability haste and bonus ad.

Stat weight determination

From the character kit analysis, we can see that Yone values attack speed, attack damage, and critical strike chance. Although This wouldn't be entirely accurate since it does not account for health, magic resist, and armor effectiveness. We will create a Yone strength value determined as a product of his dps, relative health, and healing percentage.

$$CPV = DPS_C \times \left(\frac{E(HP)}{E(HP_{18})} + \frac{E\big((DPS \times LS, AR, MR)\big) + E(HR, AR, MR)}{E(HP_{18})} + \frac{E(W_{shiel\ }, AR, MR)}{W_{cd} \times E(HP_{18})} \right)$$

Where function E is the raw health to effective health conversion. Since armor and magic resist have different effective conditions, weights will be assigned as ratios of physical and magic champions.

$$DPS_A(AD, AS, CC, CD) = AS(AD + AD\big(2 \times Crit\ chance \times (Crit\ damage - 1)\big))$$

$$E(HP, AR, MR) = HP \times (1 + \frac{AR \cdot \omega_{ar}}{100} + \frac{MR \cdot \omega_{mr}}{100})$$

$$\omega_{ar} = \frac{N_{physical}}{N_{physical} + N_{Magical}}$$

$$\omega_{mr} = \frac{N_{Magical}}{N_{physical} + N_{Magical}}$$

Considering Yone's passive, we get,

$$CC_{effective} = \begin{cases} CC\ if\ CC \leq 0.5 \\ 0.5\ if\ CC\ > 0.5 \end{cases}$$

$$AD_{bonus} = \begin{cases} 0 & if\ CC \leq 0.5 \\ 100 \cdot (CC - 0.5)\ if\ CC > 0.5 \end{cases}$$

$$AD = AD + AD_{bonus}$$

We will now calculate ability induced DPS and combine it with DPS from auto attacking,

$$DPS_P = \left(\frac{Q_d}{Q_{cd}} + \frac{W_d}{W_{cd}} \right) + \frac{0.35 \times (\frac{Q_d}{Q_{cd}} + \frac{W_d}{W_{cd}})}{E_{cd}}$$

$$DPS_C(L, RA_p, ARP\ ) = \ (DPS_p + DPS_A)(\frac{100}{100 + ((1 - ARP)(1 - RA_p)(AR_{enemy})) - \ L})$$

In the implementation we will input an enemy armor of 150.

$$Q_d = (120 + (1.05 \times AD))(2 \times CC \times (0.6037 + Ex\ Crit\ damage) \times (120 + (1.05 \times AD))$$

$$W_d = 50 + 0.14(2586) = 412$$

$$W_{shield} = 120 + (0.45 \times (AD - AD_{18}))$$

The Q damage is reflected as an average damage per Q when continuously attacking.

The cooldowns of W and Q respectively can be expressed as arrays at different break points,

$$W_{cd} = [14, 13.2, 12.4, 11.6, 10.8, 10, 9.2, 8.4, 7.6, 6.8, 6]$$

$$Q_{cd} = [4, 3.64, 3.28, 2.92, 2.56, 2.2, 1.84, 1.48, 1.33]$$

$$W_{thres} = [0, 9.46, 18.91, 28.37, 37.82, 47.28, 56.73, 66.19, 75.64, 85.1, 94.55]$$

$$Q_{thres} = [0.0, 0.15, 0.30, 0.45, 0.60, 0.75, 0.90, 1.05, 1.1111]$$

$$E_{cd} = 10 \times \frac{100}{100 + Haste}$$

Where W_{cd} and Q_{cd} are the respective ability cooldowns at different bonus AS percent break points. When using break points an issue arises where bonus attack speed is undervalued if it doesn't contribute to the Q and W cooldown directly so we will be using this formula for cooldown instead.

$$Q_{cd} = \begin{cases} 4 \times (1 - \left(0.01 \times \frac{AS_{bonus\ percent}}{1.67}\right) & if\ AS_{bonus\ percent} \leq 120 \\ 1.33 & if\ AS_{bonus\ percent} > 120 \end{cases}$$

$$W_{cd} = \begin{cases} 14 \times (1 - (0.01 \times \frac{AS_{bonus\ percent}}{1.51} & if\ AS_{bonus\ percent} \leq 94.6 \\ 5.25 & if\ AS_{bonus\ percent} > 94.6 \end{cases}$$

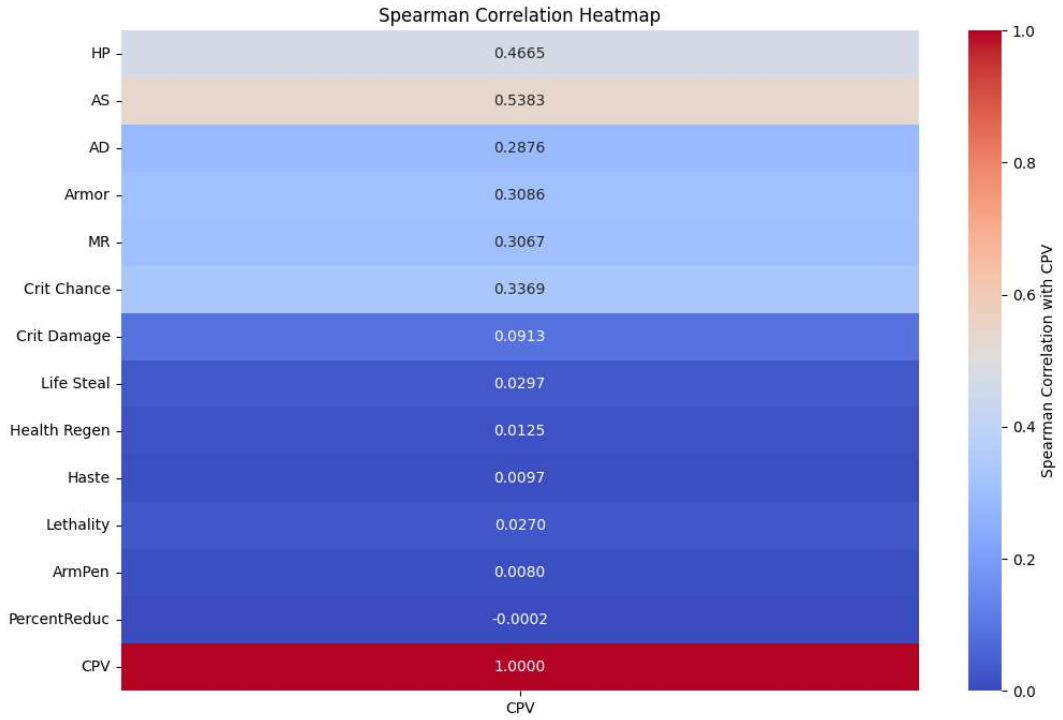
We will be creating a heatmap to understand how each variable influences the value of CPV and use this as our initialized weights. Additionally, we will be analyzing the gradient of each variable at different stat checkpoints to adjust the weights of each stat.

This is the heatmap we get when generating 50000 random samples for each variable within their respective ranges. We already figured out that attack speed would have been a high priority stat from the character kit analysis, but now we get a comprehensive understanding of how each stat is weighted.

Stat	Correlation	Normalized Weight
HP	0.4550	0.8815
AS	0.5162	1.0000
AD	0.2937	0.5687
AR	0.3032	0.5873
MR	0.3026	0.5862
CC	0.3241	0.6279
CD	0.0935	0.1811
HR	0.0100	0.0194
Haste	0.0080	0.0155
Life Steal	0.0314	0.0608
Lethality	0.0298	0.0577
Armor pen	0.0083	0.0161
A-Reduce	0.0004	0.0008

Further analysis of weight determination

An issue that arises with the use of correlation matrix is that certain attributes are calculated as being negatively impactful towards the character power value when in theory all stats should improve the character power value even by a small margin. To address this, we will be using multiple correlation methods (Pearson, Spearman, Kendall) to see if any results differ. Additionally, we will be producing confidence intervals by bootstrapping each correlation value.



Above We can see the spearman correlation method which tends to denote a similar trend with attack speed being the most valued stat. One thing to note across all heatmaps is that lethality's value seems to be inflated compared to the other means of armor penetrations which could be explained by our assumption that the target has only its base armor which may inflate the worth of lethality. Although not entirely inaccurate since Yone's primary targets tend to be low durability characters.

Item induced character strength change

We will define a new expression for character strength as a summation of their stats multiplied by their importance towards Yone. The following expression combines all stats and their respective weights to produce a single score for evaluation.

$$CS = \sum_{i=1}^{13} \sqrt{stat_i} \times W_i$$

$$\begin{aligned} CS + CS_o = {} & \left(\sqrt{HP} \times W_{HP}\right) + \left(\sqrt{AS} \times W_{AS}\right) + \left(\sqrt{AD} \times W_{AD}\right) + \left(\sqrt{AR} \times W_{AR}\right) + \left(\sqrt{MR} \times W_{MR}\right) \\ & + \left(\sqrt{CC} \times W_{CC}\right) + \left(\sqrt{CD} \times W_{CD}\right) + \left(\sqrt{HR} \times W_{HR}\right) + \left(\sqrt{H} \times W_{H}\right) + \left(\sqrt{LS} \times W_{LS}\right) \\ & + \left(\sqrt{L} \times W_{L}\right) + \left(\sqrt{ARP} \times W_{ARP}\right) + \left(\sqrt{A} \times W_{A}\right) \end{aligned}$$

With each iteration of character strength calculation, the previous character strength values are summed as an initial character strength input. The stats and weights are normalized to prevent large values from skewing the character strength value.

$$stat \max = \left(initial\ stat + \sum_{i=T(s)}^{5} v(s,i)\ \ \forall s \in S\right)$$

Where T(s) is defined as the 5 items with the highest stat s and v (s, i) is the stat s provided by item i. S is the set of stats

$$stat_i = \frac{stat}{stat\ max}$$

$$w_i = \left[\frac{\partial CS}{\partial stat_1}, \frac{\partial CS}{\partial stat_2}, \ldots, \frac{\partial CS}{\partial stat_3}\right]$$

$$W_i = W_o + learning\ rate\ \cdot \frac{N - R(w_i) + 1}{N}$$

For weight updates we use a rank-based normalization and then modify the previous iteration's weights by a percentage of the new weights dictated by the learning rate.

Passive induced character strength change

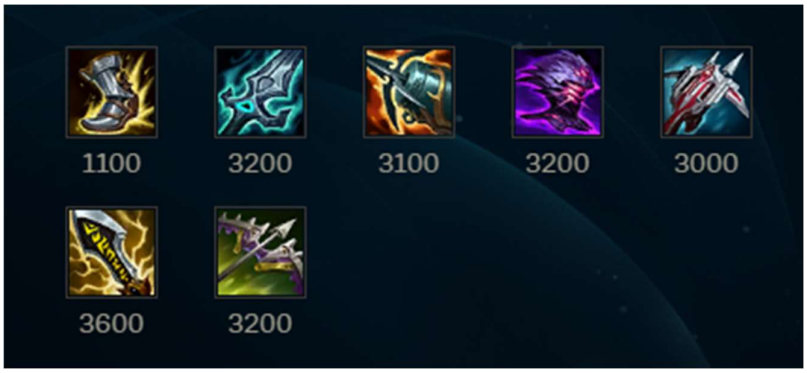
Legendary items in league of legends typically come with a special item passive that can augment a character's overall strength whether that be damage output, durability, healing, or mobility steroids. This study will not be considering the impact of dashes and blinks since these effects have varying degrees of effectiveness depending on player skill. Instead, we will look at the dps additions or survivability added through passives. For example, Hextech rocket belt will only have its dash damage considered. Similarly, items that scale off enemy health will instead use Yone's base HP in its added damage calculation. All passives will be simplified to stat sticks and added on to the item base stats. Although this does introduce multiple limitations since the effects of some item stats will vary depending on game state as such leaving the model open to inaccuracies caused by varying game conditions.

Results

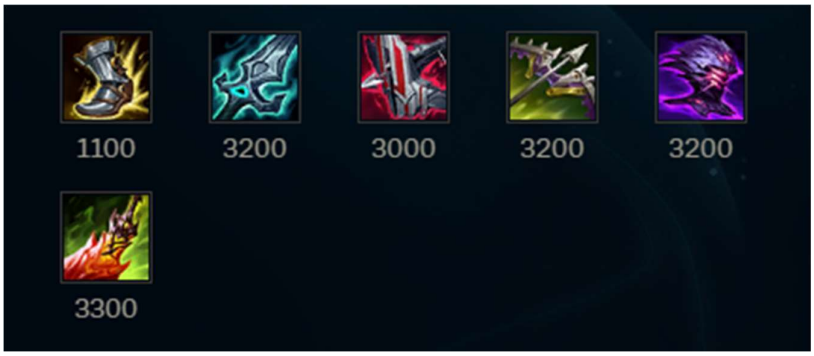
We will be comparing the results of the two models alongside the 1v9 Yone items from WayOfTheTempest as a reference for assessing how reliable some of these builds may be. For all builds it is assumed the berserkers greaves were bought first. The effectiveness of each build can only be truly understood from practical application which will also have varying performance due to player skill level or game conditions. Ultimately there isn't an objective best build. Additionally, these models do not consider build path and gold efficiency so it's up to the player's choosing which item to buy although I will attempt to provide a reasonable build path for each model.

Where lord Dominik's regards and mortal reminder are interchangeable depending on what you need. In theory this doesn't look completely horrible considering that it has all the stats Yone needs except a severe lack of AD in the early stages. Additionally, Navori's presence here may be a slight inaccuracy since its passive is converted into a rough estimate of raw ability haste value. If we exclude all item passive simplification, phantom dancer will take Navori's place due to higher base stats. When looking at the build path starting Navori's is probably the right choice since capping Q cooldown is especially important in laning and skirmishing. The only issue that arises from this is lack of AD which means that the second item must be Infinity edge or lord Dominik. In this case, we will be choosing lord Dominik's for a cleaner build path. The third and fourth items should be either blood thirster or Jak'Sho depending on what game state you're in. If the enemy damage is very burst structured, then going Jak'Sho would be preferable.

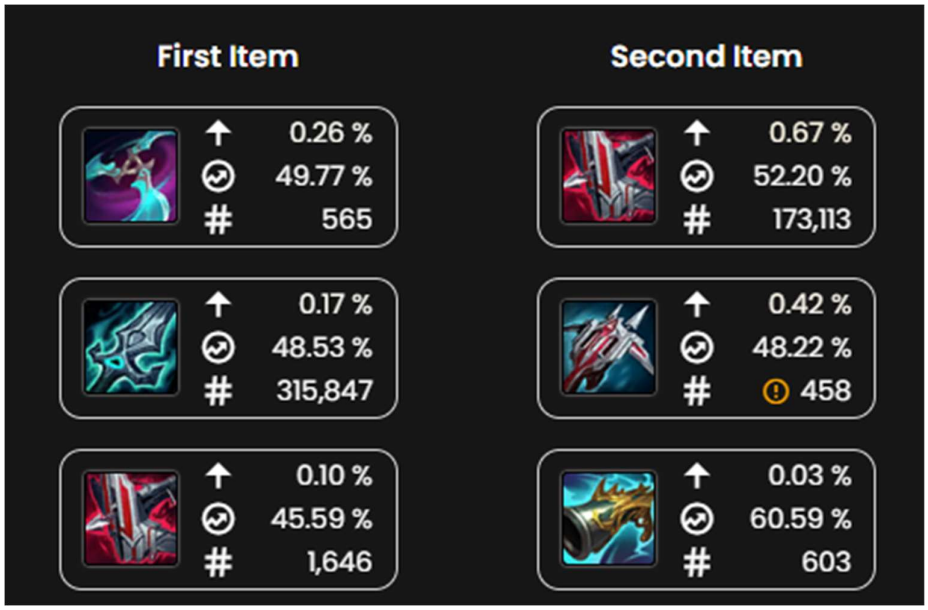
The second model provided a more believable build with the presence of blade of the ruin king which every Yone builds. We also see here that there is a tank item which suggests that the model was able to capture the importance of defensive stats as opposed to relying on life steal purely.



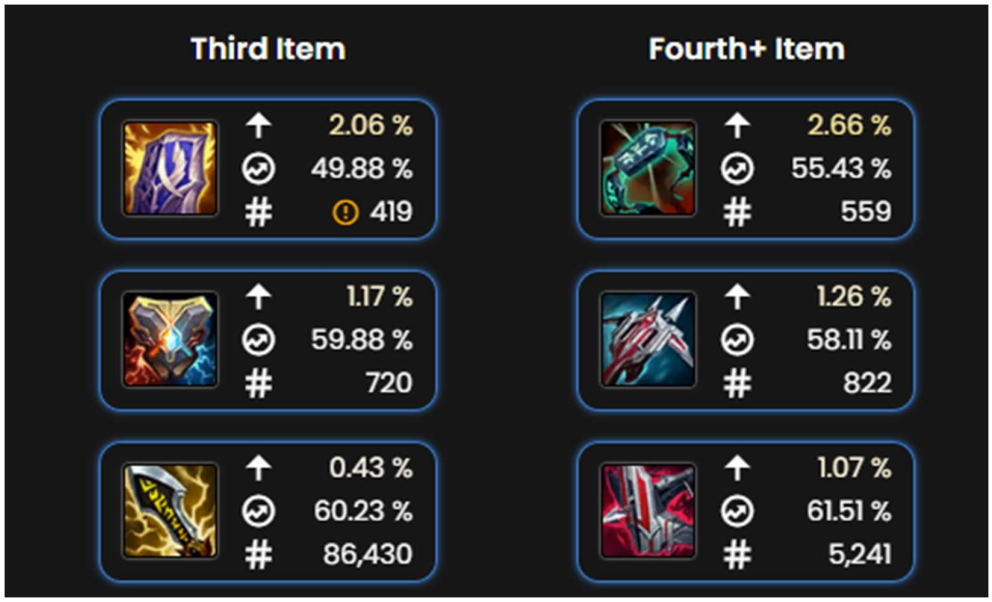
In this build, our two potential first item would be blade of the ruin king and kraken slayer since they are our only source of attack speed. Depending on the match up, you'd either take bork or kraken first. Third item should be Jak'Sho to provide balance between offensive and defensive stats. The last two items are dependent on what is needed although I think it's preferable to go Infinity edge last or else you can't utilize the full effect of its crit damage amp.



The reference build is the one I got from WayOfTempest and if we compare it to the two builds we have generated, it seems that the second model is more in line with the norm since we see a balance of offensive and defensive items. One important note is that Tempest's build has more defensive utility than the builds generated which could be explained by our models assuming that one shot and anti-heal do not exist. A potential way to implement this is to create a probability value that is a ratio of current effective health and current dps which represents the chance of being blown up. As another reference point, we will be using Coachless to compare the highest win probability items at different slots.



When looking at the highest probability items in Coachless, we can see that our first model isn't entirely off. Navori's and Lord Dominik's regards are in the top 2 for first and second item.



As for the third and fourth item, we can see that defensive items are highly valued which our model was able to capture by including Jak'Sho, but still falls short in providing a balance between DPS and durability. This may be due to the model overvaluing life steal since in the algorithm, anti-heal does not exist.

Conclusion

This study explored the optimization of Yone's item build using a weighted multi-criteria decision analysis (MCDA) framework. Through the examination of Yone's kit and the mathematical modeling of character power values (CPV). While the models demonstrated a reasonable ability to propose viable builds, they also highlighted notable limitations and areas for improvement.

Key findings revealed that attack speed consistently emerged as the most significant contributor to Yone's overall power, aligning with his passive and kit mechanics. However, the models struggled to fully account for in-game nuances such as the value of defensive utility, the inclusion of anti-heal, and the dynamics of one-shot potential in various matchups. Simplifications, such as treating item passives as static stat sticks, introduced further inaccuracies.

The generated builds provided insight into potential paths for maximizing Yone's effectiveness. The first model provided a less conventional build, while the second model produced a more traditional build, maintaining popular Yone items like Blade of the Ruin king and Immortal Shield bow. These builds compared favorably with community reference builds, such as those from WayOfTheTempest and Coachless, though the lack of nuanced considerations limited their practicality in real-game scenarios.

Future iterations of this analysis could benefit from incorporating additional variables, such as the probability of one-shot scenarios and the effectiveness of anti-heal, to enhance the model's accuracy. Additionally, factoring in build path efficiency and gold cost could provide more actionable recommendations for players at various stages of the game.

Ultimately, while this study highlights a robust framework for systematically evaluating item builds, it also affirms that the dynamic nature of League of Legends, coupled with factors such as player skill and game state, precludes the existence of a universally optimal build. Instead, the insights derived from this analysis serve as a foundation for informed decision-making, enabling players to adapt their builds to the ever-changing demands of the game.